

# Modélisation des incertitudes concernant la géométrie des failles. Impact sur les études volumétriques des réservoirs pétroliers.

## THÈSE

pour obtenir le grade de

**Docteur de l'Institut National Polytechnique de Lorraine**

**Discipline: Géosciences**

présentée et soutenue publiquement

par

**Magali LECOUR**

le 23 juin 2000

---

**Directeur de thèse: Jean-Laurent MALLET**

---

### Composition du jury

*Rapporteurs:* Jean-Pierre GRATIER  
Isabelle MORETTI

*Examineurs :* Jean-Claude DULAC  
Jean-Laurent MALLET  
Pierre THORE



---

# Remerciements

Mes remerciements s'adressent tout d'abord au **Professeur Mallet**, responsable du L.I.A.D. et du projet gOcad qui en constitue le coeur, sans qui ce travail n'aurait jamais vu le jour. En dix ans, il a réussi à créer un projet solide, et monté une équipe dont les relations vont bien au delà des simples relations de travail. Les conditions dans lesquelles j'ai pu travailler depuis maintenant plus de cinq ans sont assez exceptionnelles pour être soulignées. Parmi les nombreux avantages à travailler avec cette équipe, on peut citer le caractère international de ce projet de recherche, qui permet aux étudiants de confronter leurs résultats plusieurs fois par an aux critiques des utilisateurs, et d'avoir un échange constructif avec des géologues confirmés.

Je remercie aussi les autres membres du jury qui ont accepté de valider mon travail, et de critiquer ce mémoire. Les deux rapporteurs **Isabelle Moretti** de l'Institut Français du Pétrole et **Jean-Pierre Gratier**, professeur de l'université de Grenoble, ainsi que **Pierre Thore** de la compagnie Elf Exploration Production, et **Jean-Claude Dulac** de la compagnie TSurf Corporation.

A ces remerciements, s'ajoutent ceux adressés aux **membres du consortium** qui supportent financièrement le laboratoire depuis une décennie et contribuent aux bonnes conditions dans lesquelles se déroulent nos thèses.

Parmi ces interlocuteurs privilégiés, je tiens particulièrement à remercier les membres du *Projet Incertitudes* de la compagnie **Elf Exploration Production**, à Pau. Tout d'abord **Pierre Thore**, qui, avant de s'engager pour ce travail de thèse, m'a accueillie durant mon stage de D.E.A.. A ses cotés, j'ai appris énormément de choses sur les problèmes réels rencontrés quotidiennement dans les compagnies pétrolières, notamment en ce qui concerne les données sismiques. Ayant une vue beaucoup plus globale des problèmes à résoudre, il a su m'amener doucement à ce domaine qui m'était totalement inconnu.

Je tiens aussi à remercier **Bernard Corre**, responsable du *Projet Incertitudes*, qui, malgré un emploi du temps très chargé, a plusieurs fois pris le temps de m'expliquer les différents volets du projet, me permettant ainsi de prendre conscience de la complexité du problème général des incertitudes. Ce n'est qu'après une de ces conversations que j'ai réalisé l'ampleur et l'ambition de ce projet, dans lequel mon travail représente bien peu de choses.

Je ne peux penser à ce projet sans évoquer **Vivien de Feraudy** - *merci pour le survol des Pyrénées, c'était magique !-* et **Gilles Vincent**, dont le calme et les conversations passionnées sur les Pyrénées donneraient à beaucoup l'envie de s'installer dans la région.

Mes nombreux séjours à Pau n'auraient pas été les mêmes sans les conversations à bâtons rompus avec **Philippe Samson**, et sans l'aide apportée par **Francis Klein** pour l'installation des nouvelles versions de mon travail.

Enfin, je tiens à remercier les différents géologues et géophysiciens qui ont accepté de me consacrer un peu de leur temps, afin de me donner un aspect pratique des problèmes qu'ils rencontrent chaque jour (**Denis Schnapper** qui a coordonné les rencontres lors de mon stage D.E.A., **S. Gansarski**, **H. Morizot**, **A. Ferrer**, **A. Bossert** et **H. Pourtal** pour ce qui concerne les interprétations sismiques ; **Naamen Keskes** pour les démonstrations de sismage, et les conversations qui ont suivi; **Thierry Rives**, responsable du Projet Failles, **J-P. Gomez**, ainsi que **Gérard Ionesta**, véritable mine de renseignements).

Je garde un merci spécial pour **Sylvain Calassou**, qui s'est révélé être un interlocuteur exceptionnellement ouvert, et avec qui j'aurais aimé pouvoir collaborer beaucoup plus. J'espère que ce n'est qu'une partie remise...

Il est impossible de parler des personnes ayant participé à ce travail sans citer **Jean-Claude Dulac** (président de la compagnie TSurf) et **Isabelle Duvinage**, avec qui je joue aux «duettistes» depuis cinq ans, et avec qui ce fut un plaisir de collaborer (pourvu que cela dure le plus longtemps possible !).

Ces séjours à Houston restent de très bons souvenirs, pour des raisons diverses et variées. Parmi elles, je retiens l'énergie que l'on ressent là-bas, et les pauses café avec **Nathalie**, que j'ai énormément appréciées.

Je remercie aussi **Richard Cognot** pour m'avoir accompagnée au début de ma formation de «gOcadienne» et avoir gardé patience devant mes nombreuses questions de débutante. Il a ensuite complètement écrit le prototype de l'objet faille avant le démarrage de cette thèse. Ce prototype a non seulement permis de vérifier la faisabilité du projet, mais surtout beaucoup accéléré le travail par la suite.

Des remerciements sincères sont aussi adressés à **Arben Shtuka**, qui a non seulement codé les fonctions géostatistiques utilisées dans ce travail, me les a ensuite expliquées, puis m'a assistée tout au long de la thèse sur les problèmes liés aux simulations. Ses connaissances en ce domaine sont si bien maîtrisées, que les explications se passent volontiers de formules mathématiques, qui, je l'avoue, ne sont pas mon domaine de prédilection.

Pour palier ce manque, je dois un grand merci à **Sophie Viseur**, avec qui ce fut un réel plaisir de partager le même bureau. Son aisance à formaliser les choses, son esprit synthétique et ses nombreuses corrections m'ont beaucoup aidée, et j'espère pouvoir, à ma manière, lui rendre la pareille lors de la rédaction de sa propre thèse (- incessamment sous peu! ;-)).

Des explications claires m'ont aussi été données régulièrement par **Bruno Levy**, que je remercie tout particulièrement pour sa gentillesse, et sa disponibilité. Ces qualités en font l'un des pôles d'attraction du laboratoire (à tel point qu'il faut souvent prendre rendez-vous pour échanger quelques idées, tant sa pédagogie et sa curiosité scientifique font d'adeptes...).

Je remercie aussi tous les autres membres de l'équipe, déjà partis (**Karen Pairazian**, **Ariane Coupel**), arrivés en cours de route (**Pascal Audigane**, **Stéphane Conreaux**, **Mathieu Dazy**, **Olivier Grosse**, **Jérôme Massot**, **Guillaume Caumont**, correcteur assidu, **David Ledez**), permanents (**Pierre Jaquemin**) ou de passage (**Yannick Boisseau**, **Stéphane Müller** [à quand la prochaine fondue?]), qui, chacun à leur manière, ont contribué à la bonne humeur ambiante. Cette équipe ne pourrait fonctionner sans deux travailleuses de l'ombre. Je pense à **Nathalie Flammang** (ingénieur système) et **Monique Cugurno** (secrétaire de choc) qui nous permettent de travailler tranquillement, sans avoir à se préoccuper des tracasseries liées aux problèmes de réseau (quasi-inexistants, ou inconnus de ma part) ou administratives (nombreuses, mais souvent résolues sans qu'on le sache).

Parmi les nouveaux venus, je souhaite bonne chance à **Cécile Audinet**, que j'ai eu le plaisir de suivre durant la dernière année. Elle m'a aidée à déceler de nombreux points faibles dans les méthodes de simulation, et s'est ensuite attachée à les résoudre. Même si les résultats ne sont pas encore tout à fait complets, elle m'a aidé à clarifier les choses et ainsi permis de mettre sur papier la méthodologie à utiliser.

Je n'oublie pas mes futurs collègues de chez TSurf, qui ont de près ou de loin participé à ce travail, que ce soit en France (**Joël Conraud** et **Olivier Mariez**, à qui mes questions sont adressées le plus souvent désormais; **Thierry Valentin**, assailli de questions toute la journée, mais toujours disponible; **Nicolas Euler**, nouveau responsable du projet structural entamé avec **Isabelle Duvinage**; **Taoufik Aït Ettager**, **Karine Eludut**, **Etienne Cherrier**, **Philippe Pluyaud**, et **Didier Donner**), ou Outre-Atlantique (**Alexandra Cacheux**, **Amy Cheng**, **Nathalie Dulac**, **Fabien Bosquet**, **Stanislas Jayr**, **T.K. Huang**, **Boina Lakshmi**, et **David Segonds**).

Enfin, ces remerciements ne sauraient être complets si ma famille n'était citée. J'ai gardé le meilleur pour la fin: un grand merci à *mes parents*, à *Céline* et à *Fabien*.

---

# Résumé

Les études d'incertitudes constituent une phase clé de l'exploration pétrolière. Les données disponibles pour modéliser les réservoirs situés en profondeur étant souvent indirectes, éparses et imprécises, les incertitudes résultantes concernent de nombreux paramètres: la géométrie du réservoir, la répartition des faciès et les propriétés des couches.

Afin de diminuer les risques, la phase d'exploration doit proposer non plus un, mais plusieurs modèles balayant l'étendue des solutions possibles dans la zone étudiée. Cependant, les données concernant la géométrie des failles étant moins précises que pour les horizons, les règles mécaniques à respecter plus strictes, et la mise à jour du modèle structural plus complexe en milieu faillé, encore peu d'études ont été réalisées dans ce domaine.

Le travail présenté dans ce mémoire, vise la prise en compte d'une partie des incertitudes concernant la géométrie des failles: la modification de leur géométrie (position, pendage, forme en carte) autour d'un modèle de référence, à scénario de réseau de failles constant.

La plus grande partie de ce travail a consisté à mettre en place de nouvelles structures:

- un **objet faille** dont le rôle est de stocker les estimations de l'incertitude et de calculer rapidement une nouvelle position pour chaque point à l'aide de champs de probabilités
- un **gestionnaire de contacts** entre les différentes surfaces du modèle géologique et permettant de mettre à jour le modèle structural après chaque simulation
- un **gestionnaire de simulations** pour l'ensemble du réseau de failles, permettant aussi de mettre à jour le modèle structural.

L'objectif principal est de proposer un algorithme général permettant soit de voir l'influence de ces incertitudes sur l'estimation des volumes des réservoirs, soit de construire différents modèles géométriques représentant le réservoir, et pouvant servir ensuite à l'étude d'autres types d'incertitudes sur les propriétés des couches géologiques.

---

# Abstract

Uncertainty studies are currently a key point for petroleum exploration. Indeed, available data for deep reservoirs modelling are often uncertain, scattered and inaccurate. So, resulting uncertainties concern many parameters: the reservoir geometry, the facies distribution and rock properties.

In order to minimize risk, the exploration phase must give, not only one, but several models scanning all the possible range for the zone of interest. However, data concerning fault geometries are less precise than horizon data, fault modelling must respect more mechanical rules and structural model updating is more difficult in faulted domain, because more contacts must be preserved. Consequently, only few studies on fault geometry uncertainties have been performed at the time.

The work presented in this thesis, concerns one part of fault geometry uncertainties: the modification of their geometry (location, dip, and shape in map view) around a reference model, at constant fault network scheme.

Most of this work consisted in creating new data structures:

- a **fault object** that stores uncertainty estimations and allows for a fast computation of new locations for each fault point. The simulation method is based on the Pfield technique and try to preserve the initial geometry at best
- a **contact manager** storing all the contacts between the structural model interfaces (faults, horizons and boundary of the model). This structure allows also the model updating after each simulation loop
- a **fault network simulation manager**, that not only simulate new geometries for the faults, but also update the structural model afterwards.

The main goal of this work is to propose a global modular algorithm. This algorithm can give either the possibility to follow the evolution of reservoir volume estimations during the simulation process, or the possibility to select several realizations of the structural model geometry. The last option enables reservoir engineers to study other uncertainties like rock properties (porosity, permeability), or oil-water contacts.

---

# Table des matières

Remerciements .....	i
Résumé .....	iii
Abstract .....	iv
<b>Introduction .....</b>	<b>1</b>

## **PARTIE 1: Présentation du sujet**

<b>Chapitre 1 Présentation du cadre de travail.</b>	<b>7</b>
<b>1. Le projet incertitudes .....</b>	<b>7</b>
<b>2. Brève présentation du logiciel gOcad .....</b>	<b>10</b>
2.1. Un géo-modeleur .....	10
2.2. Une collection de structures simples représentant les différents objets géologiques.	11
2.3. Des interactions entre ces objets par l'intermédiaire du moteur d'interpolation: D.S.I. ....	14
<b>3. Avantages de travailler avec un tel géo-modeleur .....</b>	<b>16</b>
3.1. Langage C++ et héritage des structures existantes .....	16
3.1.1. Notion d'objet .....	16
3.1.2. Notion d'héritage. ....	18
3.2. Autres avantages plus spécifiques à gmcad .....	20
<b>4. La compagnie T-Surf. ....</b>	<b>21</b>

<b>Chapitre 2 Modélisation des failles: techniques disponibles</b>	<b>23</b>
<b>1. Qu'est-ce qu'une faille? .....</b>	<b>24</b>
1.1. Généralités .....	24
1.1.1. Nomenclature: quelques mots clés .....	24
1.1.2. Les différentes classifications des failles .....	26
1.1.3. Les différentes échelles d'étude .....	27
1.1.4. Les différents types de pièges associés aux failles .....	27
1.1.5. Différentes techniques d'étude .....	28
1.2. Grandes catégories de styles tectoniques en domaine extensif .....	32
1.2.1. Horst et Graben .....	32
1.2.2. Blocs basculés .....	32
a. Description de la géométrie en coupe .....	33
b. Description de la géométrie en carte .....	33
c. Blocs basculés en extension oblique .....	34
1.2.3. Blocs synthétiques .....	35
1.2.4. Les failles listriques .....	35
a. Caractéristiques en coupe .....	36
b. Caractéristiques en carte .....	36
c. Origine des failles listriques .....	37

1.3. Autres styles tectoniques .....	38
1.3.1. Domaine compressif .....	38
a. Un peu de vocabulaire... .....	38
b. Les familles de failles inverses .....	39
c. Importance de la lithologie dans la formation des pièges .....	40
1.3.2. Bassin sur décrochement .....	41
1.4. Discussion sur les règles ainsi établies .....	42
<b>2. Modélisation de la géométrie des failles. ....</b>	<b>44</b>
2.1. Hypothèse et données nécessaires .....	45
2.1.1. Hypothèses .....	45
2.1.2. Données nécessaires .....	46
2.2. Construction géométrique. ....	47
2.2.1. Construction de Verrall (Chevron construction) .....	47
2.2.2. Méthode de Davison .....	47
2.3. Méthode analytique de White (1986) .....	49
2.4. Discussion de l'application de ces méthodes au sein de nos travaux .....	52
2.4.1. Hypothèses trop fortes? .....	52
2.4.2. Seulement pour les failles listriques en domaine extensif? .....	52
2.4.3. Conservation de la masse dans une section 2D: et la 3D? .....	52
2.4.4. Comment tenir compte de données? .....	52
<b>3. Conclusion: les choix qui ont été faits pour ce mémoire .....</b>	<b>53</b>
3.1. Faire confiance au travail du géologue structuraliste: faille de référence utilisée comme une donnée. ....	53
3.2. Travaux envisageables .....	53
3.2.1. Utilisation de bibliothèques de formes .....	53
3.2.2. Utilisation des techniques précédentes: travaux de Cyril Galéra .....	54
 <b>Chapitre 3 Bilan des incertitudes sur les failles: .....</b>	 <b>55</b>
<b>1. Les différentes étapes précédant la modélisation .....</b>	<b>55</b>
1.1. Acquisition des données .....	5
1.1.1. Campagnes sismiques .....	55
1.1.2. Formation des sections stacks .....	56
1.2. Traitement des données .....	56
1.2.1. Préparation des données .....	56
1.2.2. Migration .....	56
1.3. Interprétation des données .....	58
1.3.1. Création d'un premier maillage cache. ....	58
1.3.2. Pointé des différentes structures dans les 3 directions de l'espace .....	59
1.3.3. Vérification de la cohérence .....	60
1.3.4. Utilisation de cartes d'attributs .....	61
1.3.5. Validation de l'interprétation .....	61
<b>2. Bilan des sources d'incertitude .....</b>	<b>62</b>

2.1. Quelques définitions	62
2.2. Incertitudes rencontrées durant la préparation des données	63
2.2.1. Acquisition et prétraitement	63
2.2.2. Formation du stack (sommation de plusieurs traces)	63
2.2.3. Migration	63
2.3. Incertitudes rencontrées durant la phase de calage sismique-puits	64
2.4. Incertitudes rencontrées durant la phase d'interprétation	64
2.4.1. Incertitudes liées à la présence même des failles	64
2.4.2. Incertitudes liées à la lithologie	66
<b>3. Conclusion sur les incertitudes autour des failles</b>	<b>66</b>
<b>4. Techniques existantes pour étudier les incertitudes structurales</b>	<b>69</b>
4.1. Méthode employée pour les horizons	69
4.2. Tenir compte des puits	70
4.3. Pourquoi cette méthode n'est pas directement applicable aux failles	71

## **PARTIE 2: Eude au niveau de la faille**

<b>Chapitre 4 Paramétrisation de l'Objet Faille:</b>	<b>73</b>
<b>1. Données disponibles</b>	<b>74</b>
1.1. Description des données utilisables pour construire les failles	74
1.2. Caractéristiques de ces données	75
<b>2. Présentation de l'Objet Faille</b>	<b>76</b>
2.1. La colonne vertébrale (Backbone)	77
2.1.1. Les lignes polygonales	77
a. Un ensemble de nœuds connectés	77
b. Atomic / AtomicGroup	78
c. fonctions d'édition	79
2.1.2. Héritage des fonctions d'édition des lignes polygonales	79
a. Notion d'héritage	79
b. Exemples d'héritages utilisés pour l'objet faille	80
c. Exemple d'algorithme utilisant l'héritage: ajout d'une génératrice	82
2.2. Les génératrices	83
2.2.1. Stockage sous forme de propriétés	83
2.2.2. Stockage dans une base locale	84
a. Définition de la base locale	84
b. Intérêt de la "base locale"	85
c. La classe Transformer3D	86
d. Inconvénients des changements de repères	87
e. Pourquoi ne pas utiliser une autre structure de données?	87
<b>3. Des relations étroites avec les objets gOcad existants</b>	<b>88</b>
3.1. Les lignes: édition de la géométrie	89
3.2. Les surfaces triangulées: utilisation des contraintes	90
3.2.1. Mécanisme général	90

a. Principe des associations . . . . .	90
b. Le mécanisme des contraintes . . . . .	91
3.2.2. Application directe à l'objet Faille: cas des points attracteurs . . . . .	93
a. Opérations de tirs . . . . .	93
b. Elément cible: la GFaultCell . . . . .	93
c. Construction de l' arbre octant (Octree) . . . . .	94
d. Résultats obtenus . . . . .	95
3.2.3. L'association GFaultBundle - TSurf . . . . .	96
a. Principe . . . . .	96
b. Exemple: interaction avec des points attracteurs (fuzzy control points) . . . . .	96
c. Contraintes disponibles . . . . .	97
<b>4. Conclusion sur la paramétrisation . . . . .</b>	<b>98</b>
<b>Chapitre 5 Exemples d'utilisation de cet objet sur des données réelles . . . . .</b>	<b>99</b>
<b>1. Méthode de construction sur un cas simple. . . . .</b>	<b>99</b>
1.1. Principe général de la construction . . . . .	100
1.1.1. Description des données . . . . .	100
1.1.2. Méthode de construction . . . . .	100
1.2. Projection des polygones de failles à la bonne altitude . . . . .	101
1.2.1. Méthode de "l'emporte-pièce" . . . . .	101
1.2.2. Méthode de projection . . . . .	102
1.2.3. Construction des polygones à partir des horizons. . . . .	103
1.3. Construction des lignes médianes . . . . .	104
1.3.1. Corrélation des lignes grâce à l'emporte pièce . . . . .	104
1.3.2. Corrélation des lignes à la main . . . . .	105
1.3.3. Construction des lignes médianes . . . . .	106
1.3.4. Cas particulier de l'horizon supérieur . . . . .	107
1.4. Construction de la faille . . . . .	107
1.4.1. Faille initiale . . . . .	107
1.4.2. Edition rapide de la faille brute: rééchantillonnage et extension . . . . .	108
1.4.3. Ajustement aux données . . . . .	109
<b>2. Méthode de construction sur des jeux de données plus complexes . . . . .</b>	<b>110</b>
2.1. Edition à partir de courbes interprétées . . . . .	110
2.2. Edition "à la main" . . . . .	112
2.3. Connections avec d'autres surface . . . . .	113
<b>3. Conclusion sur l'objet Faille . . . . .</b>	<b>114</b>
<b>Chapitre 6 Estimation des incertitudes et Méthodes de simulation . . . . .</b>	<b>115</b>
<b>1. Estimation des incertitudes . . . . .</b>	<b>117</b>
1.1. Le problème . . . . .	117
1.2. Estimation de l'incertitude à l'aide de formules simples . . . . .	118
1.3. Solution idéale: estimations d'incertitudes fournies par le géophysicien . . . . .	120

<b>2. Comment tenir compte des estimations de l'incertitude</b> .....	<b>121</b>
2.1. Principe .....	121
2.2. L'association FaultUncertainty .....	122
2.2.1. Stockage des estimations .....	122
2.2.2. Mécanisme de mise à jour .....	124
a. Description sommaire de la méthode choisie .....	124
b. Pourquoi ne pas faire une "vraie" extrapolation? .....	126
c. Mise au point d'un mécanisme de contrôle de la cohérence des estimations. . . . .	126
2.3. Exemple d'utilisation de ces mécanismes sur des données réelles .....	127
2.3.1. Différentes méthodes pour ajouter des vecteurs d'incertitude .....	127
2.3.2. Résultats obtenus après interpolation .....	129
<b>3. Méthodes de simulations</b> .....	<b>129</b>
3.1. Principe général .....	129
3.1.1. Quelques notions de géostatistique .....	130
a. Bref historique .....	130
b. Vocabulaire .....	131
c. Notion de variogramme .....	133
3.1.2. Utilisation de ces notions pour les simulations .....	134
a. Problèmes à éviter pour respecter la géométrie des failles. ....	134
b. Solution choisie: l'utilisation des champs de probabilités .....	135
c. Présentation des fonctions aléatoires utilisées .....	136
3.2. Méthodes simples .....	137
3.2.1. Fonction aléatoire constante = décalage latéral .....	137
3.2.2. Fonction linéaire verticalement et constante horizontalement = modification du pendage .....	138
3.2.3. Influence du type de loi	139
3.3. Méthodes plus complexes .....	140
3.3.1. Fonction constante verticalement et non linéaire horizontalement = modification de la géométrie en carte .....	140
a. Principe (très simplifié) .....	140
b. Influence de la portée sur la sinuosité .....	141
3.3.2. Fonction non linéaire à la fois verticalement et horizontalement .....	142
a. Principe de la méthode testée .....	142
b. Critique des résultats obtenus .....	143
3.4. Combinaison des trois méthodes de base .....	144
3.4.1. Pourquoi combiner les trois méthodes précédentes? .....	144
3.4.2. Principe général .....	144
a. Description de l'interface .....	145
b. Déroulement global des simulations .....	145
3.4.3. Cas où tous les vecteurs d'incertitude sont égaux .....	146
a. Code = 1: décalage latéral .....	146
b. Code = 2: changement de pendage .....	146
c. Code = 3: changement de pendage + décalage latéral .....	147
d. Code = 4: modification de la géométrie en carte uniquement .....	147
e. Code = 5: modification en carte + décalage latéral .....	147
f. Code = 6: modification en carte + changement de pendage .....	148
g. Code = 7: modification horizontale + décalage latéral + changement de pendage	149

3.4.4. Limites de validité de la méthode et cas «pathologiques» . . . . .	150
a. Premier cas «pathologique» . . . . .	150
b. Deuxième cas «pathologique» . . . . .	152
c. Troisième cas «pathologique» . . . . .	153
3.4.5. Conclusion sur les méthodes de simulation . . . . .	154

## **PARTIE 3: Etude au niveau du réservoir**

### **Chapitre 7 Construction de modèles volumiques 157**

<b>1. Modélisation du sous-sol: méthodes couramment utilisées . . . . .</b>	<b>157</b>
1.1. Différents types de modèles utilisés pour représenter le sous-sol . . . . .	157
1.1.1. But de la modélisation . . . . .	157
1.1.2. Modèles en deux dimensions . . . . .	158
1.1.3. Modèles en trois dimensions . . . . .	159
a. Modèles surfaciques . . . . .	159
b. Modèles volumiques continus (ou solides) . . . . .	160
c. Modèles volumiques discrets . . . . .	162
1.2. Méthode de construction classique de solides . . . . .	163
1.2.1. Principe . . . . .	163
a. Conditions à respecter. . . . .	163
b. Méthode de construction . . . . .	164
1.2.2. Problèmes de la méthode initiale . . . . .	164
a. Problèmes dus à la phase d'extension . . . . .	164
b. Problème rencontrés lors du calcul de la ligne d'intersection . . . . .	165
c. Problèmes de maillage . . . . .	166
1.2.3. Améliorations apportées . . . . .	166
a. Découpage sous contrainte . . . . .	166
b. Elimination des trous . . . . .	167
c. Amélioration du maillage . . . . .	168
1.3. Méthode de construction à partir des coupes sériées . . . . .	169
1.3.1. Construction de surfaces à partir de courbes frontières . . . . .	169
1.3.2. Principe de la construction . . . . .	170
1.3.3. Bilan de cette méthode . . . . .	172
a. Avantage majeur: le contact parfait est assuré par le pré-découpage . . . . .	172
b. Inconvénient: choix d'une direction privilégiée . . . . .	173
c. Idée: utiliser la même technique à partir des horizons . . . . .	173
<b>2. Connexions entre failles lors des simulations sur un modèle simple . . . . .</b>	<b>173</b>
2.1. Principe de la contrainte "Fault On Fault" (FOF) et "Horizon On Fault" (HOF) . . . . .	173
2.2. Gestion des contraintes lors des simulations . . . . .	175
2.2.1. Schéma général d'une simulation . . . . .	176
2.2.2. Transfert des contraintes initiales . . . . .	176
2.2.3. Description plus précise d'une simulation . . . . .	177
2.3. Résultats . . . . .	178

<b>Chapitre 8 Le Structural Model</b>	<b>181</b>
<b>1. Gérer différentes représentations pour un même objet géologique</b>	<b>182</b>
1.1. Les différents composants du modèle structural	182
1.1.1. Les failles	182
1.1.2. Les horizons	183
1.1.3. La boîte englobante	184
1.2. Séparation objet géologique / représentations de cet objet	184
1.2.1. la classe InterfaceModel et ses classes dérivées	184
1.2.2. la classe InterfaceRep et ses classes dérivées	185
1.3. Interface avec l'utilisateur	186
1.3.1. L'interaction se fait avec les objets géologiques, et non avec leur représentation	186
1.3.2. Gestion plus globale des objets géologiques	187
<b>2. Les relations entre les différents composants du modèle</b>	<b>188</b>
2.1. Différents types de contacts	188
2.1.1. Bord sur surface: contacts 2D	188
a. Le contact en T «simple»	189
b. Le contact multiple (Multi-contact)	190
c. Le contact multiple se terminant en T (Multi-T-Contact)	190
d. Les classes InterfaceBorder, InterfaceBorderRep et leurs dérivées	190
e. Diagramme récapitulatif pour les contacts à deux dimensions	191
2.1.2. Points triples: contacts 1D	192
2.2. Détection automatique des contacts	193
2.2.1. Découpage des bords d'horizons	193
2.2.2. Installation des liens correspondants	195
a. Pendant la phase de découpage	195
b. Après le découpage	196
c. Intervention de l'utilisateur	196
d. Contacts qui ne sont pas encore pris en compte automatiquement	197
<b>3. Construction de modèles volumiques à l'aide de surfaces triangulées «pré-découpés»</b>	<b>197</b>
3.1. Phase 1: création de toutes les surfaces triangulées et détection des contacts	198
a. A partir des horizons seuls	198
b. A partir des failles seules	198
c. A partir des deux types de données	199
3.2. Phase 2: construction des surfaces pré-découpées	199
3.2.1. Installation automatique des contraintes	199
a. A partir des Contacts 2D	200
b. A partir des Contacts 1D	200
c. Autres contraintes	200
3.2.2. Construction de surfaces de failles pré-découpées	201
a. Construction des différents groupes de lignes	201
b. Découpage au niveau des intersections	202
c. Construction des surfaces de faille «pré-découpées»	202

3.2.3. Construction du modèle volumique .....	203
a. Construction du modèle à l'aide des surfaces pré-découpées .....	203
b. Problèmes actuels posés par cette méthode .....	203
c. Solution envisagée à court terme .....	204
d. Solution envisagée à plus long terme .....	204
<b>4. Conclusion sur cette structure .....</b>	<b>205</b>
<b>Chapitre 9 Etudes volumétriques au niveau du réservoir: structures mises en jeu</b>	<b>207</b>
<b>1. Première phase: préparation du modèle .....</b>	<b>208</b>
1.1. Création du modèle initial .....	208
1.1.1. Données .....	208
1.1.2. Création du modèle de référence à l'aide du StructuralModel .....	208
a. Ajout des failles dans le modèle .....	208
b. Ajout des Horizons .....	209
c. Construction du modèle volumique (ou modèle 3D) .....	209
1.1.3. Autres manières de construire le modèle volumique .....	210
1.2. Caractérisation des régions du modèle de référence .....	210
1.2.1. Les problèmes à résoudre .....	210
a. Connaître les volumes extrêmes de chaque région (minimum et maximum) ...	210
b. Etre sûr d'associer toujours la même région avec un histogramme donné .....	211
1.2.2. Calcul des marqueurs de région .....	212
a. Isobarycentre .....	212
b. Barycentre pondéré à l'aide d'une tessellation .....	212
c. Comparaison des deux méthodes .....	212
d. Autre méthode (non implantée) .....	213
1.3. Analyse volumétrique rapide des régions .....	213
a. Calcul du volume de référence .....	213
b. Calcul rapide des volumes extrêmes .....	215
1.4. Stockage et visualisation des résultats .....	216
<b>2. Deuxième phase: boucle de simulations .....</b>	<b>218</b>
2.1. Déroulement d'une simulation .....	218
2.1.1. Simulation d'une nouvelle position pour chaque point de la faille .....	219
2.1.2. Mise à jour des surfaces .....	220
a. Principe général .....	220
b. Mise à jour sans utiliser le Structural Model .....	222
c. Mise à jour en utilisant le Structural Model .....	223
2.1.3. Reconstruction du modèle volumique .....	225
a. Avec l'objet StructuralModel .....	226
b. Stockage des résultats .....	228
<b>3. Conclusion sur ces structures .....</b>	<b>230</b>

## **PARTIE 4: Résultats et perspectives**

### **Chapitre 10 Lancement des simulations et visualisation des résultats 231**

<b>1. Description de l'interface</b>	<b>231</b>
1.1. Choix des paramètres	232
a. Fenêtre actuelle	232
b. Amélioration possible	232
1.2. Les boutons "Run", "Stop" et "Reset"	233
1.3. Sélection d'un certain nombre de modèles	233
<b>2. Résultats obtenus sur des modèles simples</b>	<b>234</b>
2.1. Exemple du modèle de Soultz	234
2.2. Exemple d'un modèle synthétique plus complexe	234
2.3. Comment tenir compte de la hiérarchie?	236
2.3.1. A court terme: classement de la liste des failles	236
2.3.2. A plus long terme: tenir compte de la hiérarchie dans les simulations	237
<b>3. Comment lier ces simulations à la construction de grilles faillées?</b>	<b>237</b>
3.1. Principe de la construction des grilles stratigraphiques faillées	237
a. Mise en correspondance des bords d'horizons	237
b. Paramétrisation des surfaces d'horizon	238
c. Définition de vecteurs guidant la construction suivant la troisième direction	239
d. Construction de la grille par extrusion	240
e. Limites de cette technique	241
f. Paramétrisation dans les trois dimensions de l'espace	242
3.2. Mise à jour après les simulations	243
3.2.1. Grille issue de la paramétrisation 3D	243
3.2.2. Grille issue de l'extrusion	243
3.2.3. Conclusion sur le passage aux grilles stratigraphiques faillées	243

### **Chapitre 11 Objectifs restants en structurale: organisation des réseaux de failles (hiérarchie et corrélation en 3 D) 245**

<b>1. Hiérarchisation</b>	<b>246</b>
1.1. Comment définir le hiérarchie du réseau de failles	246
1.1.1. Les systèmes de failles	246
1.1.2. La détermination structure Majeure/mineure dépend beaucoup de l'échelle d'observation	247
1.1.3. Cohérence entre les coupes et les cartes	248
1.2. Comment préserver cette hiérarchie lors des simulations	249
1.2.1. Stocker les familles de failles et leur ordre d'apparition	249
1.2.2. Simuler les failles en fonction de cette hiérarchie	250
1.2.3. Critique de la méthode	250
<b>2. Etude d'un autre type d'incertitudes: proposition de plusieurs scénarios</b>	<b>251</b>

2.1. le problème de la corrélation des segments de failles en 3D .....	251
2.2. Règles qui peuvent être utilisées pour vérifier la cohérence .....	252
2.2.1. Tenir compte des rejets .....	252
2.2.2. Utiliser les horizons .....	253
2.3. Comment tester plusieurs scénarios .....	253
2.3.1. Les cas de figures où cela pourrait être utile .....	253
2.3.2. Notion de Modèle Composite .....	254
2.3.3. Procédure proposée .....	254
2.3.4. Problème des connections entre failles .....	255
<b>3. Apport de l'étude des rejets .....</b>	<b>256</b>
3.1. Simplification pour une faille normale isolée .....	256
3.1.1. Un peu de théorie .....	256
3.1.2. Précautions à prendre lorsqu'il existe des failles secondaires .....	257
3.1.3. Technique employée sur un cas réel .....	258
3.1.4. Résultats obtenus .....	259
3.2. Amélioration de cette théorie dans le cas d'une tectonique polyphasée .....	260
3.2.1. Comparaison entre l'état stable et l'état après un séisme unique .....	260
3.2.2. Changements de connections au cours du temps .....	260
3.3. Modification de la théorie dans les zones d'interaction entre failles .....	262
3.3.1. Les zones de relais .....	262
3.3.2. Les zones de branchement .....	263
3.3.3. Résultats obtenus avec gOcad sur des données réelles .....	265
3.3.4. Amélioration possible de l'utilisation de cette technique dans gOcad .....	266
<b>Conclusion .....</b>	<b>269</b>
<b>1. Bilan de la thèse .....</b>	<b>269</b>
<b>2. Autres études d'incertitudes liées aux failles .....</b>	<b>272</b>
2.1. La description des terrains en contact de part et d'autre de la faille .....	272
2.1.1. La techniques des Allan Map .....	272
2.1.2. Etude des incertitudes à ce niveau .....	273
2.2. L'entraînement de matériel à l'intérieur de la faille .....	274
2.3. La diagénèse .....	274
2.4. Le calcul de la transmissivité des failles .....	275
2.5. La modélisation de l'épaisseur des failles .....	275
<b>Bibliographie .....</b>	<b>277</b>

---

# Liste des Figures

## PARTIE 1: Présentation du sujet

### **Chapitre 1: Présentation plus précise du cadre de travail**

- [1-1] Alea3D permet de suivre le point d'impact d'un puits sur un horizon lorsque la géométrie de cet horizon est modifiée, suite à une modification du modèle de vitesses ou à une nouvelle interprétation. A chaque interprétation, le nouveau point d'impact est calculé[a]. Les points d'impacts ainsi calculés (X) peuvent ensuite être reportés sur l'horizon nominal (disques sur horizon de référence)[b]. Ceci permet de vérifier si un changement d'interprétation risque de faire passer le puits de l'autre côté de la faille 8
- [1-2] Principe des simulations emboîtées de JACTA. Pour l'instant, seuls les horizons, les faciès et les propriétés sont simulés aléatoirement. Ce travail de thèse permettra d'ajouter la simulation de la géométrie des failles au processus. ....9
- [1-3] Changement de topologie après nouvelle interprétation .....11
- [1-4] Quelques structures représentant les objets géologiques. ....12
- [1-5] Grilles régulières (Voxet) et grilles stratigraphiques (SGrid). Dans le voxet [a], un modèle de vitesse a été calculé. Dans la grille stratigraphique [b], une simulation gaussienne séquentielle (SGS) a été générée pour simuler la porosité. ....13
- [1-6] Quelques exemples de contraintes géométriques pouvant être prises en compte en même temps par D.S.I. a): Control Node: le nœud est fixé (peut s'appliquer sur tout un bord ou sur quelques points correspondant à des coupes par exemple); b) Fuzzy control Points: les points issus de la sismique attirent la surface triangulée; c) Border On Straight Line: les points du bord ne peuvent bouger que le long d'une droite; d) Border On Cylinder: les points du bord ne peuvent bouger que le long d'un plan; e) Border On Surface: le bord de l'horizon doit aller se fixer sur la surface de faille; f) Border On Border: le coin de l'horizon doit coulisser sur le bord de la faille .....15
- [1-7] Le modèle doit être partagé par les différentes équipes afin de réduire les incertitudes et le temps de modélisation (d'après[17 - Tipee]). ..... 20

### **Chapitre 2: Modélisation des failles: techniques disponibles**

- [2-1] Nomenclature servant à décrire la géométrie d'une faille (d'après [27 - Foucault]) 24
- [2-2] Observations aidant à déterminer le mouvement d'une faille (non exhaustif) (d'après [27 - Foucault]. ....25
- [2-3] Différents critères pour classer les failles (d'après [27 - Foucault] .....26
- [2-4] Les failles synthétiques (1 et 2) et antithétiques (3 et 4) d'une faille principale. 26
- [2-5] Les grands types de pièges associés aux failles normales (d'après [34 - Hardman]). [a]: piège dans le mur de la faille; [b]: piège dans le toit; [c]: piège du à la fermeture hermétiq ue de la faille; [d]: "smear-gouge" (d'après [144 - Yielding 3]). ....28
- [2-6] Vue du grand canyon du Colorado (image [18 - WWW 1]). .... 28

<b>[2-7]</b>	Dispositif utilisé pour les modèles analogiques[a] et résultats obtenus [b] (librement inspiré de [14 - Mc Clay 1]). . . . .	29
<b>[2-8]</b>	L'ordinateur ne doit pas remplacer le travail des structuralistes, mais les aider dans leur travail. . . . .	30
<b>[2-9]</b>	Détection des failles (zones sombres) à partir d'analyses de pendages (avec la permission de l'équipe stratimagic: [19 - WWW 2]) . . . . .	31
<b>[2-10]</b>	La construction d'un modèle en trois dimensions permet de mieux appréhender la structure, et les contacts entre surfaces (Photo J-C. Dulac). . . . .	31
<b>[2-11]</b>	Les différents types de Horst et Grabens . . . . .	32
<b>[2-12]</b>	Caractéristiques géométriques des blocs basculés. . . . .	32
<b>[2-13]</b>	La largeur des blocs basculés est liée au taux d'extension . . . . .	33
<b>[2-14]</b>	La géométrie en carte des blocs basculés n'est pas toujours rectiligne [A] et peut être beaucoup plus complexe. [B]: relais entre failles; [C]: terminaison en échelon; [D]: tracé en zig-zag. . . . .	33
<b>[2-15]</b>	L'angle de l'extension par rapport aux structures héritées détermine si le mouvement se fera en extension pure, oblique ou en décrochement. . . . .	34
<b>[2-16]</b>	Relation entre l'angles des failles et obliquité a (angle entre l'extension et les structures héritées). [a]: failles normales pures; [b]: failles sigmoïdes obliques; [c]: failles décrochantes. . . . .	35
<b>[2-17]</b>	Les blocs synthétiques sont des sous-systèmes d'autres systèmes extensifs d'échelle supérieure. . . . .	35
<b>[2-18]</b>	[a]: Forme caractéristique des failles listriques (cuillère). [b] et [c]: caractéristiques en coupe: On observe généralement un fort rabattement des couches (cercles: "Roll-Over"). . . . .	36
<b>[2-19]</b>	Caractéristiques des failles listriques en carte: une forme courbe. . . . .	36
<b>[2-20]</b>	Origines mécaniques des failles listriques. [a]: variation progressive du régime des déformations en profondeur (DP et DT); [b]: compaction progressive des sédiments lors de l'enfouissement. . . . .	37
<b>[2-21]</b>	Origine cinématique des failles de croissance. . . . .	37
<b>[2-22]</b>	Deux figures typiques des nappes de charriage: les Klippes et les fenêtres tectoniques. . . . .	38
<b>[2-23]</b>	La géométrie des failles inverses est fortement liée à la lithologie (exemple de Duplex) . . . . .	39
<b>[2-24]</b>	Exemple de step faults (d'après [35 - Jones P.]). . . . .	39
<b>[2-25]</b>	Réseau de failles inverses. Les failles sont numérotées dans l'ordre de leur apparition (d'après[30 - Gidon]). . . . .	40
<b>[2-26]</b>	Importance de la lithologie dans la formation des pièges ([35 - Jones P.] . . . . .	40
<b>[2-27]</b>	[a]: Bassin en décrochement avec [b]: relais en transpression et [c]: relais en transtension ("pull-apart") avec des grabens obliques à la direction d'extension. . . . .	41

[2-28]	La géométrie des décrochements en 3D peut être très complexe (structures en fleur). [a]: relais décrochant compressif (fleur négative); [b]: relais décrochant distensif (fleur positive). . . . .	41
[2-29]	[a]: Forme hélicoïdale fréquemment rencontrée au niveau des failles; [b]: lanière découpée le long d'un décrochement. . . . .	42
[2-30]	Directions des contraintes principales pour les 3 styles de faille. . . . .	42
[2-31]	Quelques exemples de combinaisons de 2 styles tectoniques (des combinaisons plus complexes sont possibles). . . . .	43
[2-32]	La sédimentation joue un rôle lors de tectoniques polyphasées. [1]: les structures héritées ont une forte influence lorsque la sédimentation sur les structures réactivées est faible; [2]: l'influence est plus diffuse si l'épaisseur des sédiments augmente ou si les roches sont compétentes; [3]: cette influence disparaît au dessus d'un niveau de décollement. . . . .	43
[2-33]	Tenir compte de la forme des horizons pour déterminer la forme de la faille.	45
[2-34]	Principe de base pour cette méthode: la conservation de l'aire d'une section lors de la déformation. . . . .	45
[2-35]	Données utilisées dans la dernière version de la méthode . . . . .	46
[2-36]	“Chevron construction” de Verrall, 1981 . . . . .	47
[2-37]	Construction de Davison, 1986 . . . . .	48
[2-38]	Un cisaillement oblique avec une plus grande extension peut donner le même résultat qu'un cisaillement vertical . . . . .	49
[2-39]	L'angle de cisaillement a une grande importance . . . . .	49
[2-40]	Changement de repère utilisé dans la méthode de White.. . . .	50
[2-41]	Variables qui doivent être déterminées pour cette méthode . . . . .	51
[2-42]	L'utilisation de plusieurs horizons permet d'affiner les résultats. . . . .	51
[2-43]	Boîte de dialogue envisageable pour construire automatiquement la forme de la faille en coupe. . . . .	53

### **Chapitre 3: Bilan des incertitudes sur les failles**

[3-1]	Principe de l'acquisition des données sismiques (d'après Henry). Des ébranlements sont provoqués en surface à l'aide d'un émetteur (E) [a] et de nombreux récepteurs (ri) placés en surface recueillent les réponses [b].	56
[3-2]	Deux cas où la migration est indispensable: [a] synclinal et [b] point de diffraction. Le principe de la migration est montré en [c] (d'après [65 - Henry]). . . . .	57
[3-3]	Carte d'amplitudes utilisée pour repérer les zones où l'interprétation sera difficile: [a] problèmes de traitement qui donnent une amplitude beaucoup trop forte par rapport au reste de la zone; [b] plate-forme pétrolière déjà en place et qui empêche l'enregistrement de données à ce niveau; [c] superposition des données de cette campagne avec celles d'une ancienne campagne qui donne des amplitudes beaucoup trop fortes..	58
[3-4]	Les multiples augmentent les incertitudes au niveau des contacts horizon-faille. .	58

- 
- [3-5] Les différents types de sections qui sont interprétées. . . . . 59
- [3-6] Il faut bien sélectionner les coupes sismiques que l'on interprète en fonction de l'orientation des éléments structuraux présents dans la zone [a]. [b] les coupes perpendiculaires aux failles donnent plus d'informations que les coupes parallèles aux accidents [c]. . . . . 59
- [3-7] Vérification de la cohérence sur une coupe perpendiculaire à la faille. . . . . 60
- [3-8] Vérification de la cohérence sur une coupe parallèle à la faille. D'après les traces de failles repérées sur les coupes perpendiculaires à celle-ci (dessinées en pointillés), il vaudrait mieux remonter la trace de la faille sur cette section. . . . . 60
- [3-9] Création d'une carte "temps". Les différentes familles de failles ainsi que les zones de mauvaise qualité sismique sont dessinées à partir des sections interprétées. . . . . 61
- [3-10] Lorsque l'amplitude est trop forte [b], les failles, visibles en [a], peuvent être masquées. . . . . 63
- [3-11] [a] Sans la migration, les points diffractants posent de gros problèmes; Une migration ratée ne fait pas disparaître complètement cette parabole et entraîne des incertitudes importantes pour l'interprétation des failles lorsque les horizons se chevauchent [b] et [c]. . . . . 63
- [3-12] Incertitudes liées à la présence de failles couchées. [a] les ondes se réfléchissent comme sur les horizons et créent une zone sourde où l'interprétation est très difficile [b]. Ce phénomène est encore amplifié lorsque des hydrocarbures sont situés d'un côté de la faille (forte amplitude --> forte réflexion). . . . . 65
- [3-13] Incertitudes liées à un changement de panneau (d'après [70 - Thore 1]). . . . . 65
- [3-14] La lithologie joue un grand rôle dans la qualité de la sismique. [a]: les argiles sont de mauvais réflecteurs; [b]: lorsqu'il y a une alternance argile/sable, l'image sismique est meilleure. . . . . 66
- [3-15] La présence d'hydrocarbures peut détériorer la qualité de la sismique. . . . . 66
- [3-16] Bilan incertitudes au cours de la chaîne de traitement (d'après [70 - Thore 1]) . . . 67
- [3-17] Le projet OpenSpirit permettra de partager le modèle entre différentes applications et différents métiers (d'après [74 - WWW 3]). . . . . 68
- [3-18] Les failles peuvent jouer un rôle prépondérant dans le calcul des réserves. En effet, même une grande faille peut correspondre à un couloir de passage pour les fluides [a]; au contraire, une faille à faible rejet et donc difficilement repérable sur la sismique peut constituer une barrière [b]. Ce type de failles est repérable lorsque les pressions dans 2 puits voisins sont très différentes. . . . . 69
- [3-19] Calcul d'une surface d'horizon "plausible" compte tenu d'incertitudes estimées au préalable et connaissant la surface de référence (la "plus probable"). . . . . 70
- [3-20] Pré-traitement des vecteurs d'incertitude pour tenir compte des puits (d'après [69 - Shuguang]). . . . . 70
- [3-21] Les simulations acceptables pour les failles [b] sont moins nombreuses que pour les horizons [a]. Nous avons décidé de les éliminer lorsque la forme est trop irrégulière [c] ou que le pendage s'inverse par rapport à la faille de référence [d]. . . . . 71

## **PARTIE 2: Eude au niveau de la faille**

### **Chapitre 4: Paramétrisation de l'Objet Faille**

<b>[4-1]</b>	Les bords des horizons interprétés ne correspondent pas forcément aux failles. . . . .	74
<b>[4-2]</b>	Données disponibles pour construire les failles . . . . .	75
<b>[4-3]</b>	Différentes courbes disponibles pour modéliser les failles . . . . .	75
<b>[4-4]</b>	La paramétrisation de l'objet faille est fortement inspirée de l'objet GShape décrit par Liliane Wietzerbin [85 - Wietzerbin ]. . . . .	76
<b>[4-5]</b>	Séparation de la géométrie et de la topologie (Un Node contient un Vertex mais ne dérive pas de la classe Vertex). . . . .	77
<b>[4-6]</b>	[a]: courbe valide; [b]: topologies non valides (plus de 2 deux voisins pour certains nœuds) . . . . .	78
<b>[4-7]</b>	Quelques exemples d'Atomic / AtomicGroup. . . . .	78
<b>[4-8]</b>	Gestion des indices pour les propriétés . . . . .	79
<b>[4-9]</b>	Trois fonctionnalités beaucoup utilisées par l'objet faille . . . . .	79
<b>[4-10]</b>	Les trois façons de créer une faille (à partir de courbes, de surfaces, ou étape par étape) . . . . .	81
<b>[4-11]</b>	La seule question à se poser pour ajouter une génératrice: la courbe tombe-t-elle près d'un nœud [1], à l'intérieur d'un segment existant [2] ou en dehors de la courbe principale [3]? . . . . .	82
<b>[4-12]</b>	Mise à jour automatique lors d'une densification de la courbe principale. . . . .	82
<b>[4-13]</b>	Stockage des génératrices sous forme de propriétés . . . . .	83
<b>[4-14]</b>	Densifier les génératrices permet d'améliorer la précision de la géométrie. [a]: 4 points par génératrice; [b]: 20 points par génératrice. . . . .	84
<b>[4-15]</b>	calcul de la base locale pour chaque atome de la courbe principale . . . . .	84
<b>[4-16]</b>	Il faut orienter les génératrices de manière cohérente pour éviter les croisements. [a]: une génératrice mal orientée crée des problèmes lors de l'interpolation; [b] et [c]: les génératrices sont bien orientées (courbe principale sub-horizontale en b et sub-verticale en c). . . . .	85
<b>[4-17]</b>	Stockage des coordonnées dans une base locale . . . . .	85
<b>[4-18]</b>	Relations entre l'objet faille et les autres objets gOcad. . . . .	88
<b>[4-19]</b>	Edition de la géométrie d'une génératrice. [a]: faille initiale; [b] transformation en ligne polygonale; [c]: modification de la ligne; [d]: réinjection de la nouvelle courbe dans la faille initiale; [e]: certaines génératrices sont fixées (Control Generators); [f]: résultat après interpolation. . . . .	89
<b>[4-20]</b>	Principe de l'Association (modifié d'après [4 - Cognot 1]). . . . .	90
<b>[4-21]</b>	Structure de base pour le stockage des contraintes (d'après [4 - Cognot 1]). Dans le schéma, GObj représente un objet géologique (GOBJ = Geological Object) . . . . .	91

[4-22]	Quelques modèles Atomique-Contrôleur dans gOcad, . . . . .	92
[4-23]	Principe du tir: trouver le point d'impact le plus proche dans une direction donnée et stocker l'élément cible correspondant. . . . .	93
[4-24]	Numérotation des cellules dans le conteneur . . . . .	93
[4-25]	Construction de l'arbre octant par découpages successifs des cellules: un découpage intelligent de l'espace à trois dimensions (vue de dessus). . . . .	94
[4-26]	L'octree permet de préselectionner la boîte (a) contenant les GFaultCells (b) parmi lesquelles le point d'impact doit être recherché (c). . . . .	95
[4-27]	La définition des contraintes directement sur l'objet faille est impossible du fait de l'asymétrie de la paramétrisation. . . . .	95
[4-28]	L'installation de contrainte sur l'objet faille provoque l'installation automatique de la contrainte correspondante sur sa surface interne. . . . .	97
[4-29]	La contrainte "FaultOnFault" met en relation six associations . . . . .	97

### Chapitre 5: Exemples d'utilisation de cet objet sur des données réelles

[5-1]	Modèle simple composé de trois horizons, d'une faille principale, d'une faille synthétique se branchant sur cette faille principale et de deux failles isolées (f3 et f4) . . .	99
[5-2]	Données disponibles pour le modèle simple: 3 ensembles de points migrés profondeur . . . . .	100
[5-3]	Construire les lignes médianes à partir des bords des horizons évite les effets d'escalier (données fournies par EXXON). . . . .	100
[5-4]	Méthode de "l'emporte-pièce". [a]: surface continue; [b]: translation du polygone de faille de part et d'autre de l'horizon; [c]: construction de l'emporte pièce; [d]: découpage de la surface continue; [e]: surface finale. . . . .	102
[5-5]	Projection du polygone sur l'ensemble de points. [a]: données initiales; [b]: plan de recherche des proximités; [c]: recherche des correspondances entre les points du polygone et ceux de l'horizon sur le plan horizontal; [d]: affectation des bonnes coordonnées Z et polygone de faille projeté à la bonne profondeur. . . . .	102
[5-6]	Construction des lignes médianes à partir des horizons uniquement, en utilisant les méthodes de morphologie mathématique. [a]: ensemble de points initial; [b] extraction des points du bord; [c]: construction automatique de lignes polygonales; [d]: Edition des lignes dans les zones délicates; [e]: construction automatique des lignes médianes. . . . .	103
[5-7]	Installation de marqueurs afin de délimiter les bords externes des failles, ainsi que les zones de connexion entre failles. . . . .	104
[5-8]	Corrélation des bords par tirs successifs. [a]: découpage des courbes au niveau des fourches; [b]: opération de tir; [c]: classification des bords en fonction de la faille à laquelle ils appartiennent. . . . .	105
[5-9]	préparation des courbes avant la construction de la ligne médiane. . . . .	106
[5-10]	Il est possible de construire la ligne "médiane" plus proche de l'une des deux lignes,	

	lorsque l'utilisateur le demande. ....	106
[5-11]	Calcul de la courbure pour l'horizon supérieur et sélection de la courbe correspondant à la trace de la faille. ....	107
[5-12]	Construction de l'objet faille à partir de lignes horizontales: il suffit de cliquer successivement sur les lignes médianes après avoir donné le nom de la faille et le nombre de points par génératrices. ....	107
[5-13]	Extension de la faille dans ses deux directions principales ....	108
[5-14]	Fonctions d'édition simple sur la faille initiale [a]; [b]: rééchantillonnage dans les deux directions principales de la faille. ....	108
[5-15]	Ajustement de la géométrie de la faille à de nouvelles données en utilisant la contrainte "FCP" ....	109
[5-16]	Différents éléments permettant de passer automatiquement d'une faille partielle construite uniquement à partir des bords d'horizons, à une faille complète prenant en compte toutes les données disponibles. ....	110
[5-17]	[a]: failles construites à partir des bords d'horizons; [b]: les interprétations de failles donnent une information sur l'extension verticale des failles; [c]: réseau de faille obtenu par le processus automatique; [d]: modèle validé par le géologue structuraliste. ....	111
[5-18]	Bien que la faille obtenue de manière automatique corresponde aux données, les connaissances du géologues sont toujours nécessaires afin de construire un réseau cohérent. ....	111
[5-19]	Lorsque l'interprétation de l'horizon n'a pas été poussé assez loin, il faut une information supplémentaire pour modéliser la faille. ....	112
[5-20]	Les données présentent des incohérences: les interprétations de failles traversent les horizons [b]. A quelles données doit-on faire confiance? ....	112
[5-21]	L'édition des courbes permet de diminuer les incohérences ou de pallier le manque d'informations. ....	113
[5-22]	Extension de l'objet faille vers une faille ou un horizon. ....	113
[5-23]	Exemple d'extension lorsque l'horizon n'est pas plan. ....	114

## **Chapitre 6: Estimation des incertitudes et Méthodes de simulation**

[6-1]	Deux cas où une mauvaise interprétation peut avoir de graves conséquences lors de la production: [a]: mauvaise interprétation du pendage; [b]: mauvaise interprétation de la position réelle de la faille. ....	115
[6-2]	Parmi toutes les incertitudes affectant les failles, nous n'avons pas pris en compte: [a]: les incertitudes dues aux vitesses de migration; [b]: les incertitudes dues à une interprétation incohérente d'un horizon à l'autre; [c]: les incertitudes dues au choix du scénario pour le réseau de failles [121 - Freeman]. ....	116
[6-3]	La confiance à accorder au polygone de faille doit être modulée en fonction de la corrélation avec les données sismiques. Lorsque les deux interprétations sont concordantes, l'incertitudes est faible, mais s'il manque des données sismiques, il est beaucoup	

---

	plus difficile de prendre une décision. . . . .	117
[6-4]	Lorsqu'il existe des zones de recouvrement entre deux horizons successifs, le choix de la forme de la faille est laissé à celui qui la construit. Bien que ce choix s'effectue en fonction de règles structurales, une forte incertitude peut persister. . . . .	118
[6-5]	Calcul des propriétés utilisés pour construire les vecteurs d'incertitude . . . . .	119
[6-6]	Les deux méthodes empiriques pour calculer les vecteurs d'incertitude. . . . .	119
[6-7]	Lorsque l'incertitude est due à des zones de recouvrement, son estimation est plus difficile, surtout si le recouvrement se produit entre deux horizons différents. . . . .	120
[6-8]	Exemple de carte donnant une estimation de la qualité des données sismiques (fournie de temps en temps par l'interpréteur). . . . .	121
[6-9]	Conventions pour décrire les diagrammes de classes . . . . .	122
[6-10]	Éléments stockés dans une <code>GFaultUncertaintyValue</code> . . . . .	122
[6-11]	Pourquoi les incertitudes sont stockées dans la base locale. . . . .	123
[6-12]	Classes intervenant dans la gestion des vecteurs d'incertitude. . . . .	123
[6-13]	Les trois grandes étapes intervenant dans la mise à jour des vecteurs d'incertitude: [a]: interpolation pour les génératrices portant des données; [b]: affectation des valeurs ainsi interpolées aux génératrices extrêmes; [c]: interpolation linéaire pour affecter l'incertitude au niveau des derniers points. . . . .	125
[6-14]	Pourquoi la méthode de mise à jour n'utilise pas une vraie approximation. [a]: méthode utilisée; [b]: extrapolation dans un cas où tout se passe bien; [c]: extrapolation avec inversion des directions. . . . .	126
[6-15]	Vérification de la cohérence des estimations d'incertitude . . . . .	126
[6-16]	Avant d'être transférées sur l'objet faille, les incertitudes peuvent être stockées sur une ligne polygonale. . . . .	127
[6-17]	Les deux méthodes permettant d'ajouter des estimations de l'incertitude sur l'objet faille. La donnée de vecteurs sera utilisé pour donner une incertitude globale à la faille, la méthode scalaire sera plutôt utilisée pour faire varier l'incertitude le long de la faille. . . . .	128
[6-18]	Les quelques vecteurs d'incertitude estimés [a] sont interpolés le long de la faille [b] afin de définir le volume dans lequel la géométrie de la faille peut être simulée. .	129
[6-19]	Principe très simplifié de l'utilisation du cokrigeage colocalisé pour estimer les faciès à partir de données de puits et de données sismiques. . . . .	130
[6-20]	Quelques fonctions de densité de probabilité couramment utilisées dans les modèles, et courbes cumulatives correspondantes. . . . .	132
[6-21]	Relation entre le variogramme et la fonction de corrélation. . . . .	133
[6-22]	Comment construire un variogramme (par soucis de simplicité la grille est représentée en deux dimensions uniquement, mais la méthode est transposable en trois dimensions). . . . .	133
[6-23]	Simulation qui semble inadmissible pour représenter une faille. . . . .	134

<b>[6-24]</b>	Un type de résultat acceptable si la lithologie correspond, mais rejeté par défaut dans les méthodes de simulations proposées. ....	134
<b>[6-25]</b>	Principe général des simulations: à chaque point de la faille une fonction de densité de probabilité triangulaire (ou uniforme) est associée. Une réalisation de la variable aléatoire est ensuite utilisée pour calculer une nouvelle position à partir de la cdf associée à ce point. ....	135
<b>[6-26]</b>	Différentes fonctions aléatoires qui peuvent être proposées (seules les cadres en gras ont été testés pour le moment). ....	136
<b>[6-27]</b>	Exemple de la méthode de simulation la plus simple: la fonction aléatoire est constante sur toute la faille. Le même nombre aléatoire $p$ est utilisé pour inverser les différentes courbes cumulatives locales. ....	137
<b>[6-28]</b>	Exemple de simulation permettant une variation de pendage. Dans ce cas, la fonction aléatoire correspond à une série de droites. ....	138
<b>[6-29]</b>	A condition de faire assez de simulations, on vérifie bien que lorsqu'une loi uniforme est choisie (au centre), les simulations sont régulièrement réparties, alors que pour une loi triangulaire (en bas), les simulations sont concentrées autour de la position de référence. 139	
<b>[6-30]</b>	Quelques résultats obtenus avec la méthode décrite plus haut. ....	140
<b>[6-31]</b>	Influence de la portée sur la forme de la courbe obtenue ....	141
<b>[6-32]</b>	Calcul de la propriété utilisée pour guider la simulation. ....	142
<b>[6-33]</b>	Avec un variogramme à grande portée (en bas), le résultat est meilleur, mais la fonction aléatoire n'est toujours pas garantie monotone suivant l'axe Z. ....	143
<b>[6-34]</b>	Boîte de dialogue permettant de choisir les différents paramètres de simulation pour une faille donnée: le type de fonction aléatoire, le type de loi à inverser localement, autoriser ou non l'inversion de pendage et le choix de la portée pour la modification de la géométrie horizontale. ....	145
<b>[6-35]</b>	Fonctions aléatoires préservant la forme dans le plan horizontal ....	147
<b>[6-36]</b>	Transformation du tableau de nombres aléatoires de l'intervalle $[0,1]$ vers un intervalle plus approprié. L'image du point $p_1$ dans l'espace réel est noté: $F^{-1}(p_1) = b$ . ...	148
<b>[6-37]</b>	Sélection des intervalles en fonction du type de simulation sélectionné ....	150
<b>[6-38]</b>	Problème de cassure lorsque le rapport ugauche/udroite est différent de 1 ....	151
<b>[6-39]</b>	Modification à apporter pour des lois triangulaires asymétriques sur une même verticale. 151	
<b>[6-40]</b>	Solution rapide pour tenir compte du cas ugauche/udroite 1 ....	151
<b>[6-41]</b>	Problème lorsque les vecteurs d'incertitude varient selon l'axe horizontal de la faille. 152	
<b>[6-42]</b>	Sélection de la zone valide dans le cas de vecteurs asymétriques. ....	152
<b>[6-43]</b>	Cas conduisant à une forme on géologique (très exagéré). ....	153
<b>[6-44]</b>	Sélection de la partie valide des vecteurs d'incertitude, en fonction du type de simulation demandé. ....	153

- [6-45] Les méthodes de simulations actuelles permettent d’explorer tout le volume [a]. Il est possible d’augmenter le pendage par rapport à la faille de référence [b]. La condition “fonction monotone suivant l’axe Z” est respectée et évite d’obtenir des inversions de courbure [c]. Par contre, les diminutions de pendage sont encore limitées à cause de conditions trop restrictives. . . . . 154

## **PARTIE 3: Etude au niveau du réservoir**

### **Chapitre 7: Construction de modèles volumiques**

- [7-1] Apport de la modélisation en trois dimensions . . . . . 158
- [7-2] Différents types de représentations pour les surfaces (d’après [109 - Orlic]). . . . . 159
- [7-3] Les différents composants du type de modèle volumique utilisé dans gOcad: le modèle de Weiler. . . . . 160
- [7-4] Deux cas où la construction d’un modèle volumique continu permet de détecter des incohérences: la fuite entre deux régions lorsqu’il existe une communication anormale [en haut]; la création d’une région supplémentaire lorsque deux surfaces s’intersectent anormalement [en bas]. . . . . 161
- [7-5] Exemples de modèles volumiques discrets (un zoom permet de vérifier ce qui se passe dans chacun des cas). . . . . 163
- [7-6] Les grandes étapes de la construction d’un modèle volumique continu. . . . . 164
- [7-7] Formation de “nœuds papillon” lors de l’extension de certains bords (d’après [106 - Euler 2]). . . . . 165
- [7-8] Exemple où l’intersection n’est pas détectée. Ceci peut être du à la forte courbure de la faille de droite, au fait que la faille de gauche n’a pas été étendue assez, ou à la différence de densité de maillage entre les deux surfaces. . . . . 165
- [7-9] Au fur et à mesure que les surfaces sont découpées, la qualité du maillage se dégrade. . . . . 166
- [7-10] Utilisation des contraintes installées pour calculer la ligne d’intersection: les points d’impacts trouvés par la méthode d’interpolation sont insérés dans la ligne de contact. . . . . 167
- [7-11] Problème de création de trou (d’après [106 - Euler 2]). . . . . 167
- [7-12] Elimination des vides par projection (d’après [106 - Euler 2]). . . . . 168
- [7-13] Amélioration de la qualité du maillage . . . . . 168
- [7-14] Une structure qui permet de construire des surfaces à partir de leurs lignes d’intersection avec d’autres interfaces du modèle. . . . . 169
- [7-15] Données utilisées: trois coupes géologiques parallèles. . . . . 170
- [7-16] Appariement des traces de failles d’une coupe à l’autre . . . . . 170
- [7-17] Squelettes de failles et squelettes d’horizons. . . . . 171
- [7-18] Surfaces incorporant les lignes de contact dans leur maillage . . . . . 171

[7-19]	Construction du modèle à partir des surfaces pré-découpées. . . . .	172
[7-20]	Difficile de détecter d'une terminaison de faille lorsqu'une seule direction est privilégiée. . . . .	173
[7-21]	L'utilisation de ces contraintes crée de nombreuses associations. . . . .	174
[7-22]	Respect des contacts entre l'horizon et la faille en utilisant la contrainte HorizonOn-Fault (HOF), par l'intermédiaire de la surface interne de l'objet faille . . . . .	174
[7-23]	Fermeture du réseau de failles en utilisant la contrainte Fault On Fault (FOF), par l'intermédiaire de la surface interne de l'objet faille. . . . .	175
[7-24]	Boîte de dialogue permettant à l'utilisateur de sélectionner le mode de simulation pour chacune des failles du modèle. . . . .	176
[7-25]	Exemple de transfert pour une contrainte "Fault On Fault". . . . .	177
[7-26]	Une simulation (modification puis mise à jour des contacts) . . . . .	177
[7-27]	Vérification du maintien des contacts sur deux simulations. . . . .	178
[7-28]	Lorsque les contacts sont soudés, il est possible de construire le modèle surfacique, et donc de calculer les volumes de chacune des régions. . . . .	178
[7-29]	Exemple de modèle (encore assez simple) sur lequel les copies de surfaces et de contraintes deviennent trop lourdes à gérer de cette manière. . . . .	179

## Chapitre 8: Le Structural Model

[8-1]	L'information concernant la connectivité entre les failles est perdue lorsqu'elles sont construites indépendamment, et sans tenir compte des horizons. . . . .	182
[8-2]	Les représentations possibles pour modéliser les failles . . . . .	183
[8-3]	Les représentations possibles pour modéliser les horizons . . . . .	183
[8-4]	Représentation des limites du domaine à modéliser. . . . .	184
[8-5]	Relations entre les classes InterfaceModel, HorizonModel, FaultModel et InterfaceModelRep (les conventions utilisées pour ce schéma et les suivants sont décrites dans la figure 6-9, page 122). . . . .	185
[8-6]	Les différentes représentations possibles pour les interfaces du modèle structural: points, lignes, surfaces triangulées... . . . .	185
[8-7]	Une petite partie seulement est visible par l'utilisateur. . . . .	186
[8-8]	Le contrôle de la visualisation des objets du modèle se fait de manière plus globale et plus ordonnée. Ici, il a été décidé de visualiser tous les horizons sous forme surface triangulée, avec le maillage visible, et toutes les failles sous forme de surface triangulée, avec le maillage visible. . . . .	187
[8-9]	Regroupement de cinq appels à une même commande en une commande globale (aucun algorithme nouveau, c'est juste une question d'interface). . . . .	188
[8-10]	Les différents types de contact «2D». . . . .	189
[8-11]	Structure d'un contact en T simple . . . . .	189

---

[8-12]	Correspondance entre les InterfaceRep et les InterfaceBorderRep . . . . .	191
[8-13]	Organisation des contacts «bord sur surface» (Contact 2D) . . . . .	191
[8-14]	Cas de figure correspondant au point triple de type «contact 1D». . . . .	192
[8-15]	Organisation des contacts en points triples (Contact 1D) . . . . .	192
[8-16]	Séparation des bords externes et des bords internes correspondant à des failles . .	193
[8-17]	Découpage des bords correspondant aux failles, et appariements des bords se correspondant. . . . .	194
[8-18]	Ajout des marqueurs correspondants sur la représentation surface de l’horizon. .	194
[8-19]	Classification des bords soit par appartenance aux horizons, soit par appartenance aux failles. . . . .	195
[8-20]	Installation automatique des liens «branchement entre failles» et «association des deux lèvres d’une faille». . . . .	195
[8-21]	Installation des liens de type «associations de lèvres de failles appartenant à deux horizons consécutifs». . . . .	196
[8-22]	Rassemblement de bords pour obtenir un résultat plus géologique. . . . .	196
[8-23]	Contacts qui ne sont pas détectés actuellement. . . . .	197
[8-24]	Schéma général de construction du modèle . . . . .	197
[8-25]	Phase 1 réalisée en partant d’horizons uniquement: détection des liens [a]; regroupement des bords de faille [b]; construction des objets faille étendus [c]; construction des surfaces de failles [d] (données fournies par la compagnie EXXON et traitées par Isabelle Duvinage). . . . .	198
[8-26]	En partant de failles uniquement (critère de distance). . . . .	198
[8-27]	En partant des deux types de données (cas idéal). . . . .	199
[8-28]	Installation des contraintes sur les bords à l’aide de l’information stockées dans les différents contacts (1D et 2D) . . . . .	200
[8-29]	Partage des points du contact entre les différents objets. . . . .	201
[8-30]	Découpage des lignes constituant le squelette de faille au niveau de leurs intersection. . . . .	202
[8-31]	Surfaces pré-découpées, partageant exactement les lignes d’intersection. . . . .	202
[8-32]	Modèle construit à partir des surfaces pré-découpées. . . . .	203
[8-33]	Formation d’un triangle plat après interpolation si trois points d’un triangle du bord sont situés sur le bord. . . . .	203
[8-34]	l’éminination des triangles plat n’est pas un problème trivial (problème repéré par Nicolas Euler durant sa thèse). . . . .	204

## **Chapitre 9: Etudes volumétriques au niveau du réservoir: structures mises en jeu**

[9-1]	Etat initial du modèle structural (seulement des failles). . . . .	208
[9-2]	Etat du modèle structural après l'ajout des horizons. . . . .	209
[9-3]	Construction du modèle à partir des surfaces pré-découpées . . . . .	210
[9-4]	Suivi des volumes des régions au cours des simulations . . . . .	211
[9-5]	Calcul d'un marqueur par région, restant dans la région quelle que soit le résultat de la simulation. . . . .	211
[9-6]	Calcul des marqueurs sur un modèle synthétique cubique. . . . .	212
[9-7]	Calcul de l'intégrale associée à un triangle quelconque de la région: décomposition du volume correspondant en deux sous-volumes. . . . .	214
[9-8]	Sélection des vecteurs d'incertitude correspondant à la région. . . . .	215
[9-9]	Calcul du volume minimum et du volume maximum de la région. Dans ce cas de figure simple, les variations de volumes ne sont dus qu'à une seule faille. . . . .	216
[9-10]	Rassemblement des informations calculées lors de la phase 1. . . . .	217
[9-11]	Schéma général de l'algorithme global de simulation . . . . .	218
[9-12]	Séparation de la simulation et de la mise à jour du modèle . . . . .	220
[9-13]	Diagramme montrant les relations entre les simulateurs de faille et les gestionnaires de simulations. Les fonctions de base qui diffèrent entre les deux classes dérivées sont signalées par une étoile. . . . .	221
[9-14]	Sans utiliser de STRUCTURALMODEL, la surface courante est une copie de la surface «cachée» dans l'objet faille. . . . .	222
[9-15]	Extension des bords afin éviter les problèmes numériques. . . . .	223
[9-16]	Interpolation afin de mettre à jour les points du bord. . . . .	223
[9-17]	Avec l'utilisation d'un STRUCTURALMODEL, la surface courante est une copie de la surface «cachée» dans l'objet faille étendue, mais elle ne sert que d'intermédiaire	225
[9-18]	L'architecture des simulateurs suit celle du StructuralModel. . . . .	226
[9-19]	Introduction de la notion de hiérarchie dans les simulations. . . . .	228
[9-20]	Stockage des paramètres après chaque simulation. . . . .	228
[9-21]	Stockage des volumes du modèle courant (après 3 simulations). . . . .	229
[9-22]	Diagramme récapitulatif (classes de stockage sur fond noir) . . . . .	229

## **PARTIE 4: Résultats et perspectives**

### **Chapitre 10: Lancement des simulations et visualisation des résultats**

[10-1]	Fenêtre permettant la visualisation des résultats. . . . .	232
[10-2]	Sélection du type de simulation en fonction de la faille. . . . .	233

[10-3]	Exemple de simulations sur un modèle plus conséquent. . . . .	234
[10-4]	Evolution des histogrammes au cours d'un cycle de simulations . . . . .	235
[10-5]	Association des bords d'horizon . . . . .	238
[10-6]	Le passage en domaine paramétrique permet de définir la grille plus facilement. . . . .	238
[10-7]	Différentes manières de définir les vecteurs dans la troisième direction: constant vertical [a] ou oblique [b], correspondances entre horizons[c]. . . . .	239
[10-8]	Exemple de construction automatique des vecteurs «guides» à partir des liens installés entre deux horizons. . . . .	239
[10-9]	Principe de l'extrusion verticale: empilement des cellules en utilisant la même abscisse curviligne le long des vecteurs guides. . . . .	240
[10-10]	Possibilité de projeter les cellules du bord sur les failles pour améliorer le contact (mais avec création de cellules dégénérées). . . . .	240
[10-11]	Quelques cas conduisant à la création de grilles non valides. . . . .	241
[10-12]	Construction de la grille en utilisant une paramétrisation 3D. . . . .	242

## **Chapitre 11: Objectifs restants en structurale: organisation des réseaux de failles**

[11-1]	Les différentes échelles de l'étude des incertitudes sur les failles. . . . .	245
[11-2]	Respect du schéma de faille initial pendant les simulations: les failles recoupées par une faille plus récente doivent être déplacées dans le même sens afin de respecter les distances initiales. . . . .	246
[11-3]	Terminologie des réseaux de failles (d'après [44 - Ramsay]). . . . .	247
[11-4]	A grande échelle les failles B et C sont des failles mineures de la faille A, tandis qu'à petite échelle, ce sont des failles majeures. . . . .	247
[11-5]	Relation entre les failles secondaires et les failles principales. [a]: en coupe, [b]: en carte. . . . .	248
[11-6]	La géométrie des failles est souvent la même en carte et en coupe. . . . .	248
[11-7]	Exemple montrant les trois familles de failles . . . . .	249
[11-8]	La structure de donnée "TectoEvent": toutes les failles recoupées par une même faille sont stockées. . . . .	249
[11-9]	Stockage de la hiérarchie du modèle de la [figure 11-7] . . . . .	250
[11-10]	la corrélation en trois dimensions des sections sismiques interprétées en 2 dimensions n'est pas aisée: différentes interprétations sont possibles du point de vue géologique et le géologue doit faire des choix (connaissance de la géologie régionale, études des rejets....) (D'après [121 - Freeman]) . . . . .	251
[11-11]	Utilisation des rejets pour vérifier la cohérence d'un groupe de failles. [a]: en extension; [b]: en compression. . . . .	252
[11-12]	Corrélation à l'aide des rejets: la solution [b] est très improbable. . . . .	252
[11-13]	[a]: la corrélation est possible; [b]: la corrélation est déconseillée . . . . .	253

<b>[11-14]</b>	Tester plusieurs hypothèses pour les connections entre failles. . . . .	253
<b>[11-15]</b>	Connecter plusieurs failles grâce au modèle “composite” (l’édition d’une des représentations a une action sur toutes les autres représentations). . . . .	254
<b>[11-16]</b>	Retrouver l’état initial de la zone à étudier . . . . .	254
<b>[11-17]</b>	Editer les liens dans la zone délicate . . . . .	254
<b>[11-18]</b>	Reconstruction de la partie de modèle éditée . . . . .	255
<b>[11-19]</b>	Méthodes actuellement utilisées pour déterminer la direction de connection entre une faille secondaire et sa faille principale. . . . .	255
<b>[11-20]</b>	Différentes vue d’une surface de faille “elliptique”. [a]: vue en coupe qui montre les couches entraînées par le mouvement de la faille; [b]: vue en carte; [c]: projection des rejets sur un plan vertical parallèle à la direction principale de la faille [119 - Barnett]. [d]: profil du rejet normalisé ([54 - Walsh 1]) . . . . .	256
<b>[11-21]</b>	[a] Il faut tenir compte de toutes les failles secondaires pour avoir un diagramme régulier. [b]: le diagramme de gauche tient compte de la faille b, au contraire du diagramme de droite; [c]: le diagramme de gauche tient compte de la faille Y, au contraire du diagramme de droite (d’après [129 - Walsh 4]). . . . .	257
<b>[11-22]</b>	Technique employée pour calculer les rejets (“strike projection”): [a] calcul du rejet pour un certain nombre de couples de points de part et d’autre de la faille; [b] projection des valeurs obtenues sur un plan; [c] interpolation et tracé de contours d’isovaleurs. . . . .	258
<b>[11-23]</b>	Calcul du rejet dans un cas simple. [a]: calcul des rejets des différentes failles le long de l’horizon supérieur; [b]: calcul des rejets pour chacune des failles le long de l’horizon inférieur; [c] le rejet des failles secondaires est incorporé dans le rejet total projeté sur le plan; [d] interpolation des valeurs de rejet ainsi obtenues; [e]: carte finale des rejets. . . . .	259
<b>[11-24]</b>	Evolution du diagramme des rejets au cours des séismes successifs (d’après [127 - Walsh 2]). . . . .	260
<b>[11-25]</b>	Différentes évolutions possibles d’une faille au cours du temps (d’après [125 - Morley]).	261
<b>[11-26]</b>	Différentes relations possibles entre les segments d’une même faille (d’après [130 - Willemse]). . . . .	262
<b>[11-27]</b>	Comparaison des rejets de failles isolées [a] et de failles en interaction mécanique [b] (d’après [130 - Willemse]). . . . .	262
<b>[11-28]</b>	Comparaison entre les caractéristiques d’une faille elliptique isolée (gauche) et celle d’une ensemble de trois segments de faille (droite) avec des zones de recouvrement (grisé), d’après [130 - Willemse]). . . . .	263
<b>[11-29]</b>	Influence du branchement sur les rejets (D’après [124 - Maerten 2]) . . . . .	263
<b>[11-30]</b>	L’étude des rejets peut aider à améliorer l’interprétation du réseau de failles (simplifié d’après [123 - Maerten 1]). . . . .	264
<b>[11-31]</b>	L’analyse des rejets permet de retrouver les principaux contacts de la [figure 11-30] (Modèle initial construit par Christophe Basire [64 - Basire]) . . . . .	265

[11-32] Lorsque l'on a uniquement la description de la faille au niveau des horizons, comment relier les nœuds des bords d'horizon correspondant aux deux lèvres de la failles? 266

### **Conclusion**

[C-1]	Principe de la construction des Allan Map. ....	273
[C-2]	Génération d'un ensemble d'Allan Map, en jouant sur le rejet. ....	273
[C-3]	Deux couches poreuses juxtaposées peuvent créer un piège ....	273
[C-4]	Différentes manières d'estimer l'entraînement des argiles dans l'épaisseur de la faille (les termes anglais sont ceux couramment utilisés). ....	274
[C-5]	Incorporation des incertitudes sur les propriétés de la faille ....	275

**(Total: 302 figures)**

---

---

# Introduction

---

---

Les travaux présentés dans ce mémoire ont été réalisés dans le cadre du projet *gOcad* (Geological Object Computed Aided Design) développé au Laboratoire Infographie et Analyse de Données (L.I.A.D) de l'Ecole Nationale Supérieure de Géologie de Nancy (E.N.S.G) et dirigé par le professeur Jean-Laurent Mallet.

Ils ont été réalisés en collaboration étroite avec le centre de recherche de la compagnie *Elf Exploration Production* (situé à Pau) et plus particulièrement de l'équipe du *grand projet "Incertitudes"* dirigée par Bernard Corre. En effet le sujet de thèse correspond à une demande particulière de cette compagnie pétrolière: l'étude des incertitudes. Au sein de ce projet, mon travail a consisté en l'étude des incertitudes concernant la modélisation des failles, sous la direction de Pierre Thore, responsable du volet structural.

Enfin, au cours de la thèse, la collaboration s'est étendue à la compagnie *T-Surf*, dirigée par Jean-Claude Dulac, pour la plus grande partie de la conception informatique. Sa connaissance globale du logiciel *gOcad* a permis d'accélérer les choix informatiques et de nous concentrer un peu plus sur l'aspect "géologique" du travail. En outre, cela a rendu possible un rapprochement très fructueux entre les travaux présentés dans ce mémoire et ceux d'Isabelle Duvinage, qui travaille sur la détection automatique des failles et la construction de modèles à partir de données sismiques (thèse en cours au L.I.A.D. [103 - Duvinage 3]).

## 1 Origine du sujet: pourquoi étudier les incertitudes au niveau des failles?

Ce travail concerne principalement les compagnies pétrolières qui, à partir d'études variées, recherchent des gisements d'hydrocarbures. Un **gisement** est un réceptacle (ou *roche réservoir*) perméable, clos à la fois par une couverture imperméable (argiles et marnes) et par une disposition favorable du sous-sol appelé *piège*. Les pièges sont constitués environ à 60% de grès et de sables et à 40% de calcaires [1 - Barbier]. Parmi les différents types de pièges, une quantité non négligeable est liée à la présence de **failles**. Knipe considère qu'elles ont un impact sur 70% des réservoirs [136 - Knipe 1]. Dans un gisement, un **réservoir** est formé par la (ou les) couche(s) qui contien(nen)t les hydrocarbures (*gaz*, ou **pétrole**, appelé aussi «huile<sup>1</sup>»).

La phase d'**exploration** d'un champ pétrolier a pour but de décrire celui-ci de la façon la plus précise possible à partir d'un minimum de données "dures" (sections sismiques, carottes de puits et diagraphies), et d'une phase d'interprétation de ces données à l'aide de nombreux concepts géologiques. Cette description comprend différents aspects: l'aspect *structural* d'abord, qui permet de déterminer les différentes interfaces du modèle (horizons et failles), la description des *faciès* et l'évaluation des *propriétés* des différentes couches ensuite (porosité et perméabilité notamment). Ces propriétés sont estimées la plupart du temps par des méthodes géostatistiques dans des grilles suivant la géométrie définie durant la première phase. L'étude structurale est donc primordiale puisqu'elle conditionne les phases suivantes de l'exploration.

---

1. *petra oleum* = "huile de pierre"

L'exploration se déroule donc en quatre grandes phases [figure 1]:

1. l'acquisition de données,
2. la construction d'un modèle surfacique cohérent du point de vue géologique et dans les trois directions de l'espace,
3. la construction d'une grille stratigraphique où les propriétés pétrophysiques peuvent être simulées
4. la simulation des écoulements d'hydrocarbures en fonction des valeurs de porosité et de perméabilités estimées dans les différentes couches.

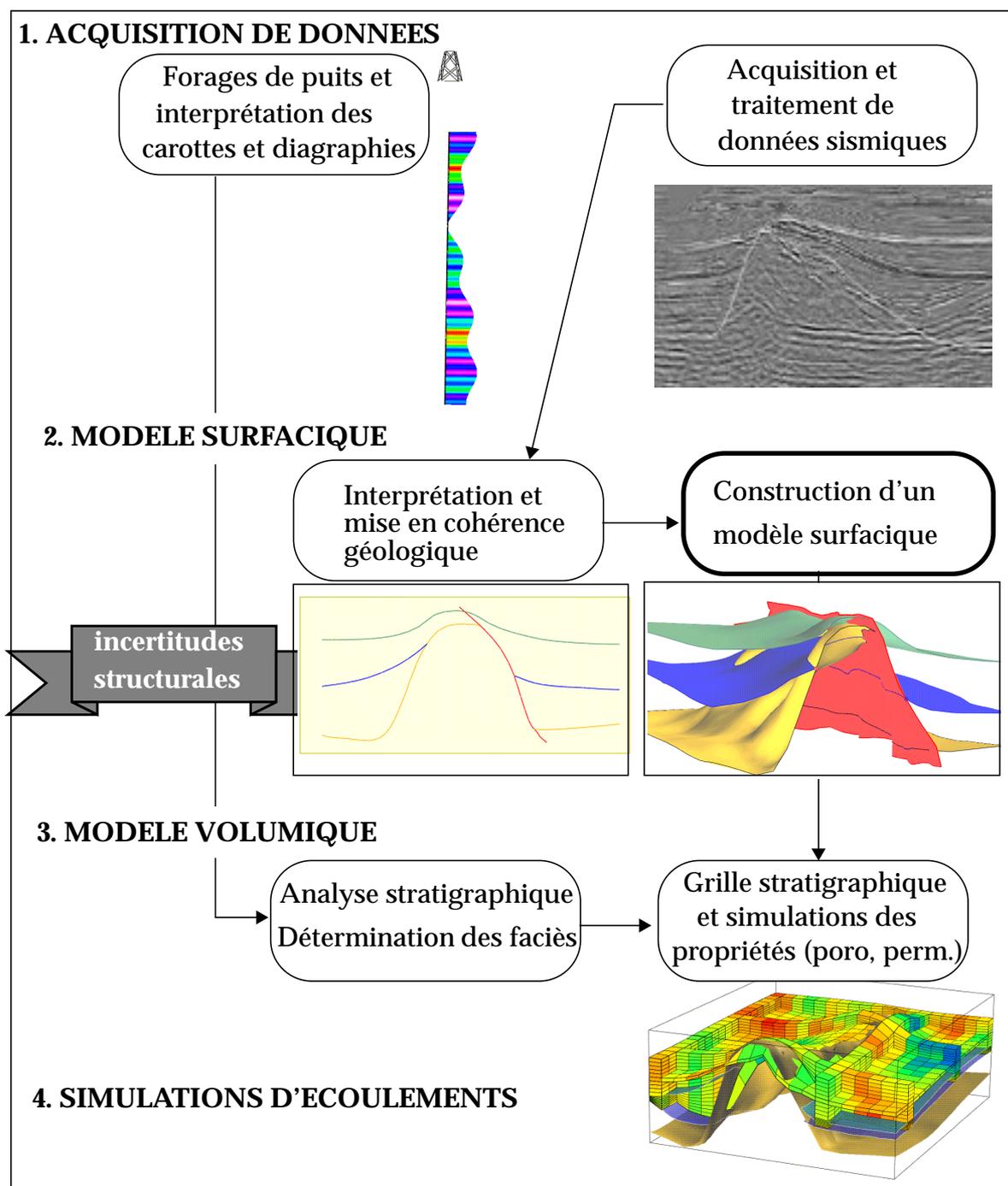


Figure 1 : Les grandes phases de l'exploration d'un champ pétrolier. Le travail présenté dans ce mémoire concerne la phase 2 (modèle surfacique).

Cette description permet d'évaluer:

- le *volume de roche*: volume géométrique inscrit entre le toit d'un réservoir, sa base éventuelle et le plan d'eau. C'est à cette étude que les résultats présentés dans ce mémoire sont destinés.
- les *accumulations*: quantité d'hydrocarbures qui après migration dans la roche mère sont piégés dans la roche réservoir d'un gisement.

$$\text{Accu} = V \text{ Roche} * \text{porosité} * \text{saturation huile} * \text{net to gross} \text{ (coefficient multiplicatif)}$$

- les *réserves*: volume d'huile qu'on espère récupérer en surface pour un schéma de développement donné

Cependant, les terrains explorés sont généralement situés en profondeur, parfois sous une hauteur d'eau importante dans le cas des champs marins. Du fait du prix des forages (de 0.5 à 12 millions de dollars par puits sur terre, deux fois plus en mer [32 - Grau]) ces **données directes** (carottes de puits) sont **peu nombreuses et relativement espacées**. Toutes les autres données disponibles donnent *indirectement* une image du sous-sol (campagnes sismiques, diagraphies). Cette image s'altérant avec la profondeur, les descriptions sont entachées de **nombreuses incertitudes** [73 - Houlding]. De plus, les étapes de la chaîne de traitement qui permet de passer des données sismiques brutes à une image interprétable sont nombreuses et comportent des inconnues (vitesse de propagation des ondes en profondeur par exemple). Enfin, ces images du sous-sol sont **interprétées** par un ensemble de géologues structuralistes. Cette interprétation n'est pas une science "exacte" et dépend de l'état de connaissance de la zone étudiée et du volume de données disponibles. De plus en plus, la *modélisation en 3 dimensions* est utilisée, afin de minimiser les incertitudes dues à cette phase d'interprétation [2 - Catalayaud].

Traditionnellement, **3 modèles structuraux** sont construits (*modèle minimum, modèle médian et modèle maximum*). Ceci est dû en partie à la difficulté et à la longueur de la construction d'un modèle fermé et cohérent. Cependant, cette approche n'est plus suffisante [70 - Thore 1], et parfois irréaliste: prendre la valeur minimum de chacun des paramètres étudiés pour obtenir le scénario catastrophe conduit souvent à construire un modèle très loin de la réalité et inacceptable du point de vue géologique [3 - Damsleth]. Mais il est très difficile de sélectionner les paramètres significatifs du point de vue des incertitudes. Ainsi, les compagnies pétrolières ne se contentent plus à ces trois modèles du réservoir, mais cherchent à **balayer l'étendue des formes possibles** que peut prendre ce modèle. En effet, les nouveaux champs sont de plus en plus réduits et comportent des structures de plus en plus complexes, augmentant beaucoup les incertitudes sur l'estimation des volumes, principalement aux abords des failles. L'industrie pétrolière étant une *industrie à haut risque* (1 puits foré sur sept seulement est producteur) [32 - Grau], **l'estimation et la réduction de ces incertitudes est devenue une priorité** pour la plupart des compagnies pétrolières. Ainsi, en limitant au maximum les surprises pouvant intervenir au cours de l'exploitation (tant du point de vue structural que des propriétés des différentes couches constituant le réservoir), les coûts d'exploitation sont diminués. Cette façon de voir le **modèle** comme **une des représentations plausibles de la réalité** est relativement récente dans le domaine pétrolier, et plus encore en géologie structurale.

Si ce domaine de recherche est encore **très nouveau dans le monde pétrolier**, les statistiques et **géostatistiques** sont largement utilisées dans le domaine minier où la nature stochastique des processus géologiques est prise en compte depuis longtemps [101 - Vistellius]. Plus récemment, ces sciences ont fait leur apparition dans le domaine pétrolier, notamment en ce qui concerne la *simulation des chenaux* à partir de données de puits [3 - Damsleth], [88 - Deutsch 1], [97 - Suro], [99 - Tran]. En géologie structurale, des études ont été réalisées sur des horizons non faillés [68 - Samson], mais en ce qui concerne la *modélisation des failles* le champ était encore très ouvert. C'est pourquoi un travail de thèse a été proposé sur ce sujet. Nous avons utilisé au maximum les connaissances actuelles en géostatistiques, ainsi que les techniques déjà employées sur les horizons, en les adaptant aux structures particulières que sont les failles. Ce travail a été rendu possible en grande partie grâce à *Arben Shtuka*, spécialiste en géostatistiques.

## 2 Les trois principaux objectifs du travail présenté dans ce mémoire

Bien que très ouvert, ce sujet comportait un certain nombre d'objectifs précis:

- Premièrement, la **modélisation des failles et de leurs incertitudes** à partir de données réelles telles que des ensembles de points représentant les différents horizons et les courbes décrivant la trace de ces failles sur les horizons (*polygones de failles*). En effet, les failles étant des objets structuraux particuliers, il nous a semblé intéressant de voir si une structure spécifique ne serait pas mieux adaptée à la modélisation des failles que les structures classiques (surface triangulées) utilisées pour modéliser les horizons. Cette partie a été abordée lors d'un stage DEA [81 - Lecour 1] et permise en grande partie grâce au travail de *Richard Cognot* qui a créé un prototype "d'objet faille" afin d'étudier la faisabilité du projet [75 - Cognot 2]. Les premiers essais réalisés sur des cas réels ont été très encourageants et montré que la paramétrisation choisie était bien adaptée aux données disponibles et aux objets particuliers que sont les failles. Cet aspect sera développé dans le chapitre 4 de ce mémoire.

Cette partie à elle seule me paraît intéressante pour la création rapide de formes de failles à partir des données sismiques, même si aucune étude d'incertitudes n'est réalisée par la suite. Ce point sera rediscuté dans le chapitre 4.

- Le deuxième et principal objectif de cette thèse est de **faire "vibrer" les failles** autour d'une *position de référence* (positionnement et géométrie). Cette partie rend possible l'étude de l'impact des variations de géométrie des failles sur le calcul des réserves.

Pour ceci, il était nécessaire de respecter certaines *contraintes* (contexte tectonique par exemple). Une des difficultés de cette partie de la thèse a été de récolter les informations décrivant ces contraintes à respecter. En effet, la plupart du temps, les règles sont bien présentes dans l'esprit des structuralistes, mais très difficiles à définir de manière quantitative. Quelques exemples de telles règles sont donnés dans le chapitre 2. De plus ces règles dépendent énormément des cas étudiés et rentrent difficilement dans une optique de "semi-automatisation" du processus de construction des failles.

Pour ces raisons, nous nous sommes surtout attachés à **respecter au maximum la forme de la faille de référence** dessinée par le géologue structuraliste, ainsi que la hiérarchie entre les failles. Cependant, nous avons gardé à l'esprit la possibilité de respecter de nouvelles contraintes d'ordre mécanique lorsque celles-ci seront mieux formalisées.

- Enfin, ce travail devait garder à l'esprit le fonctionnement du logiciel JACTA pour lequel il est destiné. Ce logiciel permet actuellement de simuler de façon emboîtée différents paramètres (géométrie des horizons, répartition des faciès, propriétés des couches). Ces simulations sont réalisées à l'intérieur de grilles irrégulières qui suivent la stratigraphie du modèle. Actuellement, les incertitudes concernant les failles ne sont que partiellement prises en compte dans ce logiciel, du fait de la difficulté de modéliser des **grilles faillées** complexes, thème de recherche menée au L.I.A.D. toujours d'actualité pour de nombreux géomodeleurs.

C'est pourquoi, en attendant que les recherches aboutissent dans ce domaine, nous avons utilisé un autre moyen pour étudier l'impact des incertitudes sur les failles sur le calcul des réserves: les modèles surfaciques. Ceux-ci sont suffisants pour réaliser des études volumétriques. Il fallait toutefois **rester compatible avec le logiciel existant** et prévoir les algorithmes de manière à ce que les 2 artisans de ce logiciel puissent intégrer le plus facilement possible un module de simulation du réseau de failles.

## 3 Echelles de travail et plan du mémoire

### 3.1 Echelles de travail

Le travail s'est effectué à différentes échelles:

- **A petite échelle:** la première partie de l'étude s'est effectuée à l'*échelle de la faille* elle-même, afin de déterminer les éléments nécessaires à la bonne description des surfaces de faille compte tenu des données disponibles, et à la prise en compte de données d'incertitudes. C'est aussi à cette échelle qu'ont été mises au point les différentes méthodes de simulation de nouvelles positions et de formes pour les failles.
- **A plus grande échelle:** ces méthodes de simulations ont ensuite été étudiées à l'*échelle du réservoir*, afin de pouvoir maintenir les relations cohérentes entre les différents éléments du modèle structural (horizons et failles).

### 3.2 Plan suivi dans ce mémoire

Le plan suivi dans ce mémoire est bâti à la fois sur les différents objectifs à atteindre et sur les deux échelles d'étude.

#### 3.2.1 Présentation du cadre de la thèse

Cette première partie, contenant trois courts chapitres, permet de mieux cerner le sujet, et de décrire les problèmes posés de façon détaillée.

1. Le premier chapitre **présente les 3 équipes** avec lesquelles ce travail s'est effectué: le *projet incertitudes* de la compagnie Elf Exploration Production qui est à l'origine du sujet, le *projet gOcad* où la majeure partie du travail s'est effectuée ainsi qu'une brève présentation de la compagnie *T-Surf*.
2. Le deuxième chapitre concerne lui la description de **ce qu'est une faille** et les techniques couramment utilisées pour déterminer leur *géométrie*.
3. Le dernier chapitre de cette partie présente succinctement **les étapes** précédant la construction d'un modèle structural en trois dimensions, et donne ainsi un bref **bilan des incertitudes** affectant les failles.

#### 3.2.2 Etude des incertitudes à l'échelle de la faille elle-même

Cette seconde partie concerne plus directement l'étude des incertitudes au niveau des failles, mais uniquement à petite échelle et sur une faille isolée.

4. Le quatrième chapitre présente une description précise de la **paramétrisation choisie** pour modéliser les failles et les incertitudes qui lui sont associées. Cette paramétrisation a été choisie pour répondre à différents critères: elle est adaptée aux données sismiques servant à modéliser les failles; elle est suffisamment flexible pour permettre des modifications de la forme de la faille en fonction de nouvelles données ou d'une nouvelle interprétation structurale; et surtout, elle permet de **définir une zone d'incertitude** autour de la faille de manière flexible. Cette paramétrisation permet donc la construction d'une **surface de faille "de référence"**, correspondant au mieux à l'interprétation du géologue structuraliste, en incorporant différents types de données.
5. Le cinquième chapitre montre sur quelques exemples, l'utilisation de cette structure de données.

6. Le chapitre six décrit trois **méthodes de simulation** permettant de générer de nouvelles formes et positions de faille dans la zone d'incertitude préalablement définie. Ces méthodes, assez simples, utilisent le même schéma de base, et s'attachent à respecter au maximum la surface de référence modélisée en accord avec le géologue structuraliste. Ce chapitre décrit aussi comment la définition de **contraintes géométriques** permet de conserver tous les contacts (à la fois faille-faille et horizon-faille) au cours du processus de simulation. Ces contraintes géométriques peuvent être utilisées dans le cas de modèles simples.

Ces deux chapitres permettent de détailler les mécanismes de base que nous avons utilisés pour modéliser les incertitudes au niveau des failles. Afin d'étudier des modèles plus complexes, il faut ensuite passer à une autre échelle de travail.

### 3.2.3 Etude des incertitudes à l'échelle du réservoir

7. Le septième chapitre décrit différentes manières de **construire des modèles volumiques**, et les raisons qui ont guidé notre choix pour construire les modèles volumiques: le modèle volumique continu.
8. Le huitième chapitre décrit une **nouvelle structure de données** permettant de gérer les relations entre les différentes interfaces du modèle structural (horizons et failles) au cours des simulations, ainsi que la **mise à jour du modèle volumique** après chaque simulation, sans avoir à tout reconstruire tout le modèle depuis le début. Cette structure de données est indispensable lorsque le modèle devient compliqué et contient un grand nombre de contacts entre les failles et les horizons.
9. Le neuvième chapitre de ce mémoire présente les différentes étapes de l'algorithme de **simulation d'un modèle structural**. Cet algorithme permet de générer un grand nombre de simulations sur un modèle donné et de suivre au fur et à mesure l'évolution des volumes des différentes régions du modèle.

### 3.2.4 Résultats et perspectives

La dernière partie concerne les premiers résultats obtenus et les perspectives. En effet, les incertitudes sur les failles ne se bornent pas à des petites variations autour d'un scénario donné, mais sont de natures très variées.

10. Le dixième chapitre montre sur deux exemples simples les résultats obtenus, ainsi que l'interface utilisateur qui est proposée.
11. Le onzième chapitre expose en quoi être capable de générer rapidement **plusieurs scénarios** de réseaux de failles est important pour la définition de la géométrie du réservoir. Les prémices de ce qui pourrait être fait dans ce sens sont décrites à la fin de ce chapitre.

Enfin, une conclusion générale donne un **bilan** de la méthodologie employée, de ses **limites** actuelles et des **améliorations** qui peuvent être apportées par la suite (par exemple, en quoi l'**étude des propriétés de la zone de faille** est fondamentale pour l'étude réservoir et la suite de l'étude des incertitudes liées aux failles).

---

---

# Chapitre 1 Présentation du cadre de travail.

---

---

Comme cela a été énoncé dans l'introduction, ce sujet fut proposé par une compagnie pétrolière (Elf Exploration Production).

Cependant, le travail de thèse s'étant effectué dans trois endroits différents (Pau pour Elf, le L.I.A.D. à Nancy pour la majeure partie du travail et T-Surf Corporation à Houston pour le restant), il me semble utile de présenter quelles étaient les attentes du projet incertitudes et comment le projet gOcad permettait de répondre à ces attentes.

## 1. Le projet incertitudes

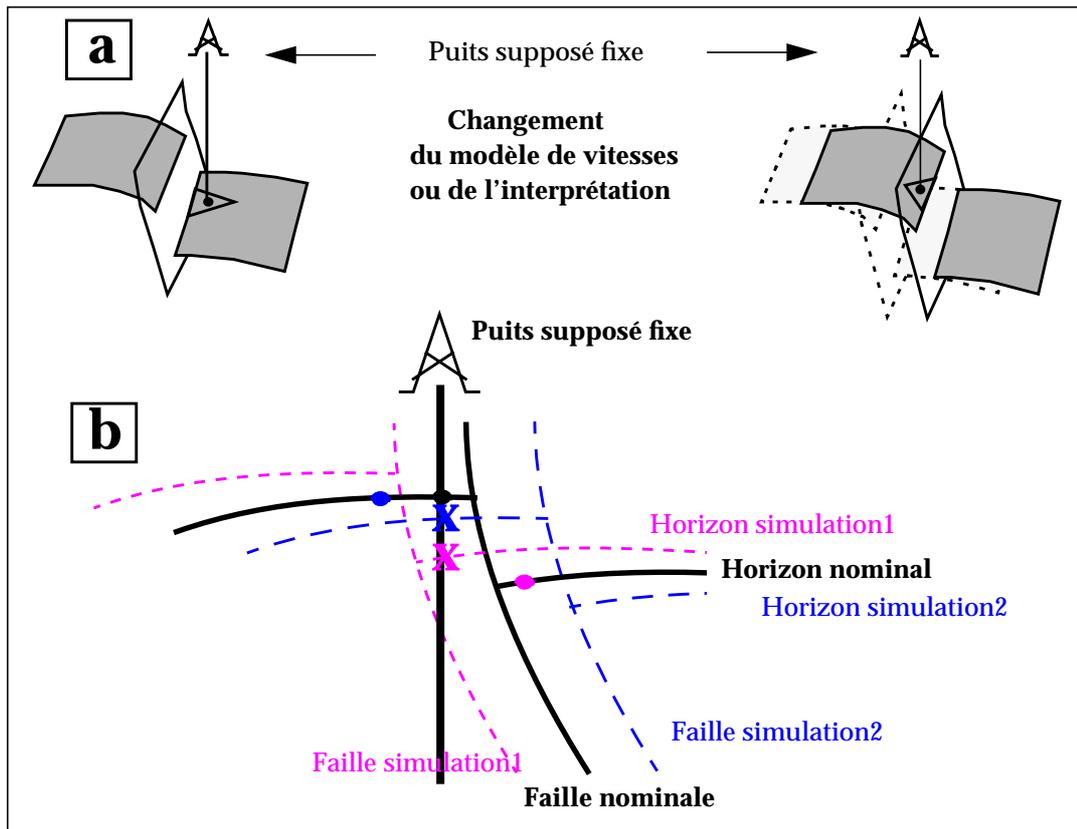
Ce projet de recherche de la société *Elf Exploration production* a vu le jour en 1993. Il a pour but de mettre au point des techniques permettant de fournir **plusieurs images équiprobables d'un même modèle de réservoir**, compte tenu des nombreuses incertitudes apparaissant au cours de la modélisation et de l'étude de ce modèle. Ces incertitudes sont de différents ordres: géophysiques (vitesses de migrations mal connues, mauvaise sismique), structurales (complexité de la zone à étudier), propriétés des roches (fluides présents, type de roches concernées), etc.

L'équipe pluridisciplinaire qui travaille sur ce projet est en charge de l'étude globale des incertitudes à partir de l'acquisition sismique jusqu'à l'exploitation du champ [72 - Vincent]. Le projet comporte donc plusieurs volets:

- le premier concerne l'aspect **structural** du modèle. Il permet d'étudier les incertitudes sur la géométrie du réservoir. C'est dans cette partie que s'inscrit ce travail de thèse. Un premier logiciel [*Alea*] permet d'estimer les incertitudes structurales, une seconde application [*Alea 3D*] permet d'étudier l'influence de ces incertitudes sur l'implantation des puits.
- le second volet, concerne plus particulièrement l'étude des incertitudes sur les **faciès** et sur les **propriétés** des roches. Il inclut les incertitudes structurales déterminées auparavant grâce au logiciel [*Jacta*] qui peut tenir compte de nombreux types d'incertitudes.
- le dernier s'intéresse à l'étude des incertitudes sur l'**écoulement des fluides** dans le réservoir, compte tenu de toutes les incertitudes précédentes.

Une collaboration entre Elf E. P. et le L.I.A.D. a démarré au début de cette thèse mais va bien au delà du travail décrit dans ce mémoire. Elle intervient principalement à deux niveaux: le logiciel Alea3D et le logiciel JACTA.

Le travail concernant Alea3D a été réalisé principalement par Pierre Thore, en collaboration avec Taoufik Ait Ettajer [figure 1-1]. L'apport du travail présenté dans ce mémoire à Alea3D sera présenté dans le chapitre six.



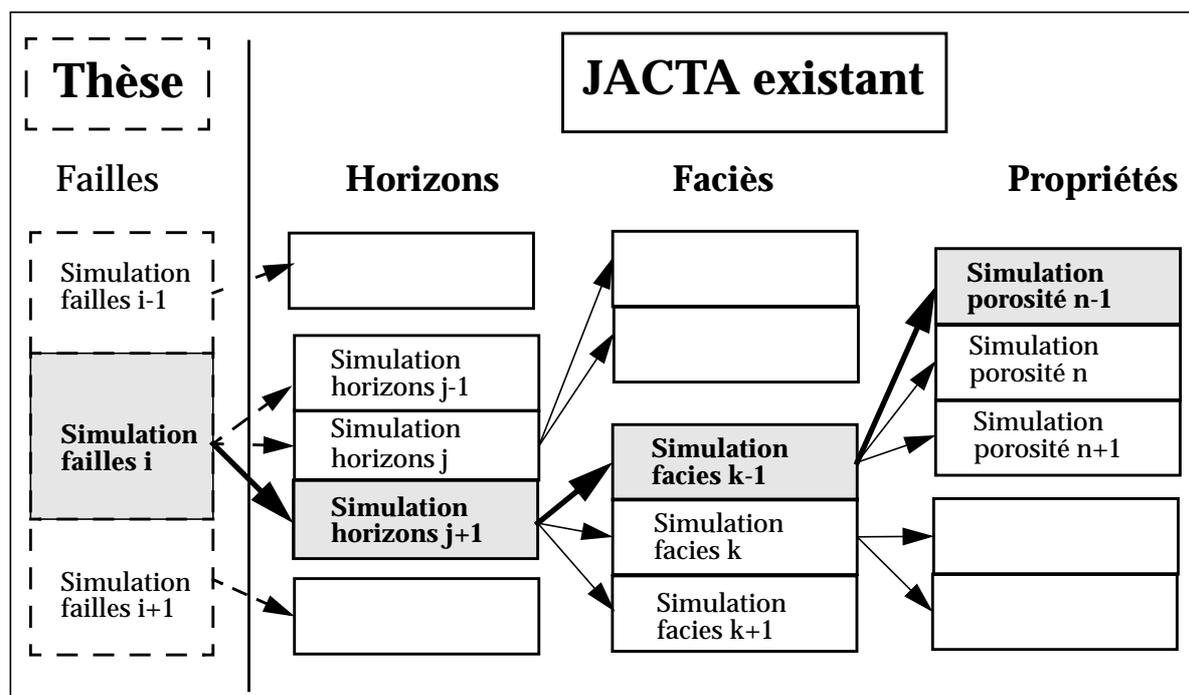
**Figure 1-1 :** Alea3D permet de suivre le point d'impact d'un puits sur un horizon lorsque la géométrie de cet horizon est modifiée, suite à une modification du modèle de vitesses ou à une nouvelle interprétation. A chaque interprétation, le nouveau point d'impact est calculé [a]. Les points d'impacts ainsi calculés (X) peuvent ensuite être reportés sur l'horizon nominal (disques sur horizon de référence) [b]. Ceci permet de vérifier si un changement d'interprétation risque de faire passer le puits de l'autre côté de la faille.

Cette collaboration a aussi permis la mise au point d'un logiciel *JACTA*, mixant de manière emboîtée des incertitudes structurales (sur les horizons principalement, pour l'instant), des incertitudes sur les cartes de faciès et des incertitudes concernant les propriétés (porosité et perméabilité par exemple). Ce travail a été réalisé par Arben Shtuka et Philippe Pluyaud<sup>1</sup> pour la partie géostatistique/informatique et par les membres du projet incertitude pour les spécifications «métier».

En un cycle de simulations, il est ainsi possible de produire relativement rapidement un grand nombre de modèles équiprobables, compte tenu des incertitudes fournies à chaque niveau [figure 1-2]. Pour chaque modèle créé, les réserves sont calculées. Un histogramme permet de visualiser un bilan de ces simulations et d'extraire quelques modèles représentatifs (cas "catastrophe", cas "jackpot" et quelques modèles intermédiaires).

Les modèles volumiques sélectionnés sont ensuite fournis aux simulateurs d'écoulement afin de tenir compte de nouveaux types d'incertitudes. Les simulateurs d'écoulement ne travaillant que sur des grilles, les modèles choisis sont transformés sous forme de grilles dont les cellules suivent la stratigraphie [appelées *SGrid* dans *gOcad*].

1. De la compagnie T-Surf S.A.



**Figure 1-2 : Principe des simulations emboîtées de JACTA.** Pour l'instant, seuls les horizons, les faciès et les propriétés sont simulés aléatoirement. Ce travail de thèse permettra d'ajouter la *simulation de la géométrie des failles* au processus.

*Restait à insérer les incertitudes structurales concernant les failles...*

En effet, pour l'exploration et la production pétrolière, la modélisation précise des failles est essentielle (elles interviennent non seulement dans la délimitation du volume de roche mais aussi dans la compartimentation interne du gisement).

Cependant, les failles sont des objets "complexes" obéissant à des règles géologiques précises, ce qui rend leur modélisation délicate. La détermination de la hiérarchie dans le réseau de failles est une contrainte supplémentaire à respecter lors de telles simulations sur un modèle faillé. De plus, dans tous les logiciels de modélisation actuels, la représentation des failles reste assez rudimentaire au sein des grilles, compte tenu des difficultés de connections entre les cellules de part et d'autres des failles, ou de l'apparition de cellules dégénérées inacceptables pour les simulateurs d'écoulement par exemple [113 - Quettier]

Compte tenu de toutes les difficultés liées à la nature même des failles, cet aspect des incertitudes structurales a été laissé de côté au début du projet.

Comme cela a été annoncé dans l'introduction, le but initial de cette thèse est de préparer l'ajout d'un nouveau module de simulation au logiciel existant: la simulation de la géométrie des failles.

Nous verrons au cours de ce mémoire les objectifs atteints, et pourquoi certains d'entre eux n'ont pas pu être abordés.

## 2. Brève présentation du logiciel gOcad

### 2.1 Un géo-modeleur

Le logiciel gOcad a été créé en 1988 par le Professeur Jean-Laurent Mallet. Partant du principe que les logiciels de Conception Assistée par Ordinateur (C.A.O.) classiques n'étaient pas adaptées à la construction de modèles géologiques, il a créé un nouveau logiciel dédié à la modélisation des objets du sous-sol.

En effet, les logiciels existants étaient souvent issus du monde automobile où la topologie<sup>1</sup> des objets représentés ne change pas, où les modifications de géométrie sont restreintes, et pour lequel le besoin d'honorer des données de natures diverses et irrégulièrement réparties se fait moins sentir.

*“Une porte de voiture se ramène toujours à un rectangle de tôle percé d'une vitre” (J-L Mallet)*

Pour de telles modélisations, il est important de créer de “belles” lignes ou surfaces sans forcément attacher des propriétés aux objets modélisés. Les représentations paramétriques telles que les courbes de Bézier suffisent pour modéliser des objets aussi simples [6 - Mallet 1].

Au contraire, les structures géologiques sont très différentes d'un champ à l'autre, en fonction de la tectonique et des types de terrains rencontrés. L'acquisition d'un grand nombre de données précises sur de telles étendues et à de telles profondeurs étant impossible du point de vue économique, elles sont décrites par des données de natures diverses et ayant des degrés de confiance variables:

\* Les données de puits qui sont précises (précision de l'ordre du centimètre) et directes mais peu nombreuses et dispersées

\* Les données sismiques qui sont continues, couvrent un volume beaucoup plus grand, mais ont un degré de confiance moindre (résolution de l'ordre de la dizaine de mètres). En effet, une longue chaîne de traitement est nécessaire pour transformer les données sismiques brutes en données interprétables et de nombreuses inconnues interviennent au cours de cette chaîne de traitement. Cet aspect incertain des données sismiques sera développé dans le troisième chapitre.

\* Les informations structurales telles que le pendage d'une couche, le rejet que doit avoir une faille à un niveau précis. Ces données ne sont pas toujours des données «sures» mais correspondent au schéma structural que le géologue a en tête au moment de l'interprétation ou de la mise en cohérence du modèle. Elles peuvent changer lorsque de nouvelles données sont disponibles, ou lorsqu'un nouveau concept géologique est utilisé pour interpréter la zone étudiée.

De plus, en géologie, les objets modélisés sont très étroitement liés aux *propriétés* qui leur sont attachées, et il faut être capable de représenter et d'éditer en même temps la géométrie et les propriétés de ces objets. Deux propriétés très étudiées sont la porosité (volume de vide) et la perméabilité (connections entre les vides) des couches.

Enfin, un nouveau puits ou une nouvelle campagne sismique peuvent à tout moment remettre en question l'interprétation en cours. Ainsi, dans la figure suivante, une nouvelle interprétation de la sismique nécessite d'ajouter une faille au modèle existant. Il faut donc être capable d'insérer la nouvelle interprétation à ce modèle, sans avoir à le reconstruire complètement [figure 3-1].

---

1. **Topologie:** description des relations entre les éléments constitutifs d'un objet ou entre plusieurs objets d'un même modèle

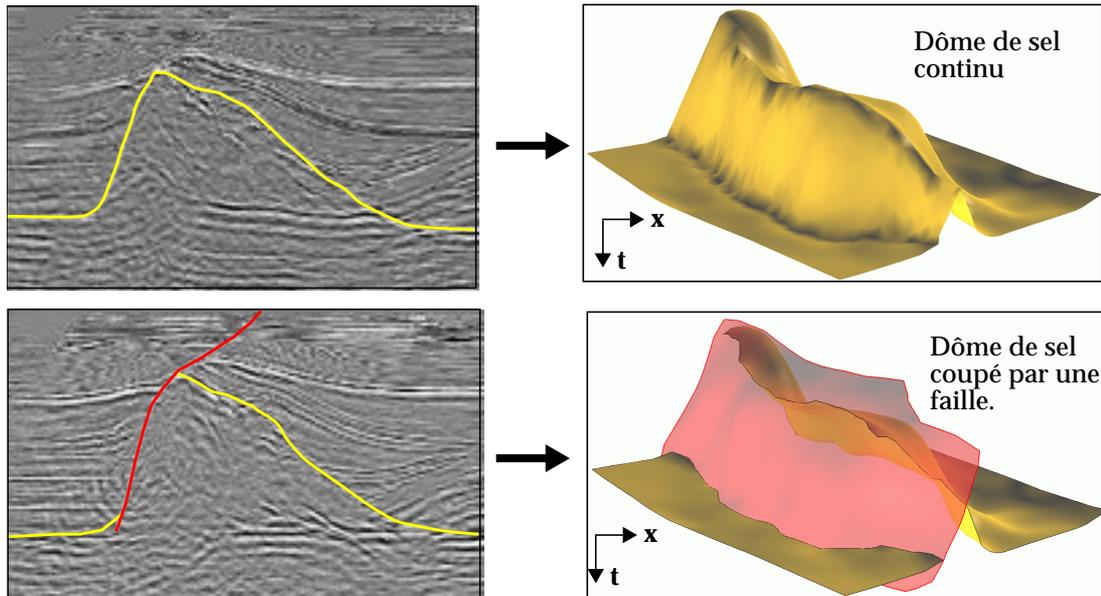


Figure 1-3 : **Change de topologie après nouvelle interprétation**

Il est clair que lorsqu'on cherche à gérer les relations entre objets géologiques, les deux interprétations précédentes n'ont pas du tout la même signification du point de vue topologique.

Sans appliquer un changement aussi radical, de petites modifications de la géométrie des surfaces doivent régulièrement être faites lors de la construction du modèle, sans pour autant nécessiter la reconstruction totale de ce modèle. En effet, la modélisation d'objets géologiques se fait souvent lors de processus itératifs permettant d'améliorer les interprétations des données à chaque itération.

Le paragraphe suivant décrit brièvement l'approche suivie par gOcad. Cette description se veut la plus simple possible. Une description plus précise et plus mathématique du fonctionnement du logiciel, ainsi que de nombreux articles ou thèses déjà consacrés à ce sujet sont disponibles ([4 - Cognot 1], [6 - Mallet 1], [7 - Mallet 2]).

## 2.2 Une collection de structures simples représentant les différents objets géologiques.

L'approche utilisée par gOcad est une *approche discrète* particulièrement adaptée aux objets complexes que sont les formes géologiques. Chaque type de données peut ainsi être représenté par des **éléments de base** qui peuvent ainsi être répartis de manière dense dans les zones complexes. Par exemple, il suffit de 2 triangles pour représenter une surface plane paralléloïdale, alors qu'une zone fortement plissée demande un maillage plus précis.

Pour les représentations les plus simples, on peut citer:

- \* des ensembles de points

Ces points représentent la plupart du temps des *interprétations de données sismiques migrées temps*. Ce sont soit des horizons pointés manuellement ou à l'aide de processus semi-automatiques, soit des failles pointées sur des sections sismiques verticales.

- \* des lignes

Elles peuvent correspondre à des *interprétations de failles* dessinées par le géologue structuraliste lors de la mise en cohérence du modèle. Elles correspondent aussi aux délimitations du modèle sur une coupe verticale (modèle en 2 dimensions).

\* les surfaces triangulées

Elles sont principalement utilisées pour représenter les interfaces du modèle (failles et horizons). Chaque surface peut être composée de plusieurs morceaux (par exemple, un horizon coupé par une faille sera représenté par 2 portions de surfaces, de part et d'autre de la faille). Ceci permet de garder la signification géologique des différents objets représentés: l'horizon ainsi défini peut ensuite être traité globalement ou partie par partie, en fonction de ce que l'utilisateur demande.

A ces objets de base, s'ajoutent un certain nombre d'**objets spécialisés** comme:

\* les puits, qui permettent de visualiser les différentes propriétés enregistrées lors des analyses diagraphiques, de spécifier des marqueurs correspondant à leur intersection avec certains horizons ou certaines failles (localisation et pendage)

\* les gshapes ("geological shapes") permettant de modéliser les éléments des environnements fluviaux (chenaux, lobes....)

\* les volumes tétraédrysés qui servent notamment au lancé de rayons.

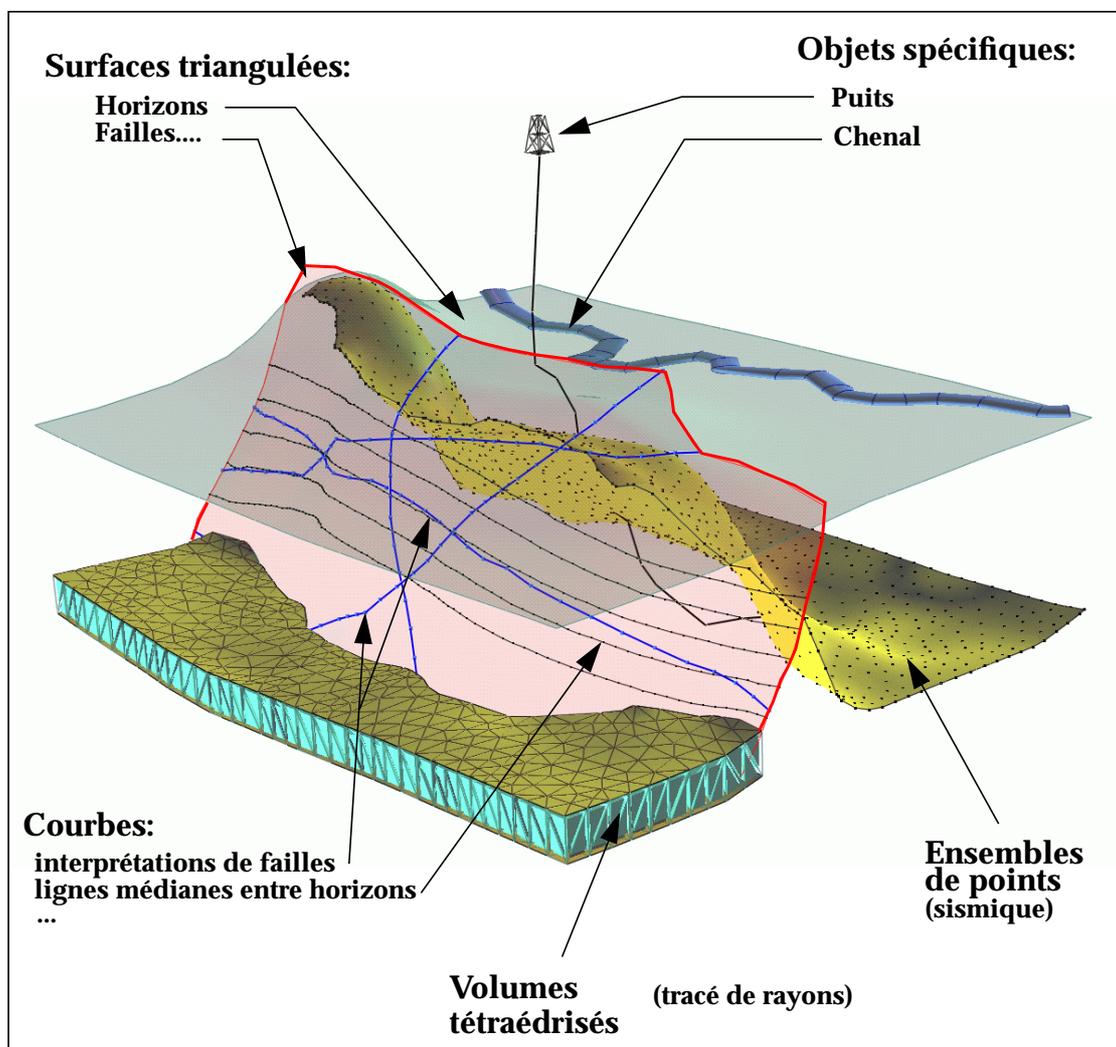
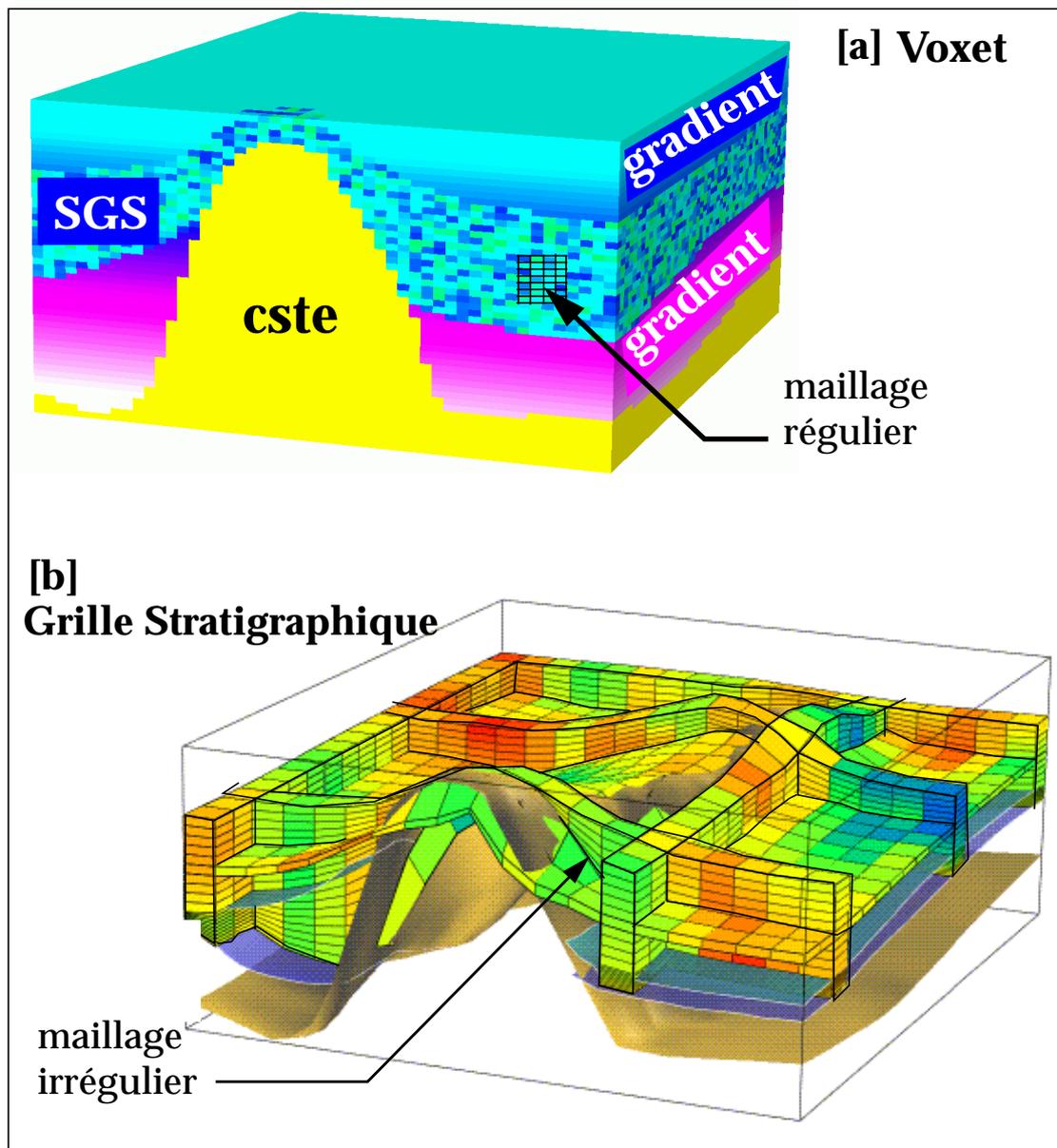


Figure 1-4 : Quelques structures représentant les objets géologiques.

De telles structures simples (ensembles de points, lignes polygonales, surfaces triangulées) suffisent à modéliser des structures géologiques complexes. Les objets spécifiques (puits et chenaux) permettent de rajouter de l'information pour l'étude des propriétés des différentes couches.

Enfin, deux types de **grilles** permettent de réaliser des simulations et de visualiser les propriétés dans les différentes régions délimitées par les interfaces décrites précédemment (horizons et failles). Deux types de grilles sont disponibles [figure 1-5]:

- les **grilles régulières** (ou Voxet dans gOcad). Ces grilles peuvent être utilisées pour créer des *modèles de vitesse* à partir d'un modèle surfacique. Ainsi, chaque région du modèle utilise sa propre définition de la vitesse (constante dans le sel, gradient, simulation géostatistique dans les zones chenalisées...)
- les **grilles qui suivent la stratigraphie** ("Stratigraphic Grid" dans gOcad). Un travail conséquent a été effectué durant ces dernières années afin que ces grilles puissent épouser la géométrie des failles. Les simulations de propriétés se font soit de façon continue de part et d'autre de la faille, soit séparément dans les deux blocs. Ces simulations peuvent être conditionnées par des données de puits.



**Figure 1-5 : Grilles régulières (Voxet) et grilles stratigraphiques (SGrid).**  
 Dans le voxet [a], un *modèle de vitesse* a été calculé. Dans la grille stratigraphique [b], une simulation gaussienne séquentielle (SGS) a été générée pour simuler la *porosité*.

### 2.3 Des interactions entre ces objets par l'intermédiaire du moteur d'interpolation: D.S.I.

Dans le domaine des géosciences, il ne suffit pas d'être capable de représenter les différentes formes géologiques, et leurs propriétés. Il faut en outre être capable de gérer les *interactions* entre les différents objets. Une nouvelle campagne sismique, une meilleure connaissance du style de tectonique ou du milieu de dépôt de la zone étudiée peuvent à tout moment amener les géologues à modifier le modèle existant au vu des nouvelles informations.

Or, dans de nombreux cas, il est plus judicieux de modifier un modèle existant aux endroits où la nouvelle interprétation ne correspond pas, plutôt que de tout reconstruire à partir des nouvelles données. Ceci permet de conserver le travail effectué sur les zones qui ne sont pas modifiées.

Les nouvelles interprétations doivent pouvoir agir sur le modèle existant de différentes manières:

- \* Une *nouvelle interprétation* des données sismiques (ensemble de points) doit pouvoir interagir sur l'ancienne surface modélisée et pouvoir l'attirer vers les nouvelles positions.

- \* Afin de définir des régions dans le modèle, il est primordial de créer des *contacts parfaits* entre les différentes interfaces (horizons et failles). Si plus tard, l'une de ces surfaces est modifiée, il peut être utile d'avoir un mécanisme qui permet de modifier en conséquence les surfaces avec lesquelles elle est en contact.

- \* Une *propriété* connue au niveau d'un puits doit pouvoir entrer comme donnée dure lors d'une simulation géostatistique.

\* ...

Comme on le voit, ces interactions font intervenir des objets très différents (deux surfaces, des points et une surface, un puits et une grille...). De plus, un même objet peut intervenir dans plusieurs types d'interaction, en étant tantôt "*contrôleur*", tantôt "*contrôlé*".

Pour le logiciel gOcad, toutes ces interactions sont gérées sous forme de contraintes installées sur les différents objets. Une contrainte permet le plus souvent de contrôler le comportement (géométrie ou propriétés) d'un objet par des informations issues d'un autre objet (le contrôleur).

Parmi ces contraintes, on peut citer de manière non exhaustive [figure 1-6]:

- \* **Points d'attraction** (*Fuzzy Control Points FCP*): un objet (généralement un ensemble de points) en attire un autre

- \* **Nœud de contrôle** (*Control Node CN*): un nœud est fixé et ne sera pas modifié lors de l'interpolation. Une contrainte semblable peut aussi s'appliquer pour fixer des valeurs de propriété en certains points connus (au niveau des puits par exemple).

- \* **Bord sur surface** (*On Triangulated Surface OTS*): le bord de la surface contrôlée doit se coller sur une autre surface triangulée (contrainte très utilisée pour le calcul des contacts entre les différentes interfaces afin de fermer les modèles structuraux).

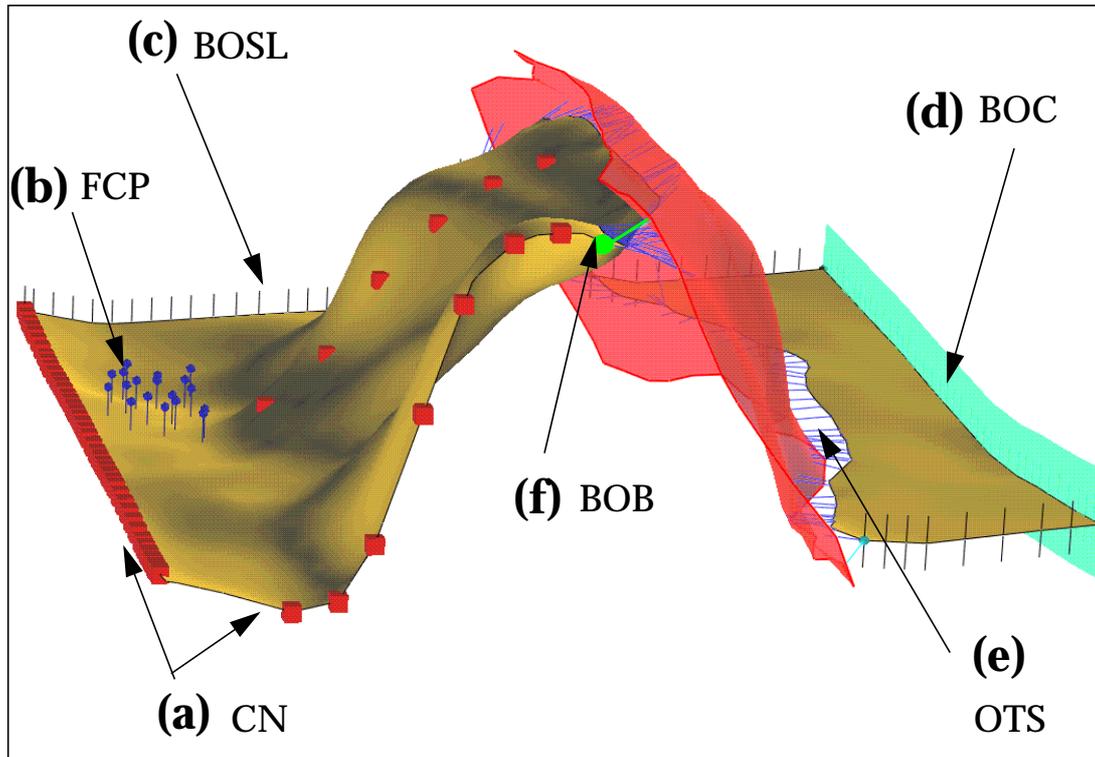
- \* **Bord sur bord** (*Border On Border BOB*): impose à un point du bord d'une surface (horizon le plus souvent) de bouger le long du bord d'une autre surface (faille en général). Cette contrainte permet de conserver la largeur de l'empreinte de la faille sur l'horizon lors de l'interpolation de celui-ci.

\* **Bord bougeant selon un axe (*Border On Straight Line BOSL*)**: les points du bord contrôlés par cette contrainte ne peuvent se déplacer que le long d'un axe

\* **Bord bougeant delon un plan (*Border On Cylinder BOC*)**: les points du bord contrôlés par cette contrainte ne se déplacent que le long d'un plan lors de l'interpolation.

\* **Définition du rejet (*Fuzzy Vectorial Link FVL*)**: cette contrainte permet de fixer la valeur et la direction du rejet entre les 2 lèvres d'une faille

\*...



**Figure 1-6 :** Quelques exemples de contraintes géométriques qui peuvent être prises en compte en même temps par l'interpolateur D.S.I. a): **Control Node**: le nœud est fixé (peut s'appliquer sur tout un bord ou sur quelques points correspondant à des coupes par exemple); b) **Fuzzy control Points**: les points issus de la sismique attirent la surface triangulée; c) **Border On Straight Line**: les points du bord ne peuvent bouger que le long d'une droite; d) **Border On Cylinder**: les points du bord ne peuvent bouger que le long d'un plan; e) **Border On Surface**: le bord de l'horizon doit aller se fixer sur la surface de faille; f) **Border On Border**: le coin de l'horizon doit coulisser sur le bord de la faille.

L'interpolateur DSI (**Discrete Smooth Interpolator**: le coeur de gOcad) permet de tenir compte au mieux de toutes les contraintes installées sur un objet ou sur une partie d'un objet. Cette méthode d'interpolation est une méthode itérative qui converge rapidement et qui est stable, même lorsque les données réparties de manière hétérogène [6 - Mallet 1].

Un autre avantage non négligeable de cette méthode est qu'elle est définie de manière globale et qu'elle permet d'interpoler autant les propriétés que les coordonnées géométriques.

Enfin, des coefficients de certitude (*certainty factors*) peuvent être assignés à chacune des contraintes et donner un poids plus important à certaines contraintes (par exemple un point fixé par le passage d'un puits aura plus de poids qu'un point attiré par une donnée issue de la sismique).

### 3. Avantages de travailler avec un tel géo-modéleur

Comme nous l'avons décrit précédemment, gOcad est particulièrement adapté à la disparité des données disponibles, à la modélisation des formes géologiques et de leurs propriétés, mais surtout à l'édition de ces objets géologiques. Il répond ainsi aux nouveaux besoins qui se sont fait sentir dans le domaine des géosciences: construire les modèles directement en 3 dimensions en tenant compte des données hétérogènes.

Concernant plus particulièrement le travail présenté dans ce mémoire, travailler avec ce logiciel présente de nombreux avantages qui sont résumés dans les paragraphes suivants.

**Remarque:** ce travail de thèse se trouvant à la frontière entre deux domaines (la géologie et l'informatique), certains paragraphes (dont le suivant) décrivent les structures informatiques utilisées, et contiennent parfois des exemples de code. Ils permettent d'avoir une description plus précise du travail réalisé, mais ne sont pas indispensables pour la compréhension de l'ensemble de ce mémoire.

#### 3.1 Langage C++ et héritage des structures existantes

Ce logiciel est écrit en langage C++, langage orienté objet, qui lui donne un certain nombre d'avantages lors de la programmation. Le propos n'est pas ici, de détailler ce qu'est un langage orienté objet, mais de montrer brièvement et à l'aide d'exemples simples, dans quelle mesure cela a facilité ce travail de thèse.

##### 3.1.1 Notion d'objet

Une des notions de base de la programmation orientée objet est justement la notion d'objet. En programmation classique, une structure de données permet de regrouper sous un même nom de structure un certain nombre de champs suffisants pour décrire cette structure.

*Exemple de code 1..1. programmation classique*

```
1  typedef struct surface
2      {
3      char* nom ;
4      int nb_points ;
5      Point3d* points ;
6      char* couleur ;
7      }
```

Sur une telle structure, il est possible de définir des opérations utilisant les champs de cette structure. Une fonction peut par exemple utiliser le tableau de points pour les dessiner sur un écran. Mais la structure surface elle-même n'en n'est pas responsable, et n'a pas connaissance qu'elle peut être dessinée. Cette structure ne sert que pour le stockage des champs nécessaires à la définition d'une surface.

En programmation orientée objet, les structures de données (*classes*) contiennent non seulement les champs nécessaires à leur description, mais contiennent en plus un certain nombre de fonctionnalités communes à tous les objets de la même classe. Ces fonctionnalités sont communément appelées "*fonctions membres*" (ou méthodes) de la classe. C'est la classe elle-même qui effectue les actions qui sont déclarées dans sa définition. Le paragraphe suivant montre aussi qu'elle est responsable de la libération des champs qui la composent lorsqu'elle est détruite.

*Exemple de code 1..2. programmation orientée objet (exemple C++)*

```

1  class Surface
2  {
3      public:
4          // Constructeurs
5          Surface(
6              char* name, int nb_points, Point3d* points,
7              char* couleur = nil
8          ) ;
9          Surface( char* name, int nb_points, Ligne* bord ) ;
10         Surface() ;
11
12         // Destructeur
13         ~Surface() ;
14
15         // Fonctions membres
16         void dessine(Support support) ;
17
18         void interpole(argument1, argument2);
19         void interpole(argument1, argument3, argument4) ;
20
21         inline int nb_points() const { return nb_points_; }
22         inline const Point3d& point(int index) ;
23
24     private:
25         int nb_points_;
26         Point3d* points_ ;
27
28     } ;
29
30 //-----
31 inline const Point3d& Surface::point(int index) {
32     return points_[index] ;
33 }

```

Ici, la *classe Surface* contient non seulement ses variables membres (*nb\_points\_*, *points\_*), mais aussi un certain nombre de fonctions comme *dessine(...)* qui lui permet de se dessiner à l'écran.

Deux types de fonctions sont à remarquer. Les *constructeurs* et le *destructeur*.

Les *constructeurs* (lignes 5 à 8, 9 et 10) décrivent toutes les manières dont un objet du type "Surface" peuvent être construit. Dans le cas présent, on autorise 3 manières de créer une surface, le dernier constructeur construisant une surface (vide) par défaut lorsqu'aucune information n'est donnée.

Le *destructeur* (ligne 13) est responsable de la libération de la mémoire lorsque l'objet est détruit. Ainsi, si une surface S1 a été créée, le tableau contenant la localisation des points qui la composent est libéré à sa destruction. Cette sécurité n'était pas assurée dans l'exemple de code 1..1.

On voit ici un autre avantage du C++: c'est un langage typé. Ceci signifie que si plusieurs fonctions sont définies avec le même nom mais avec des paramètres différents, la fonction appelée correspond aux paramètres donnés en arguments. Ceci est valable non seulement pour les constructeurs, mais aussi pour les autres fonctions membres de la classe (comme la fonction *interpole(...)*).

### 3.1.2 Notion d'héritage.

Cette notion a beaucoup été utilisée lors de la thèse. Le C++ permet en effet de définir une classe (*classe dérivée*) à partir d'une classe déjà existante (*classe parente*), et ainsi de lui faire bénéficier de toutes les définitions de la classe parente.

*Exemple de code 1.3. dérivation d'une classe "Faille"*

```

1  class Faille1: public Surface
2      {
3          // Constructeurs
4          Faille(
5              char* name, int nb_points, Point3d* points,
6              char* couleur = nil, int type
7          ) ;
8          Faille() ;
9
10         // Destructeur
11         ~Faille() ;
12
13         // Fonctions membres
14         void calcule_rejets( const List<Horizons>& horizons) ;
15
16         inline float pendage() const { return pendage_ ;}
17         inline float rejet(int i) const { return rejets_[i] ;}
18         inline void stocke_rejet(int i, float rejet) ;
19
20         private:
21             float pendage_ ;
22             float* rejets_ ;
23     } ;
24
25     //-----
26     inline void Faille::stocke_rejet(int i, float rejet)
27     {
28         rejets_[i] = rejet ;
29     }

```

Dans l'exemple de code 1.3, le fait de déclarer la classe faille "*public*" vis à vis de la classe surface, lui fait directement hériter de toutes les définitions précédentes. Ainsi, un objet de type Faille peut appeler la fonction "*dessine()*" de la classe Surface sans que celle-ci n'ait été redéfinie pour la classe Faille.

Les classes dérivées peuvent en outre contenir de nouveaux champs, comme ici le pendage global de la faille ou le rejet en chacun des points.

De même, de nouvelles fonctionnalités peuvent être définies comme le calcul de rejets à partir d'une liste d'horizons (fonction *calcule\_rejets(...)* ligne 14).

---

1. Remarque: Cet exemple de code n'est donné que pour l'exemple et ne correspond en rien à la définition de l'objet Faille sur lequel nous avons travaillé et qui sera présenté dans le chapitre 4 (voir «Paramétrisation de l'objet faille» page 75).

Un autre avantage de la notion d'héritage, est de faire bénéficier d'une nouvelle fonctionnalité toutes les classes dérivées d'une même classe parente lorsque cette fonction est ajoutée à la classe parente (voir l'exemple de code 1..4).

*Exemple de code 1.4. héritage de nouvelles fonctionnalités*

```
1  class Surface
2      {
3      public:
4          // Constructeurs
5          [...]
6
7          // Destructeur
8          [...]
9
10         // Fonctions membres
11         [...]
12         inline float age() const { return age_; }
13         inline void donne_age( float age) { age_ = age ;}
14
15
16     private:
17         [...]
18         float age_ ;
19     } ;
```

Ici, toutes les classes dérivées de la classe Surface, peuvent stocker leur âge géologique, variable qui est très importante lors de la création de modèles structuraux. Le stockage de ce nouveau champ est automatique pour toutes les classes dérivées, par simple héritage, sans avoir à modifier le code de toutes ces classes.

Ceci a pour effet une forte réduction de lignes de code et favorise la maintenance du code, puisque qu'une erreur corrigée au niveau de la classe parente est automatiquement répercutée dans toutes les classes dérivées, sans risque d'oublier de corriger une des classes, comme dans des langages non orientés objet.

On comprend qu'avec l'utilisation de telles notions, l'ajout d'un module "faille" dans un logiciel existant permet de bénéficier de beaucoup de structures définies au cours des 10 dernières années. Comme il le sera détaillé dans le chapitre 4, l'objet spécifique sur lequel nous avons travaillé pour modéliser les incertitudes des failles s'appuie très fortement sur les lignes polygonales et bénéficie ainsi de très nombreuses fonctionnalités d'édition.

La description de certains autres avantages apportés par le langage C++ pour ce travail de thèse seront détaillés plus tard. De plus, une présentation plus détaillée des potentialités de ce langage pourra être trouvée dans [8 - Booch], [9 - Ellis], [10 - Stroustrup].

### 3.2 Autres avantages plus spécifiques à gOcad

\* La plupart des étapes de la chaîne d'exploration peuvent être faites dans le même logiciel, de la construction du modèle de vitesse à l'étude des propriétés physiques des différentes couches, en passant par le modèle structural. En effet, il est apparu de plus en plus que partager le même modèle lors des différentes phases de l'exploration est indispensable (gain de temps d'une part puisque les données n'ont plus à être transformées d'un format à un autre, mais surtout meilleure vérification de la corrélation du modèle avec les données initiales à chacune des étapes) [5 - Duey], [16 - Schenck], [17 - Tipee], [88 - Deutsch 1].

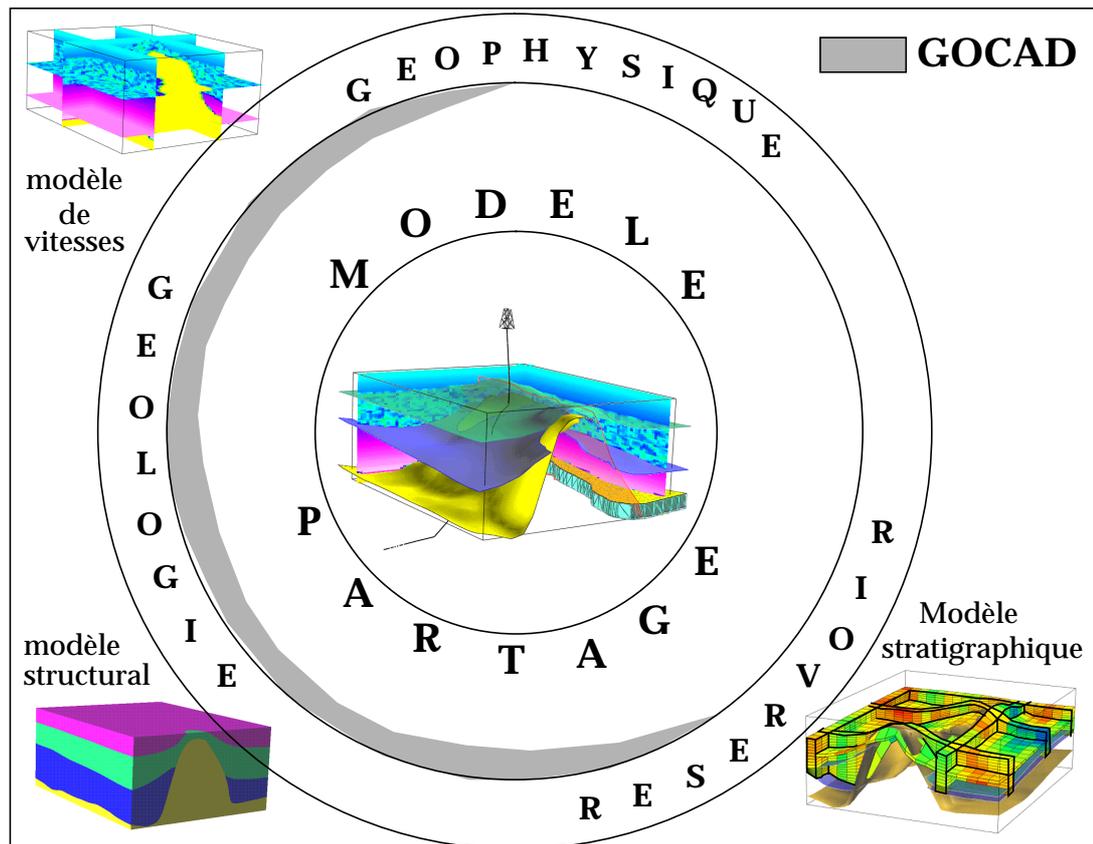


Figure 1-7 : Le modèle doit être partagé par les différentes équipes afin de réduire les incertitudes et le temps de modélisation (d'après [17 - Tipee]).

\* Les méthodes statistiques déjà implantées sont nombreuses et permettent d'axer le travail sur le choix des méthodes et le réglage des différents paramètres plutôt que sur l'implantation d'une méthode en particulier. En effet, la plupart des méthodes géostatistiques présentes dans la GSLib de Stanford [89 - Deutsch 2] ont été reportées dans gOcad.

#### \* Système de contraintes non figé

La liste de contraintes DSI présentée précédemment (voir p 14) est non exhaustive: dès qu'une contrainte sur la géométrie des failles peut être décrite à l'aide d'une équation linéaire [6 - Mallet 1], cette contrainte peut être facilement ajoutée au processus de simulation, sans remettre en cause la méthodologie globale.

#### \* Grilles stratigraphiques faillées

Un des sujets de recherche importants actuellement en géosciences concerne la construction de grilles faillées. En effet, il est important de pouvoir fournir au simulateurs d'écoulement des grilles suffisamment précises au niveau des failles et répondant à des critères bien définis en terme de topologie et de géométrie.

Ce sujet de recherche est un des sujets prioritaires actuellement dans gOcad et a permis d'obtenir des résultats prometteurs dernièrement [112 - Cognot 3], [118 - Levy 3]. Ces avancées laissent espérer une intégration plus rapide de ce travail dans le logiciel JACTA.

## 4. La compagnie T-Surf

En 1997, une compagnie a vu le jour afin de s'occuper de la partie commerciale du logiciel gOcad, le fonctionnement sous forme de consortium devenant difficile à gérer avec autant de compagnies (28 compagnies pétrolières ou minières et 45 universités sont membres du consortium de recherche).

Cette compagnie, dirigée par Jean-Claude Dulac, est basée sur deux sites: une partie travaille à Nancy, donc de façon très rapprochée avec l'équipe recherche, tandis qu'une autre partie est basée à Houston, où la plupart des compagnies pétrolières sont représentées. C'est ainsi que, durant cette thèse, j'ai été amenée à faire de nombreux séjours à Houston, afin de travailler en collaboration avec Jean-Claude Dulac.

Cette collaboration avait comme principaux avantages:

- Profiter de sa connaissance globale du logiciel et du langage C++. Ceci a permis une décision plus rapide des structures de données nécessaires à la réalisation de ce travail et un suivi très régulier de l'avancement de l'implantation de ce travail.
- Confronter les méthodes de modélisation de diverses compagnies. Ainsi, même si ce travail a été financé par Elf Exploration Production et correspond à leur méthodologie, il s'inscrit dans le projet plus global et peut être utilisé avec d'autres méthodologies. Les discussions avec des géologues structuraux d'autres compagnies pétrolières (notamment *Don Medwedeff* de la compagnie ARCO a Dallas), m'ont permis de découvrir d'autres aspects de la modélisation, et d'avoir une vision plus large du problème.
- Enfin et surtout, ces séjours ont permis de commencer une collaboration étroite avec Isabelle Duvinage, autre étudiante en thèse au LIAD, qui a commencé sa thèse à Houston. Isabelle travaille sur la construction automatique de réseaux de failles à partir d'horizons interprétés [103 - Duvinage 3]. Nous avons travaillé sur une structure de données commune, qui sera présentée dans le chapitre 7. Cette structure appelée «Structural Model» pour modèle structural facilite énormément la gestion des contacts entre les différentes surfaces du modèle lorsque celui-ci devient complexe.



---

---

## Chapitre 2 Modélisation des failles: techniques disponibles

---

---

La connaissance de la géométrie précise des failles est importante, non seulement pour construire un modèle géologique représentant au mieux la réalité, mais surtout pour pouvoir évaluer correctement les réserves potentielles d'un réservoir. En effet, elles constituent l'un des types de pièges à hydrocarbures les plus importants et la reconnaissance de telles structures est souvent une des clés du succès [47 - Sheriff]. Malheureusement, les zones faillées sont souvent les plus mauvaises sur les sections sismiques et ne permettent pas une détermination correcte de leur géométrie [16 - Schenck]. Il faut souvent utiliser des notions plus **conceptuelles** sur le «*style tectonique*» de la région qui permet d'éliminer certaines formes de failles inacceptables du point de vue géologique [129 - Walsh 4].

La première partie de la thèse a été consacrée à une *étude bibliographique*, ayant pour but la détermination des paramètres importants à étudier lors de la modélisation des failles; puis lors de la modification de leur forme au cours des simulations. Ce chapitre résume les articles lus sur différentes méthodes utilisées pour étudier la géométrie des failles (principalement en contexte extensif). Ces techniques le plus souvent déterministes, pourraient servir à la construction d'une première surface de faille "réaliste". D'autre part, elles pourraient aider à déterminer des paramètres caractérisant cette faille et permettant le contrôle des simulations obtenues par la suite de telle sorte qu'elles ne s'éloignent pas trop de cette solution initiale plausible. Cependant, nous verrons au cours de ce chapitre qu'il est très difficile d'obtenir des règles précises concernant la forme des failles, celle-ci dépendant beaucoup du cas étudié.

Une première partie décrit brièvement **à quoi correspond une faille** et précise les principaux termes qui seront employés par la suite. Ce glossaire est suivi d'un *catalogue des différents styles tectoniques* qui montre à quel point l'interprétation d'une faille nécessite une étude au cas par cas et peut difficilement être automatisée. Malgré l'effort de classification effectué [25 - Davis], [27 - Foucault], [30 - Gidon], les styles de base sont relativement nombreux. De plus il est souvent difficile de déterminer quel type de faille correspond à la zone étudiée puisque dans les zones profondes (donc anciennes) et complexes, plusieurs styles de base peuvent être mélangés, la tectonique peut avoir varié au cours du temps... C'est pourquoi les géologues structuralistes sont actuellement sceptiques sur une automatisation de l'interprétation des failles.

Dans une deuxième partie, une méthode permettant de **déterminer la forme la plus probable d'une faille, en utilisant le pointé d'au moins 2 horizons** (plus on utilise d'horizons, meilleur est le résultat) est décrite. Cette technique utilise très peu de données initiales (les semis de points des horizons [au moins 2], la donnée de la géologie régionale avant la déformation, ainsi que la valeur du rejet horizontal [celui-ci pouvant être estimé à partir des semis de points]). Elle utilise la conservation de la masse lors de la déformation, et peut même tenir compte de la compaction. Cependant nous expliquerons pourquoi cette technique n'a pas été employée dans ce travail de thèse.

La conclusion de ce chapitre décrit les choix qui ont été faits à partir du type de données qui nous sont fournies, et ce qui peut être envisagé dans un avenir proche, en utilisant ces techniques en pleine évolution.

## 1. Qu'est-ce qu'une faille?

L'appellation de faille provient de l'ancien français "*faillir*" (manquer), parce que le mineur était souvent incapable de retrouver le filon ou la couche qu'il exploitait après la traversée d'une telle structure [27 - Foucault]. On parle de faille dès qu'une déformation cisailante provoque une cassure de terrain accompagné d'un *déplacement relatif* des parties séparées (comportement fragile des roches, par opposition au comportement ductile créant une simple déformation des couches). Leur caractère fondamental est d'introduire un décalage brutal des repères dans la masse rocheuse [30 - Gidon].

En ce qui concerne l'étude réservoir, les failles ont un rôle important puisqu'elles *délimitent les différents blocs* du réservoir, agissant directement sur la volumétrie des réservoirs potentiels. Elles interviennent aussi dans la circulation des fluides, puisqu'elles peuvent soit constituer des *couloirs* où les fluides se déplacent, soit former des *barrières* en mettant en contact des zones perméables avec des zones imperméables [22 - Abrahamsen], [34 - Hardman].

### 1.1 Généralités

#### 1.1.1 Nomenclature: quelques mots clés

Les deux parties séparées par la faille sont appelés **compartiments** (le *mur* situé sous la faille et le *toit* la surmontant, lorsque la surface de faille est inclinée). Les surfaces engendrées par les cassures sont appelées les **lèvres** de la faille. Lorsque ces dernières sont polies par le frottement des deux compartiments, elles donnent lorsque l'érosion les dégage un **miroir de faille**, sur lequel on peut voir des **stries** qui donnent la direction du mouvement.

Lorsque l'on étudie une faille, divers éléments caractéristiques peuvent être mesurés [figure 2-1]:

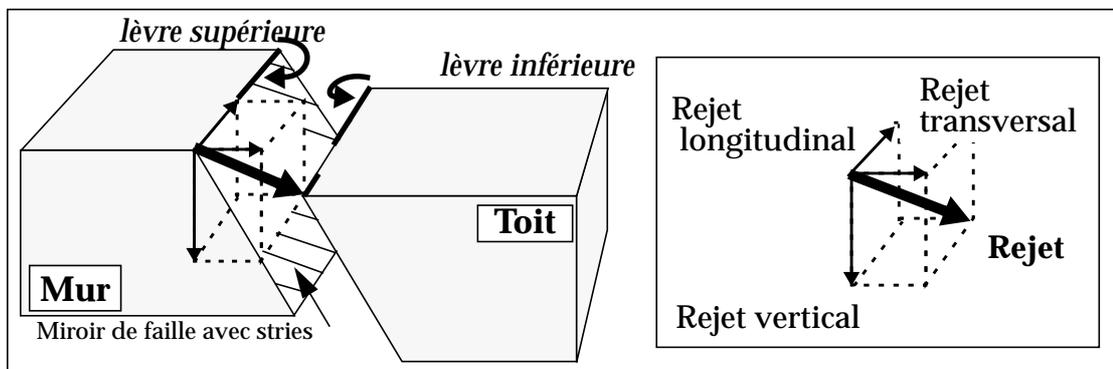


Figure 2-1 : Nomenclature servant à décrire la géométrie d'une faille (d'après [27 - Foucault])

- **le rejet**: il correspond à la valeur du déplacement entre deux points qui étaient situés à la même position avant l'apparition de la faille. Le rejet est variable le long de la faille et peut varier du décimètre à plusieurs kilomètres verticalement, et de plusieurs dizaines à plusieurs centaines de kilomètres horizontalement.

Il peut être considéré comme la somme vectorielle de trois composantes orthogonales: l'une, horizontale et située dans le plan de la faille, correspond au rejet horizontal longitudinal ou décrochement; la seconde, elle aussi horizontale mais perpendiculaire à la précédente, correspond au rejet horizontal transversal (raccourcissement ou distension suivant les cas); la troisième, verticale, s'appelle le rejet vertical.

Les rejets importants correspondent à une succession d'épisodes tectoniques dans la zone considérée, et il est souvent difficile d'évaluer non seulement l'ampleur des mouvements, mais aussi de comprendre les différentes étapes de l'histoire tectonique de la zone.

- **le regard:** côté vers lequel est tournée la lèvre du compartiment soulevé.  
Le sens du rejet des failles peut être connu par différentes observations [figure 2-2]:
  - soit la mise en évidence du *décalage des compartiments* lorsque l'on a des repères suffisamment reconnaissables
  - soit l'observation de la *direction des stries* provoquées par le déplacement des débris sur les lèvres et qui s'interrompent brusquement
  - soit la *crystallisation de calcite* dans les cavités laissées à la surface de la faille lors d'une extension (zones de faible pression).
  - soit l'apparition de *stylolites* qui se forment dans le cas d'une compression.

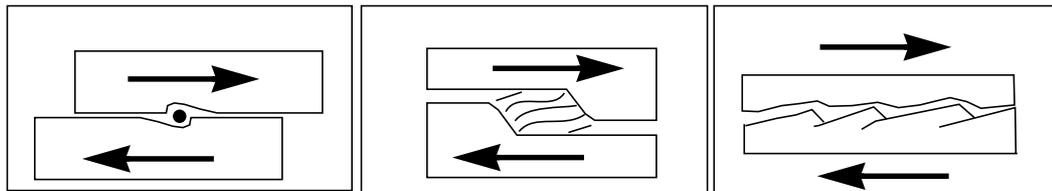
Cependant, dans notre cas, ces signes ne sont que très rarement observables.

#### Vue en coupe

Stries de faille/  
marques

recristallisation  
de calcite

Ecailles



#### Vue en trois dimensions

Stries de faille/  
marques

recristallisation  
de calcite

Ecailles

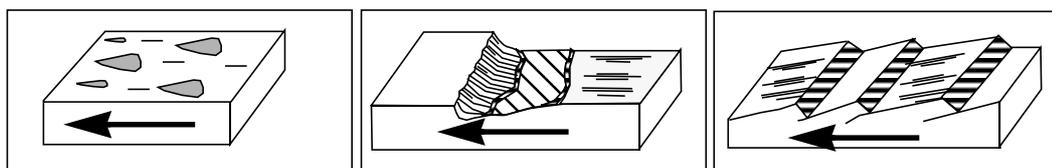


Figure 2-2 : Observations aidant à déterminer le mouvement d'une faille (non exhaustif) (d'après [27 - Foucault]).

- **le pendage:** c'est l'angle que fait le plan moyen de la faille avec un plan horizontal. Lorsque l'on observe la faille sur une coupe verticale, il faut faire attention au fait que l'on ne mesure, dans le cas général, que le *pendage apparent*, généralement plus faible que le pendage réel. Cependant, si la coupe est perpendiculaire à la faille, le pendage apparent est égal au pendage réel. Cette propriété a été étudiée notamment par Walsh et Watterson [54 - Walsh 1].

### 1.1.2 Les différentes classifications des failles

Il existe de très nombreuses classifications des failles en fonction des critères suivants [figure 2-3]:

- **le pendage [A]:** failles *verticales* (1) ou *obliques* (2)

- **le rejet [B]:** faille *normale*, *inverse* ou de *décrochement*

Une faille normale (3) est une faille à rejet vertical prédominant, et dont le plongement est dans le sens du compartiment abaissé.

Une faille inverse (4) est une faille à rejet vertical prédominant et dont le plongement est dans le sens opposé à celui du compartiment abaissé.

Pour les failles en décrochement, le rejet est uniquement horizontal, et dans le plan de la faille (qui est quasiment vertical dans ce cas). Ce décrochement peut être *dextre* (5) ou *senestre* (6).

- **selon les rapports avec les couches [C]:** faille *conforme* ou *contraire* (elles peuvent être soit *normales*, soit *inverses*).

Une faille conforme est inclinée dans le même sens que les couches qu'elle affecte (7: normale, 9: inverse), alors qu'une faille contraire est inclinée dans le sens contraire (8: normale, 10: inverse).

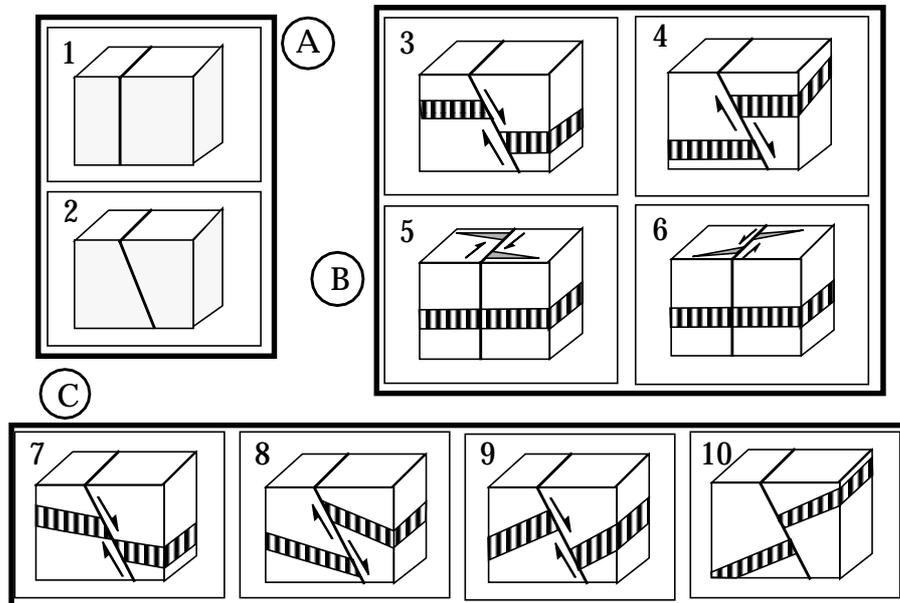


Figure 2-3 : Différents critères pour classer les failles (d'après [27 - Foucault])

- **le rapport du rejet avec d'autres déplacements [figure 2-4]:** failles *synthétiques* (même sens que la faille Majeure) ou *antithétiques* (sens opposé).

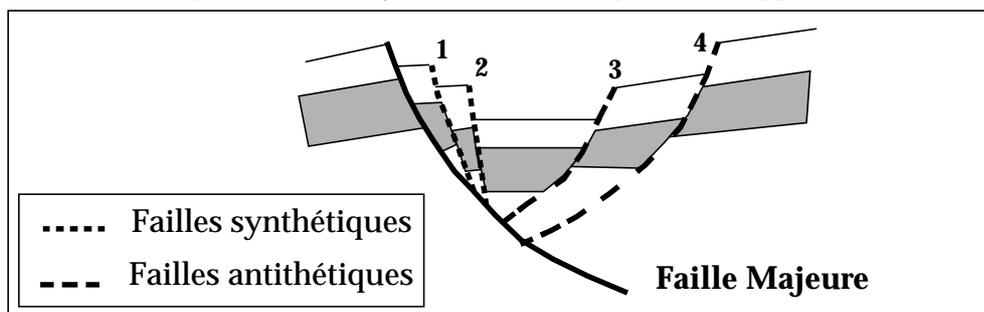


Figure 2-4 : Les failles synthétiques (1 et 2) et antithétiques (3 et 4) d'une faille principale.

### 1.1.3 Les différentes échelles d'étude

Nous avons travaillé à une échelle très particulière: l'*échelle du réservoir*. L'extension des failles pouvant varier de quelques mètres à plusieurs dizaines voire centaines de kilomètres, ce paragraphe permet de situer notre échelle de travail dans un contexte [22 - Abrahamsen] plus général. Parmi les différents types de failles, on distingue [22 - Abrahamsen]:

- **les fractures:** elles ne sont pas considérées comme des failles à proprement parler, puisqu'elles ne provoquent pas de déplacement des terrains affectés. Par contre, en géologie pétrolière, elles jouent un rôle fondamental dans l'écoulement des fluides, notamment dans les couches carbonatées. C'est pourquoi leur répartition est abondamment étudiée [33 - Gringarten], [24 - Cacas]. Elles apparaissent lorsque la roche est relativement homogène.
- **les failles «invisibles» (ou *sub-sismiques*).** La longueur et le rejet de ces failles est très difficile à estimer, puisque l'on considère que ces failles ont un rejet vertical inférieur à 5m. Comme les fractures, elles ont un rôle très important dans l'*écoulement des fluides* et bien que difficilement repérables, elles sont très étudiées. De nombreux articles établissent des règles sur leur répartition en fonction de l'étude des failles de plus grande échelle, en utilisant des dimensions fractales par exemple [41 - Munthe], [43 - Omre], [55 - Yielding 2].
- **les failles visibles sur la sismique.** Notre travail s'applique à ce type de failles. On considère qu'elles ont un rejet pouvant aller de quelques mètres à une centaine de mètres, et que leur longueur varie de 500 mètres à 10 kilomètres environ. Cette catégorie de failles intervient à la fois dans la compartimentation du réservoir et dans la circulation des fluides.
- les grandes **failles sismiquement actives:** qui peuvent atteindre plusieurs centaines de kilomètres.

**Il est à souligner qu'une faille correspond souvent à une ZONE BROYÉE (zone de cisaillement).** Cependant, à l'échelle à laquelle nous nous plaçons, on peut la considérer comme une surface (au moins dans un premier temps).

### 1.1.4 Les différents types de pièges associés aux failles

Le pétrole est formé à partir de matières organiques enfouies rapidement en milieu anaérobie dans une roche mère. Il ne peut être conservé que s'il est piégé dans une *roche réservoir* (ou roche magasin) suffisamment poreuse (calcaire, sable,...) et surmontée d'une couche imperméable (argile, sel,...) dans des structures adéquates appelées *pièges*.

Ces pièges sont de différentes natures (anticlinaux, dômes de sel, pièges stratigraphiques...) [47 - Sheriff]. Les failles interviennent souvent dans la formation des pièges et sont connues pour être l'un des facteurs déterminants dans la formation des pièges les plus importants du point de vue économique. On distingue différents types de pièges liés aux failles [figure 2-5] [34 - Hardman]:

- Les pièges situés dans le **compartiment du toit** de la faille [a]. C'est le piège le plus classiquement connu. Il apparaît lorsqu'une couche perméable surmontée d'une couche imperméable se retrouve en contact avec cette même couche imperméable après le mouvement d'une faille.
- Les pièges situés dans le **compartiment du mur** de la faille [b]. Ces pièges ont longtemps été considérés comme secondaires par rapport à la première catégorie, mais il arrive que certains ne soient pas si négligeables.

- Les pièges dus à la **fermeture hermétique** de la faille. Même lorsque deux couches perméables sont mises en contact de part et d'autre de la faille, il arrive que les hydrocarbures soient piégés à ce niveau, du fait de la différence de taille entre les grains de part et d'autre [c]. Ce phénomène apparaît aussi lorsque les argiles situés sous les couches perméables sont entraînés le long du plan de faille, déposant ainsi une couche imperméable le long de la faille. On mesure le taux d'argiles entraînées par un coefficient: le "*smear-gouge ratio*" [d] [137 - Knipe 2], [144 - Yielding 3].

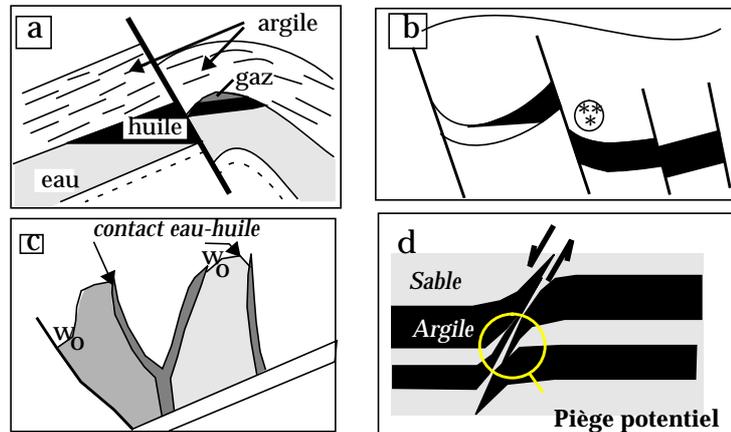


Figure 2-5 : Les grands types de pièges associés aux failles normales (d'après [34 - Hardman]). [a]: piège dans le mur de la faille; [b]: piège dans le toit; [c]: piège du à la fermeture hermétique de la faille; [d]: "smear-gouge" (d'après [144 - Yielding 3]).

### 1.1.5 Différentes techniques d'étude

Pour étudier les failles, différentes techniques sont utilisées par les géologues, en fonction du problème étudié ou de l'accessibilité des failles.

- La technique la plus naturelle consiste à **observer les failles directement sur un affleurement**. Les études de terrain n'est bien entendu pas possible partout, mais les informations tirées de telles études permettent de mieux comprendre les mécanismes qui sont à l'origine de la formation des failles, ainsi que la validation ou le rejets des modèles établis par d'autres techniques. C'est pour cette raison que les bassins de l'ouest des Etats-Unis, où l'érosion a dégagé les structures, sont beaucoup étudiés [figure 2-6].



Figure 2-6 : Vue du grand canyon du Colorado (image [18 - WWW 1])

Cependant, l'histoire de ces *failles fossiles* n'est pas toujours bien connue et de nombreuses incertitudes persistent quand à l'explication de leur formation.

- Lorsque les failles sont trop profondes, on les étudie indirectement à l'aide de méthodes géophysiques. Parmi ces méthodes, l'**étude géophysique des failles normales actuelles** présente l'intérêt d'étudier des failles qui viennent de se former et donc qui n'ont été modifiées ni par la rotation des blocs, ni par la compaction [12 - Jackson 1]. C'est ainsi que certaines règles ont pu être établies sur les pendages des failles normales: ils sont compris généralement entre 30 et 60 °, et ne sont jamais inférieurs à 20 °. Cependant, dans les cas qui nous intéressent, les failles sont rarement actives et ont subi de nombreuses transformations. Même si ces études sont satisfaisantes pour l'étude de la géométrie des failles lors de leur formation, il faut utiliser les résultats obtenus par de telles méthodes avec beaucoup de précautions dans le cadre de la géologie pétrolière.
- Une autre technique consiste à décrire un grand nombre de **sections sismiques** en deux dimensions et à en dégager des lois et des abaques [20 - Wernicke], ou plus simplement de vérifier si les modèles existants sont valides [21 - Yielding 1]. Le principal reproche que l'on peut faire à de telles techniques est que l'axe vertical de ces sections correspond au temps que mettent les ondes pour aller de l'émetteur au récepteur. Pour avoir l'équivalent de ces *images "temps"* en domaine profondeur, il faudrait donc corriger l'image obtenue par la vitesse de propagation des ondes dans les différentes couches. En effet, en raison de l'augmentation des vitesses sismiques en profondeur, une faille plane aura une allure courbe sur l'image temps. Une fois encore, il faut donc utiliser les règles établies par cette technique avec beaucoup de précautions.
- En laboratoire, de nombreux **modèles analogiques** sont étudiés. Combinés à des **études numériques**, ils permettent de valider des modèles de comportement mécanique établis par le calcul, [11 - Dresen], [13 - Kranz], [14 - Mc Clay 1], [15 - Mc Clay 2], [28 - Fougère], [51 - Vendeville 1], [52 - Vendeville 2].

Dans ces études, un matériel est placé dans un dispositif permettant de simuler les conditions de la formation des failles sous différents régimes de contraintes. Le matériel est sélectionné afin de représenter au mieux les propriétés géométriques et physiques des couches étudiées malgré l'effet d'échelle. Des bandes de couleur régulièrement espacées permettent de faire des mesures d'extension et de visualiser rapidement l'évolution des structures au cours du temps [figure 2-7].

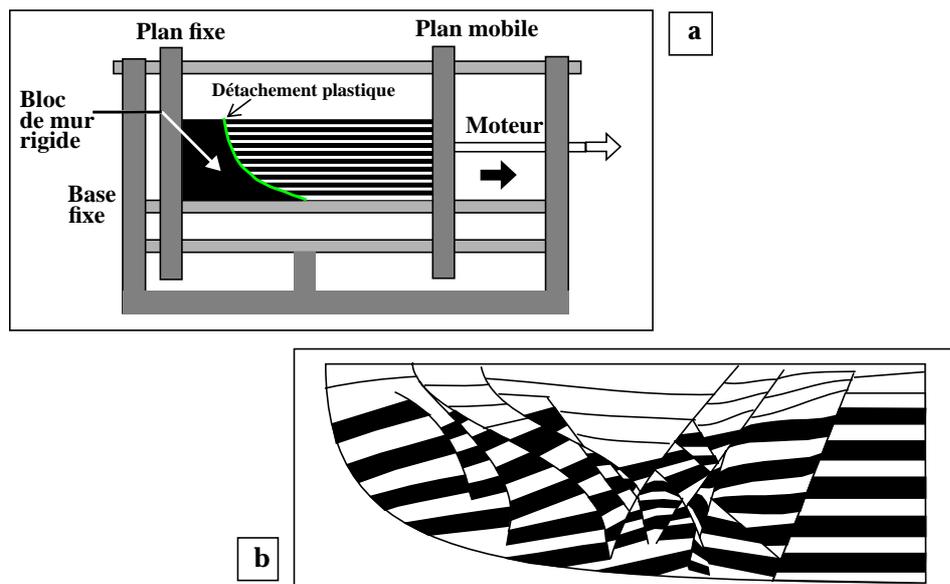


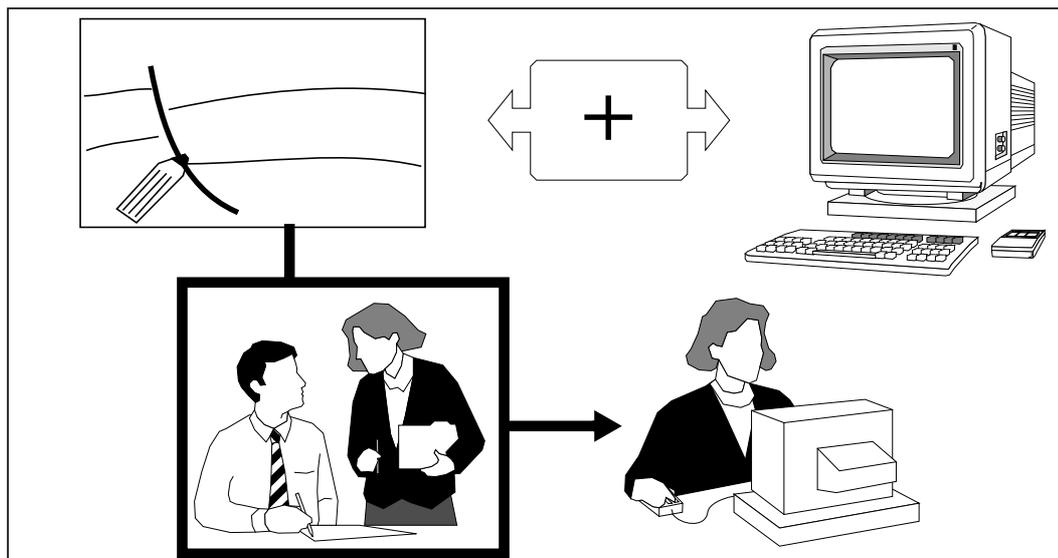
Figure 2-7 : Dispositif utilisé pour les modèles analogiques[a] et résultats obtenus [b] (librement inspiré de [14 - Mc Clay 1]).

- Plus récemment, des techniques d'**analyse d'images sismiques** sont employées pour déterminer les zones caractéristiques des sections sismiques [35 - Jones P.]. Un logiciel, initié par l'équipe de Naamen Keskes (Elf Exploration Production) sous le nom de **sismage** (et sous sa version commerciale, *stratimagic*, vendue par CGG Petrosystem) permet, entre autres, de faire apparaître les zones de failles sur les cartes d'amplitude [figure 2-10]. Ces techniques sont très performantes dans les zones les plus superficielles, les zones profondes plus bruitées donnant de moins bons résultats.

En ce qui concerne nos travaux, ces images pourraient être utiles pour *réduire la zone d'incertitude* due au pointé sismique. Cependant, il faut garder à l'esprit que parfois des artefacts apparaissent, faisant croire à la présence de failles qui en fait n'existent pas [47 - Sheriff]. De plus, les zones atteintes des plus fortes incertitudes sont les zones profondes, où les vitesses sont les plus mal connues. Or, ce sont aussi dans les zones profondes que les techniques d'analyse d'image donnent de moins bons résultats à cause du bruit qui perturbe l'image sismique.

- Enfin, de plus en plus d'articles montrent que la **modélisation directement en trois dimensions** devient de plus en plus indispensable, surtout dans les zones complexes [73 - Houlding], [135 - Hoffman], [107 - Lasseter], [109 - Orlic]. Comme nous le verrons dans le chapitre suivant, les incertitudes dues au passage de l'interprétation (2D) à la modélisation (3D) sont en effet une des grandes sources d'incertitude [121 - Freeman]. Pouvoir vérifier la cohérence de l'interprétation en trois dimensions au fur et à mesure devient une nécessité pour l'interpréteur. L'utilisation d'ordinateurs de plus en plus puissants et d'une chaîne d'outils intégrés permet de faire travailler les différentes équipes ensemble et, en leur faisant partager les résultats, de réduire les incertitudes.

Cependant, l'ordinateur ne pourra pas résoudre tous les problèmes. L'utilisation de logiciels de modélisation peut grandement faciliter le travail des structuralistes par la vision en 3 dimensions, mais comme le montre la section suivante, les styles tectoniques sont si nombreux qu'à l'heure actuelle l'automatisation semble utopique. De plus, il manque encore une communication efficace entre les logiciels d'interprétation sismique et ceux de modélisation.



**Figure 2-8 :** L'ordinateur ne doit pas remplacer le travail des structuralistes, mais les aider dans leur travail.

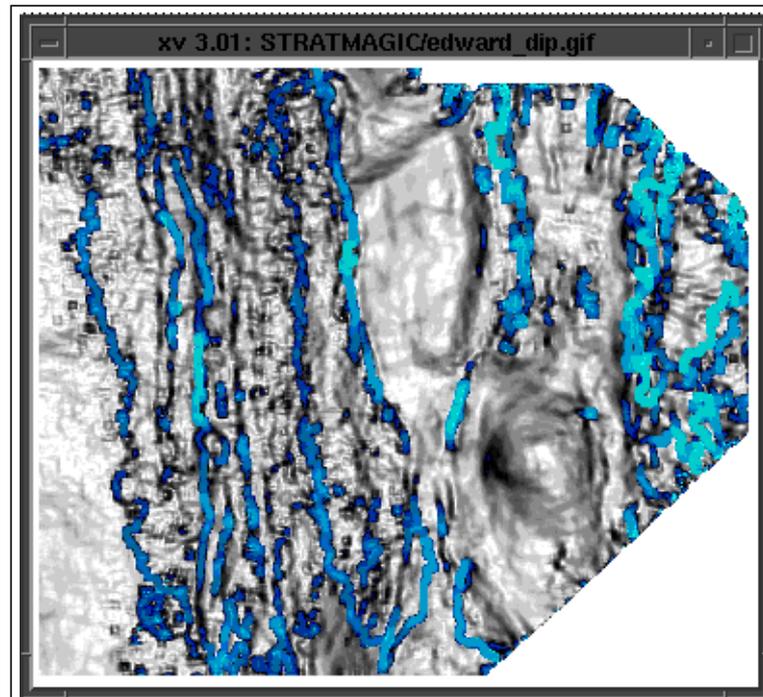


Figure 2-9 : Détection des failles (zones sombres) à partir d'analyses de pen-  
dages (avec la permission de l'équipe stratimagic: [19 - WWW 2])

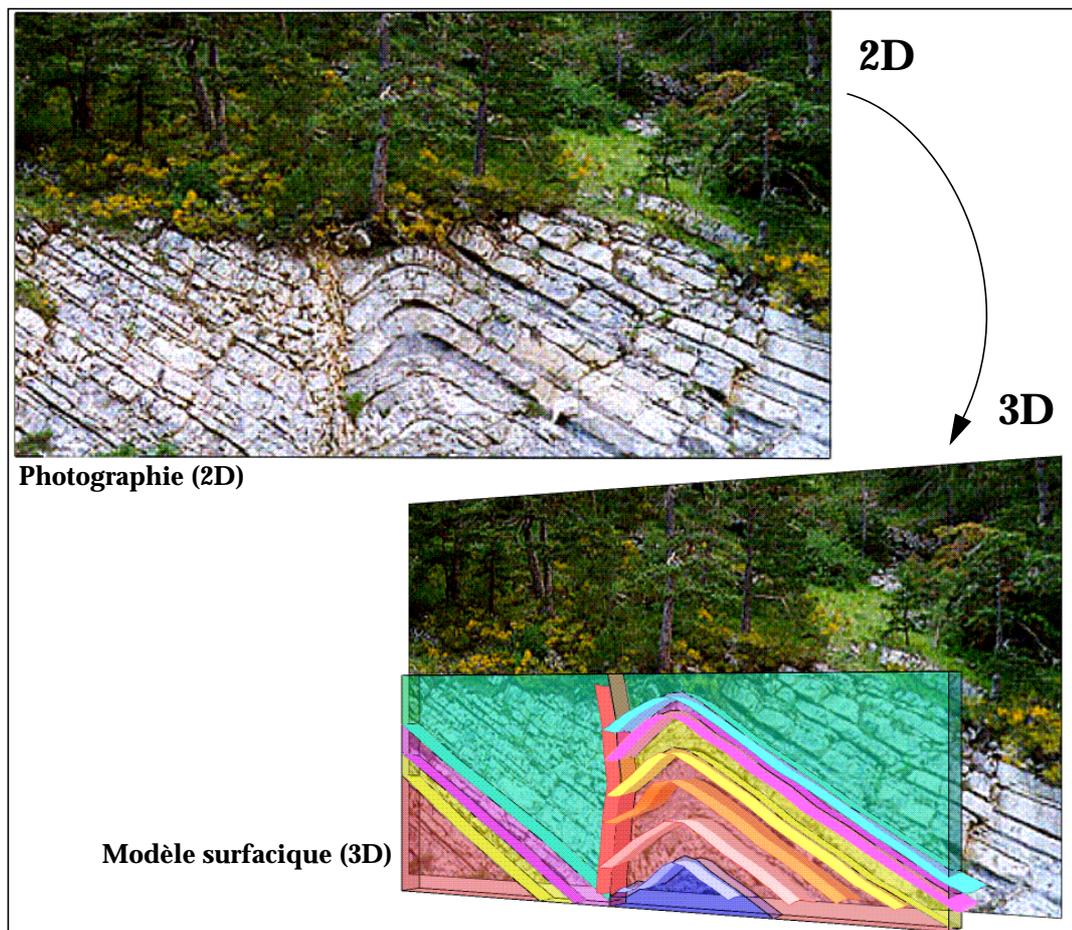


Figure 2-10 : La construction d'un modèle en trois dimensions permet de  
mieux appréhender la structure, et les contacts entre surfaces (Photo J-C. Dulac).

## 1.2 Grandes catégories de styles tectoniques en domaine extensif

*“L’observation objective n’existe pas, aucune observation ne se fait sans une référence à un concept déjà acquis”.*

Partant de cette constatation [document interne Elf Exploration Production], on ressent très vite la nécessité d’avoir un bagage étoffé de schémas à confronter à la réalité. En effet, il est reconnu que l’interpréteur doit faire preuve d’imagination lors de son interprétation [47 - Sheriff]. Cette imagination doit cependant être nourrie d’un certain nombre de styles tectoniques caractéristiques, afin de combler les lacunes des documents [32 - Grau]. La lecture de nombreux articles permet d’avoir quelques bases sur le sujet.

### 1.2.1 Horst et Graben

Les structures de Horst et Graben se retrouvent à toutes les échelles: de 5 à 100 kilomètres de large et de 5 à 30 kilomètres d’extension verticale. On appelle un Horst la partie surélevée de la structure et le graben la partie enfoncée. Ils se situent principalement dans les dispositifs des *grands lacs est africains*, dans la *Mer rouge*, ou, plus proche de nous, en *Alsace*.

On observe différents types de grabens [figure 2-11]:

- symétrique: le nombre et la répartition des failles est identique de part et d’autre de l’axe du graben.
- asymétrique: on retrouve une faille majeure sur le bord, et de nombreuses failles de l’autre côté. On observe que la somme des rejets de ces failles correspond au rejet de la faille majeure.
- asymétrique en symétrie miroir: un miroir au milieu du horst sépare 2 grabens qui sont tous les deux asymétriques.

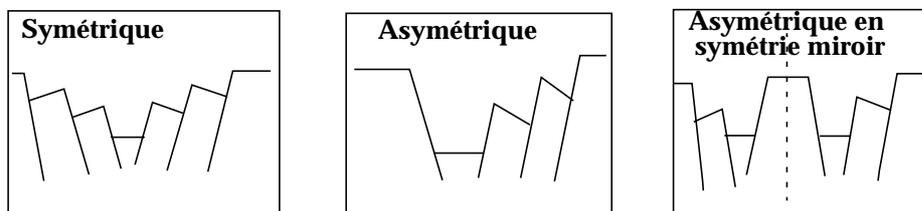


Figure 2-11 : Les différents types de Horst et Grabens

### 1.2.2 Blocs basculés

Les blocs basculés sont des structures très communes dans les domaines en extension, lorsque celle-ci est *dissymétrique*. Les géométries observées sont analogues à toutes les échelles. Toute la croûte supérieure fragile en est affectée (largeur des blocs de 5 à 20 km). A l’échelle de l’affleurement, ces failles sont rectilignes. On les trouve principalement dans la zone “*Basin and Range*” des USA et du Mexique.

Les blocs basculés sont caractérisés par des rejets équivalents sur tous les blocs et un espacement régulier des différents blocs [figure 2-12].

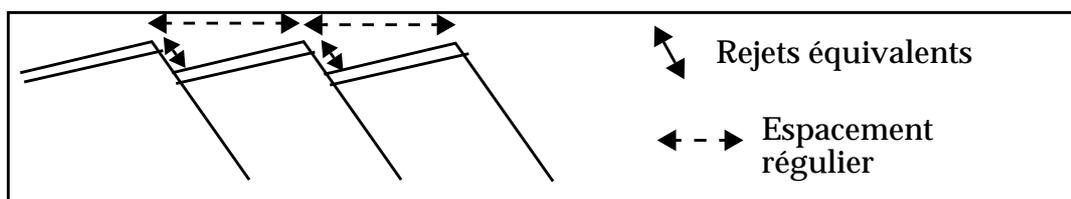


Figure 2-12 : Caractéristiques géométriques des blocs basculés.

### a) Description de la géométrie en coupe

En règle générale, on peut dire que la géométrie de la faille en coupe est fortement liée à l'épaisseur de la zone ductile à sa base.

Lorsque celle-ci est épaisse, on trouvera le **modèle en dominos**, au contraire, si la zone ductile est mince (simple décollement), on aura préférentiellement des **failles listriques**.

- Le modèle "en dominos" est constitué de blocs délimités par des failles normales planes. Dans ce cas, il existe une relation entre le pendage de la faille et la quantité d'extension [23 - Brun], [37 - Jackson 2], [figure 2-13].

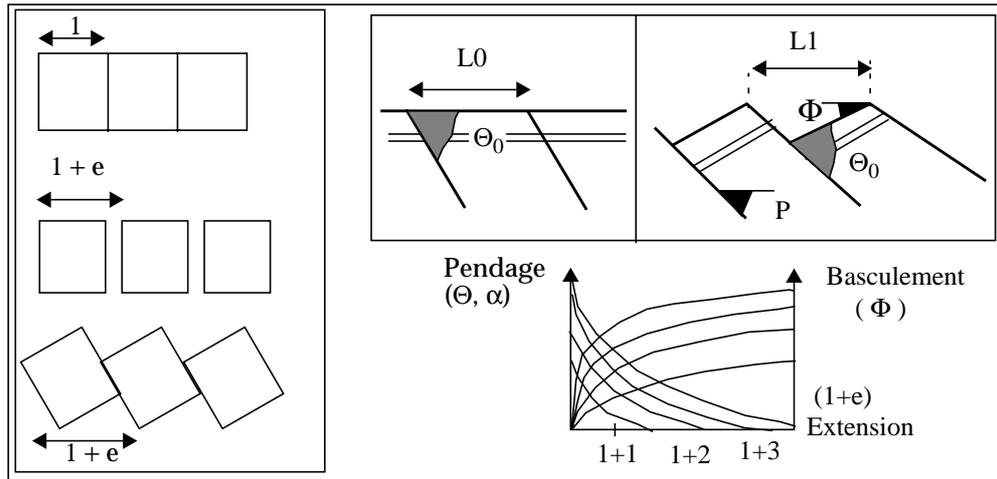


Figure 2-13 : La largeur des blocs basculés est liée au taux d'extension

Géométriquement, on obtient la relation: 
$$e = \frac{\sin \Theta_0}{\sin \Theta}$$

- Le modèle des failles listriques correspond à celui où les blocs sont délimités par des failles normales courbes qui deviennent tangentes au niveau de décollement (*failles listriques*). Le fait que les failles soient courbes laisse supposer que les blocs subissent des déformations internes.

### b) Description de la géométrie en carte

Les géométries en carte sont relativement complexes. En effet, des hétérogénéités locales peuvent modifier la forme rectiligne de failles normales [figure 2-14].

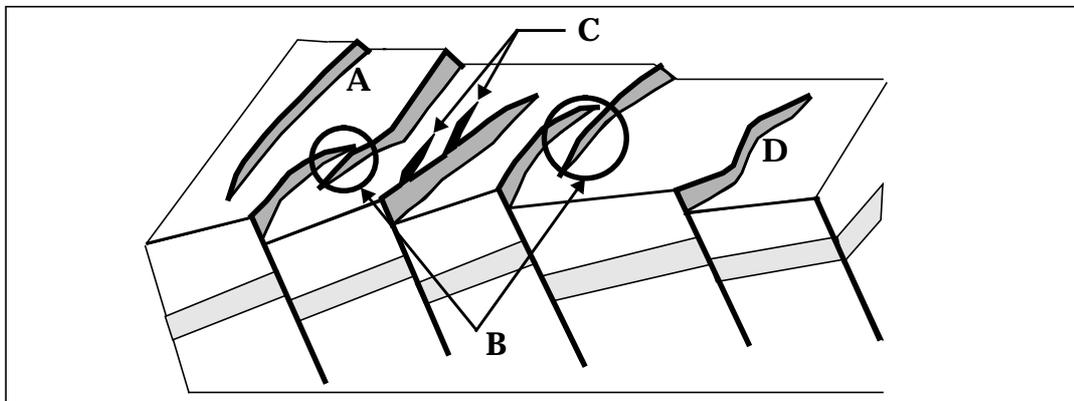


Figure 2-14 : La géométrie en carte des blocs basculés n'est pas toujours rectiligne [A] et peut être beaucoup plus complexe. [B]: relais entre failles; [C]: terminaison en échelon; [D]: tracé en zig-zag.

### c) Blocs basculés en extension oblique

Lorsqu'il existe un accident profond dans le socle ou lorsque l'on a une forme héritée d'un ancien accident, la nouvelle direction d'extension peut être perpendiculaire ou oblique à la direction de la zone déformée. De nombreuses études ont été faites sur de telles structures et les relations suivantes ont été mises en évidence [50 - Tron 2]:

- **Relation entre les composantes du mouvement et l'angle d'extension par rapport à la structure de base [figure 2-15]:**

On distingue trois grands types de mouvements en fonction de l'angle que fait l'extension avec la structure existante ( $\alpha = \text{obliquité}$ ):

1. Extension pure ( $\alpha = 90^\circ - 75^\circ$ ): lorsqu'il n'y a pas d'héritage, ou que l'extension est perpendiculaire à la direction de l'héritage. Le mouvement a uniquement une composante verticale ("*dip slip*").
2. Extension oblique ( $\alpha = 75^\circ - 30^\circ$ ): les composantes normales et décrochantes du mouvement ont des amplitudes comparables.
3. Décrochement extensif ( $\alpha = 15^\circ$ ): le glissement a une composante décrochante dominante ("*stike slip*").

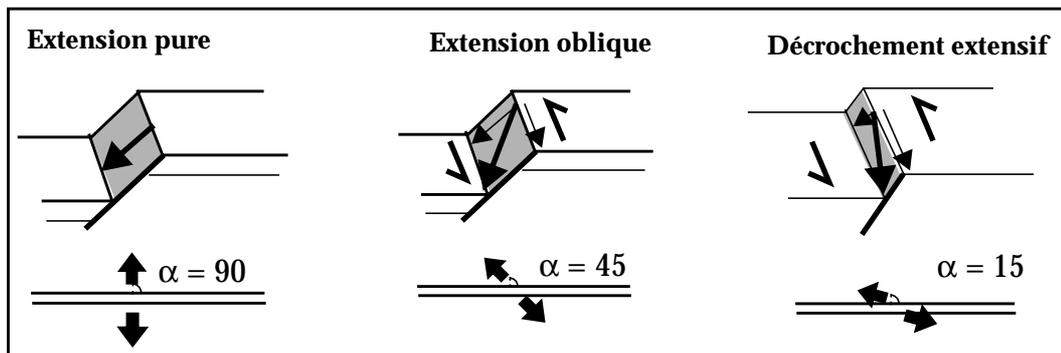


Figure 2-15 : L'angle de l'extension par rapport aux structures héritées détermine si le mouvement se fera en extension pure, oblique ou en décrochement.

- **Relation entre le pendage initial et la direction des failles:**

“Plus l'angle entre la direction de l'extension et la zone déformée diminue, et plus le pendage initial des failles se rapproche de la verticale“ [49 - Tron 1].

Ainsi, en extension pure, l'initialisation de la faille se fait avec un pendage moyen de  $60^\circ$ , alors qu'en décrochement pur, le pendage initial de la faille est proche de la verticale.

- **Relation entre l'orientation des failles et la direction de l'extension [figure 2-16].**

Lorsque l'obliquité est supérieure à  $80^\circ$ , on obtient des failles en échelon perpendiculaires à la direction d'extension.

Lorsque l'obliquité est intermédiaire ( $45^\circ - 80^\circ$ ), on obtient des failles sigmoïdes obliques à la direction d'extension. Les blocs entre les failles prennent un aspect en fuseau.

Enfin, lorsque l'obliquité est faible ( $30^\circ - 50^\circ$ ), on obtient des failles quasiment parallèles à la direction d'extension, avec une très forte composante décrochante.

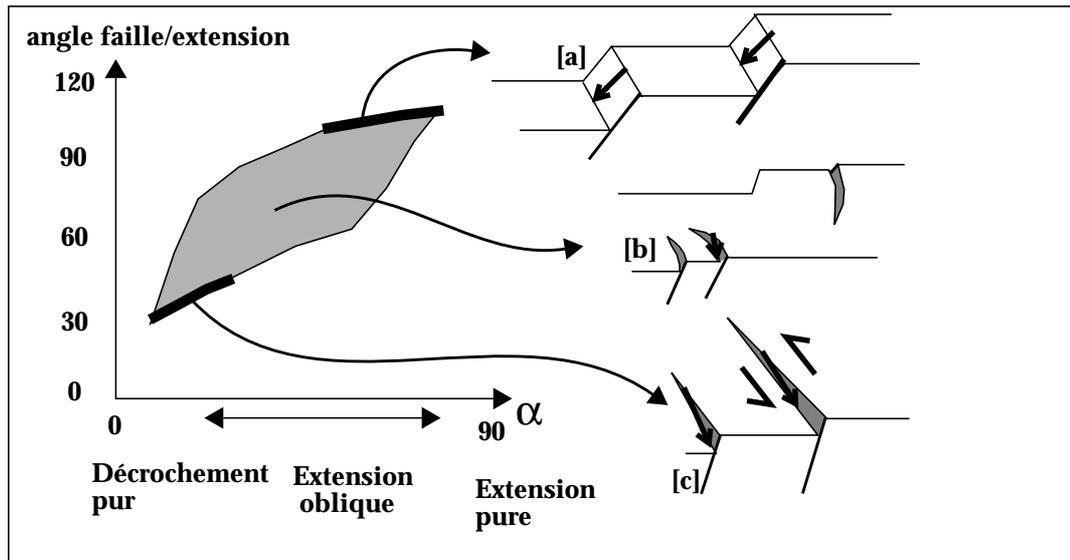


Figure 2-16 : Relation entre l'angle des failles et l'obliquité  $\alpha$  (angle entre l'extension et les structures héritées). [a]: failles normales pures; [b]: failles sigmoïdes obliques; [c]: failles décrochantes.

### 1.2.3 Blocs synthétiques

Les blocs synthétiques apparaissent toujours comme des sous-systèmes d'autres systèmes tectoniques extensifs d'échelle supérieure [figure 2-17].

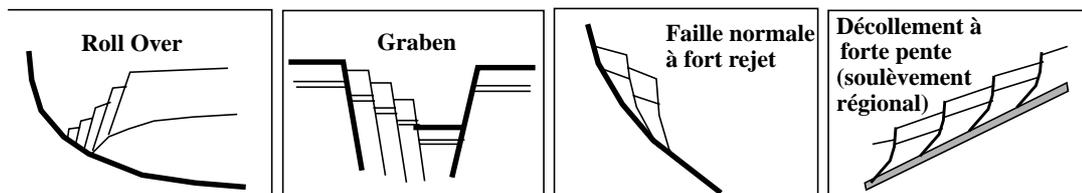


Figure 2-17 : Les blocs synthétiques sont des sous-systèmes d'autres systèmes extensifs d'échelle supérieure.

D'un point de vue mécanique, ils apparaissent lorsque le pendage des séries est incompatible avec le mouvement qu'impose le jeu des failles normales aux différents blocs.

On obtient de telles structures lorsque la rotation imposée aux blocs ne suffit pas à compenser un basculement initial important des séries, ou lorsque lors d'une déformation en cisaillement simple, la rotation est faible ou nulle (c'est à dire que l'extension est faible ou nulle).

### 1.2.4 Les failles listriques

Les failles listriques n'apparaissent que dans la croûte supérieure (10 premiers kilomètres). Elles nécessitent la présence d'une interface correspondant à un contraste lithologiques (sel, argile...).

Elles tirent leur nom du grec «listron» qui veut dire pelle. En effet, en trois dimensions, ces failles ont une forme caractéristique de cuillère, concave vers le haut [46 - Shelton] [figure 2-18] (a).

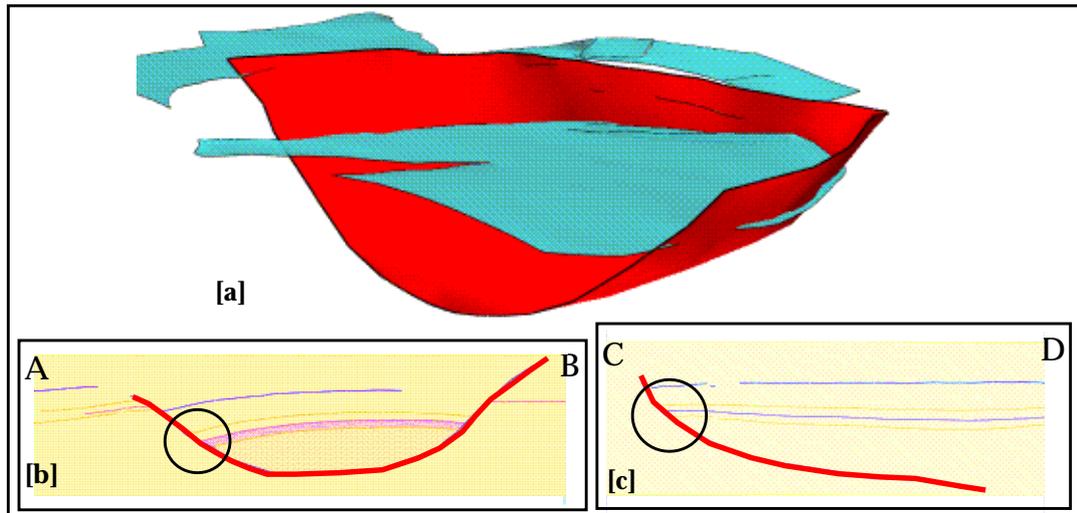


Figure 2-18 : [a]: Forme caractéristique des failles listriques (cuillère). [b] et [c]: caractéristiques en coupe: On observe généralement un fort rabattement des couches (cercles: "Roll-Over").

#### a) Caractéristiques en coupe

Les failles listriques sont généralement *régulièrement espacées* et l'on retrouve des *rejets similaires* d'une faille à l'autre [51 - Vendeville 1]. Le rejet au sommet de la faille est généralement proche de  $60^\circ$ .

On observe généralement un fort rabattement des couches à leur niveau [figure 2-18] (b) et (c) [20 - Wernicke], ainsi que des *variations d'épaisseurs* des dépôts syntectoniques.

#### b) Caractéristiques en carte

En carte, les failles listriques ont généralement une *géométrie courbe* [figure 2-19]. Elles sont disposées *perpendiculairement au sens du mouvement* le long du décollement. La concavité de la surface de faille est dirigée vers l'aval du système.

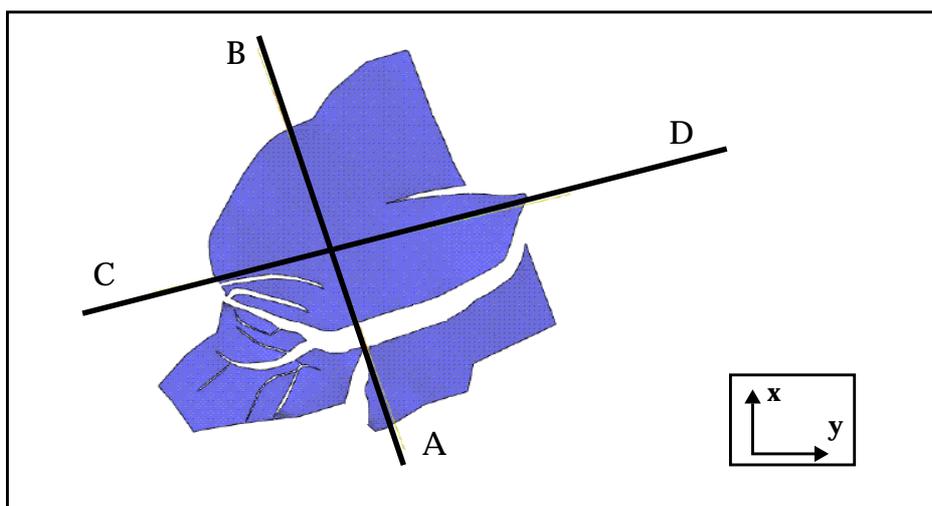


Figure 2-19 : Caractéristiques des failles listriques en carte: une forme courbe.

### c) Origine des failles listriques

On donne différentes origines possibles pour la formation de failles listriques. Ces origines peuvent être soit mécaniques soit cinématiques [26 - Dula], [46 - Shelton], [53 - Vendeville 3].

- **Variation progressive du régime des déformations en profondeur (origine mécanique)**

Dans les conditions proches de la surface, le régime de déformation est fragile (cassant). Les failles sont des plans bien individualisés et s'inclinent avec un pendage moyen de  $60^\circ$ . Par contre, en profondeur, les conditions de température et de pression sont telles que l'on passe en déformation ductile. Le cisaillement y est moins penté et plus large [figure 2-20] (a).

- **Compaction progressive des sédiments lors de l'enfouissement (origine mécanique)**

Suivant cette hypothèse, les failles sont aussi initiées à  $60^\circ$ . La compaction des sédiments en profondeur donne des pendages de plus en plus faibles pour le plan de faille [figure 2-20](b).

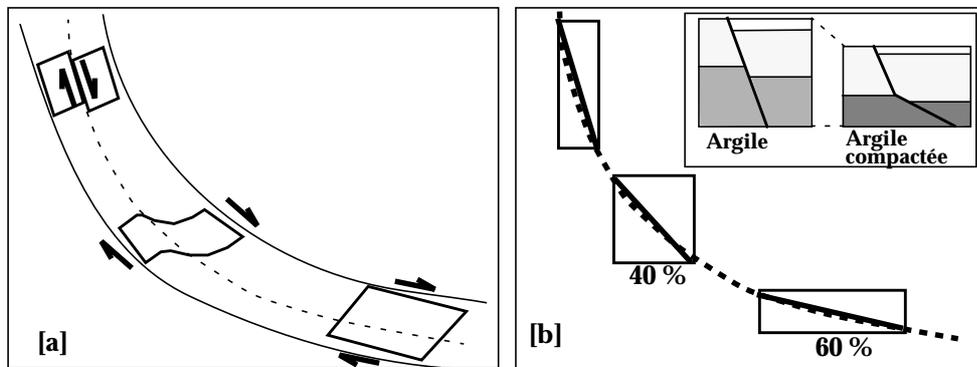


Figure 2-20 : Origines mécaniques des failles listriques. [a]: variation progressive du régime des déformations en profondeur ( $\Delta P$  et  $\Delta T$ ); [b]: compaction progressive des sédiments lors de l'enfouissement.

- **Failles de croissance (origine cinématique)**

Les failles de croissance évoluent au cours du temps en fonction du taux de sédimentation. Comme dans les deux cas précédents, elles sont initiées avec un pendage de  $60^\circ$ . Par la suite, une extension provoque un basculement (cf blocs basculés), puis de nouvelles failles sont initiées avec un pendage à  $60^\circ$ . En fin de processus, on obtient de failles listriques avec un pendage de  $60^\circ$  en surface et environ  $30^\circ$  en profondeur [figure 2-21].

Dans ces contextes gravitaires, les géométries sont variables. Elles sont contrôlées par différents paramètres dont le taux de sédimentation: plus le taux de sédimentation augmente, plus l'intensité de la déformation et donc le taux d'extension augmente, ce qui entraîne une forte rotation ainsi qu'une forte asymétrie [51 - Vendeville 1].

Une zone très étudiée pour l'exploration pétrolière se situe dans le golfe du Mexique. On trouve aussi de telles failles dans le delta du Niger.

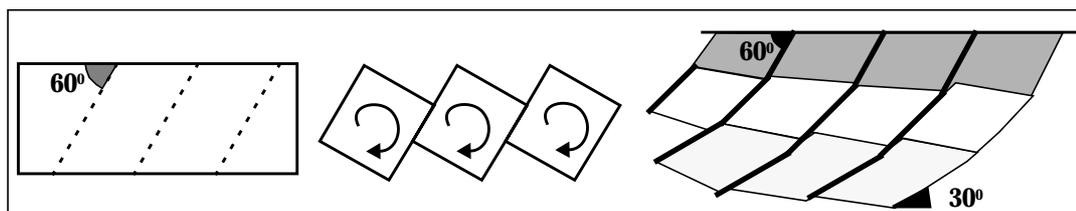


Figure 2-21 : Origine cinématique des failles de croissance

### 1.3 Autres styles tectoniques

Dans tous les exemples réels que nous avons eu à étudier, le contexte tectonique correspondait à une *extension*. C'est pourquoi l'étude des autres styles tectoniques n'a pas été poussée plus avant, d'autant plus qu'ils sont généralement encore plus complexes à décrire. Les paragraphes suivants donnent cependant quelques notions de ces styles tectoniques.

#### 1.3.1 Domaine compressif

La tectonique compressive fait son apparition lorsque la contrainte compressive prédominante est horizontale [47 - Sheriff].

En contexte compressif, l'acquisition sismique et le forage de puits coûtent extrêmement cher. C'est pourquoi il est indispensable de bien connaître les principes géométriques simples pour interpréter les données de façon optimale [35 - Jones P.].

##### a) Un peu de vocabulaire

Les failles en domaine extensif sont décrites avec un vocabulaire qui leur est propre [31 - Ghosh], [38 - Mc Clay 3], [40 - Michel].

Les **failles chevauchantes** (*thrust fault*) ou **chevauchements** (*thrust*) sont des failles inverses dont le pendage est inférieur à  $45^\circ$ . Elles placent régulièrement des terrains plus anciens sur des terrains plus récents. Ce sont des structures que l'on rencontre à grande échelle (quelques kilomètres à une dizaine de kilomètres). Elles ont été décrites pour la première fois en 1935 dans les Alpes suisses. On trouve aussi des exemples de telles failles en Ecosse ou dans les Appalaches. Elles sont donc communément rencontrées dans les zones de montagnes plissées.

Si le bloc de mur reste fixe et que la faille est due au mouvement du bloc de toit, on parle de **charriage** (*overthrust*), dans le cas contraire, on parle de **sous-charriage** (*underthrust*). Lorsque des terrains sont entraînés assez loin du fait d'un charriage, on parle de **nappe de charriage** (*thrust sheet*). Ce terme est dû au fait que les couches du toit sont très fines comparées à leur longueur et leur largeur.

Les terrains qui restent en place sont appelés *autochtones*, tandis que les roches se déplaçant sur de grandes distances sont appelées roches *allochtones*. Si le déplacement est restreint, on utilise alors le terme de terrain *parautochtone*.

Lorsqu'après érosion, des terrains allochtones se trouvent isolés au milieu des terrains sous-jacents, on est en présence d'une **Klippe** (ou lambeau de charriage). Le cas inverse est appelé une **fenêtre tectonique**.

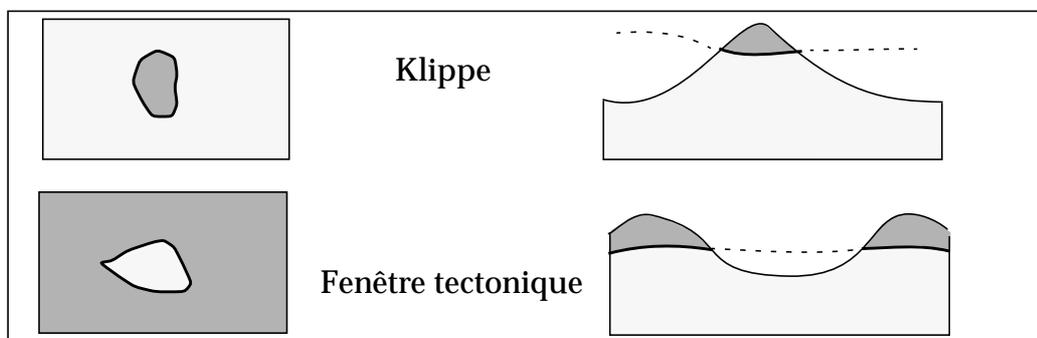


Figure 2-22 : Deux figures typiques des nappes de charriage: les Klippes et les fenêtres tectoniques.

Les **duplex**, sont constitués d'un ensemble d'écailles tectoniques imbriquées, comprises entre deux contacts anormaux majeurs de décollement. La forme des failles en contexte compressif dépend énormément de la lithologie. On distingue les **plats** (flat) au niveau des couches incompetentes horizontales (argile et sel), et les **rampes** localisées au niveau des couches compétentes (sables et calcaires) [44 - Ramsay], [figure 2-23]. Ces structures ont été beaucoup étudiées par l'équipe de John Suppe de l'université de Princeton [48 - Suppe].

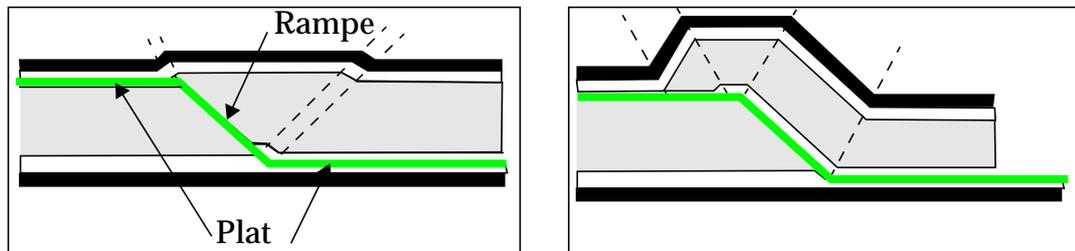


Figure 2-23 : La géométrie des failles inverses est fortement liée à la lithologie (exemple de Duplex)

Lorsque le plan de faille est parallèle aux couches stratigraphiques sur de longues distances, on obtient un **profil en escalier** (*step faults*) [figure 2-24]. La stratigraphie des couches du bloc de toit est le principal facteur qui contrôle l'emplacement des rampes successives: celles-ci se situent préférentiellement dans les niveaux résistants.

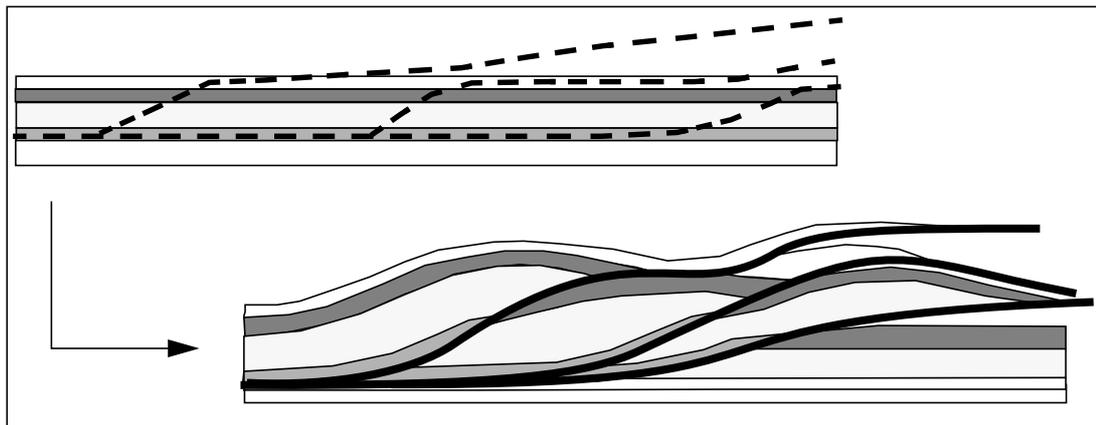


Figure 2-24 : Exemple de *step faults* (d'après [35 - Jones P.]).

## b) Les familles de failles inverses

Dans la plupart des cas, la déformation du volume rocheux induit un régime de cisaillement qui privilégie le développement de fractures synthétiques du mouvement cisailant global.

Les failles qui se forment successivement apparaissent chacune en avant de la précédente dans le sens du cisaillement, chaque cassure se formant sous celle apparue précédemment [30 - Gidon].

Lorsque le soubassement s'enfonce le long d'une discontinuité profonde, la compression horizontale dans la partie superficielle entraîne la formation de réseaux de failles inverses plus ou moins développés en fonction du taux de raccourcissement [figure 2-25]

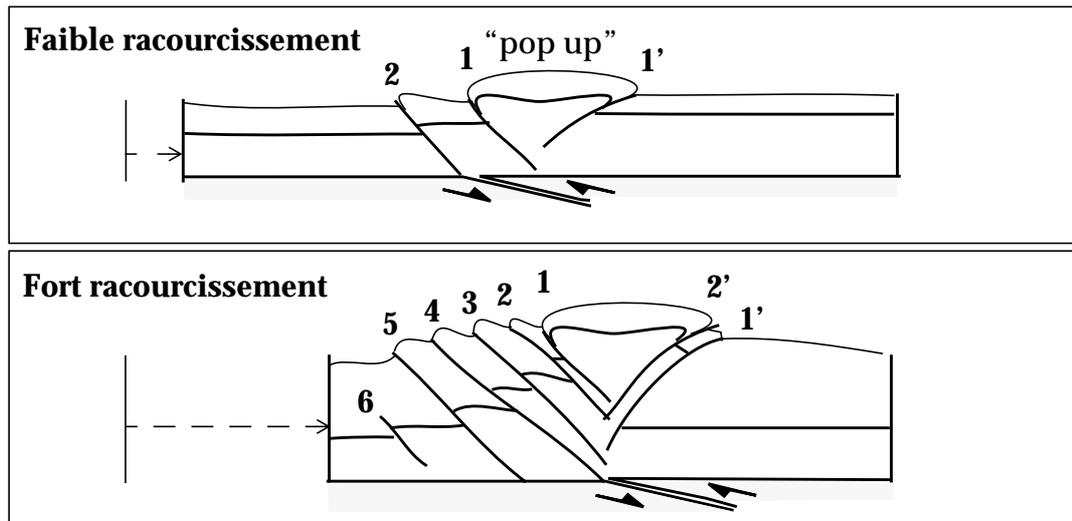


Figure 2-25 : Réseau de failles inverses. Les failles sont numérotées dans l'ordre de leur apparition (d'après [30 - Gidon]).

Il est beaucoup plus rare que les deux familles conjuguées soient également représentées. Ceci se produit lorsque le volume rocheux se déforme en aplatissement, sans cisaillement d'ensemble [30 - Gidon].

### c) Importance de la lithologie dans la formation des pièges

La lithologie influe beaucoup sur le comportement des couches aux abords d'une rampe. Ainsi, si les unités fines (argiles) se déforment facilement [figure 2-26] [a], les roches plus massives comme les carbonates ont plus de mal à se déformer et se fracturent vers le haut de la rampe [figure 2-26] [b]. C'est ainsi qu'une *alternance de couches fines et de couches massives* forme le **piège idéal** [figure 2-26] [c].

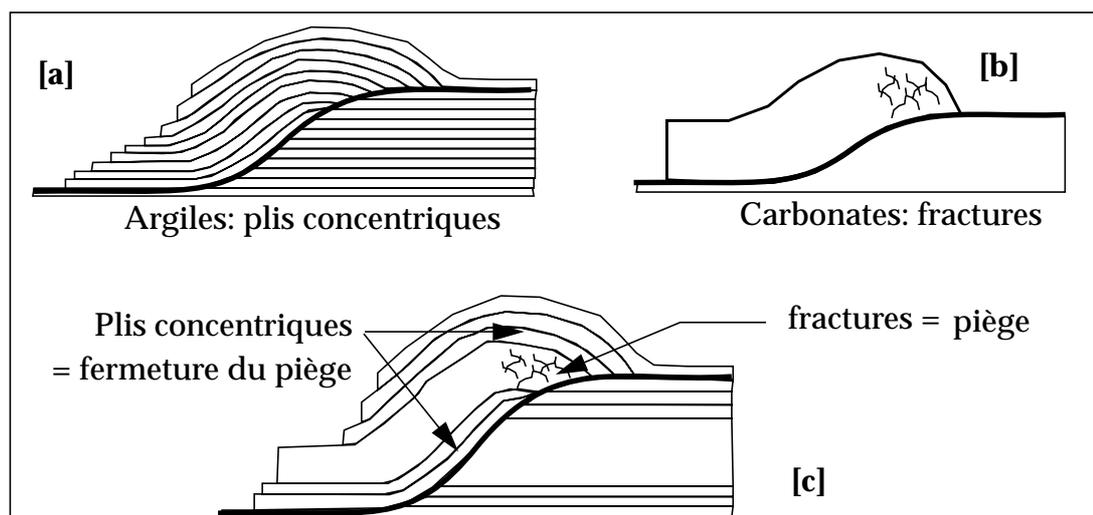


Figure 2-26 : Importance de la lithologie dans la formation des pièges ([35 - Jones P.]

Cette présentation des structures en compression présentée dans ce mémoire est loin d'être exhaustive. Plus d'informations pourront être recueillies dans [15 - Mc Clay 2].

### 1.3.2 Bassin sur décrochement

D'après les observations, on peut donner comme règle qu'un décrochement n'est *JAMAIS* rectiligne. Le long d'un décrochement, on trouve deux grands types de structures [39 - Mann], [figure 2-27]:

- les relais en **transtension** («pull apart») avec des failles normales. Ces bassins ne sont pas cylindriques et peuvent avoir différentes formes. Avec une forte composante décrochante, la déformation s'effectue dans une zone très étroite. Par contre lorsque la zone comporte une forte composante normale, le bassin est très asymétrique et beaucoup plus large.
- les relais en **transpression** avec des failles divergentes vers la surface et convergentes en profondeur, à rejet apparent inverse. A ces niveaux, on trouve souvent de grandes variations d'épaisseurs et de faciès.

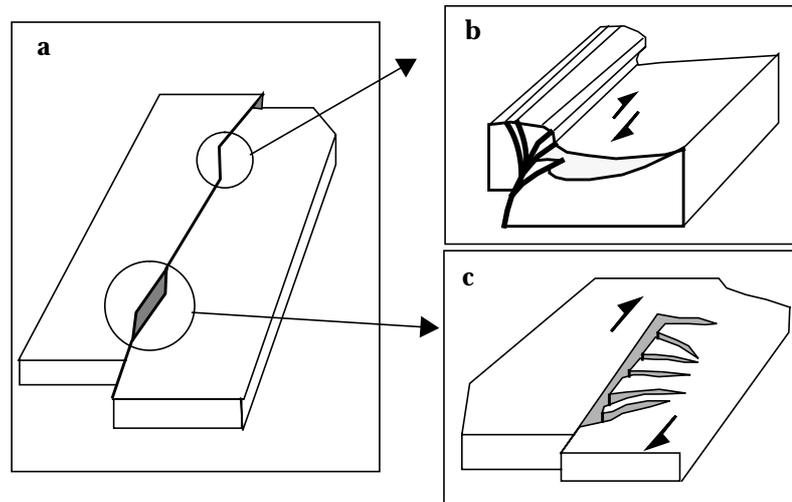


Figure 2-27 : [a]: Bassin en décrochement avec [b]: relais en transpression et [c]: relais en transtension («pull-apart») avec des grabens obliques par rapport à la direction d'extension.

Dans de telles structures, l'étude en trois dimensions est indispensable puisque les coupes ne donnent pas la composante horizontale du mouvement le long du décrochement.

De plus, au niveau d'un décrochement, on trouve généralement des failles en relais qui s'organisent de façon complexe, donnant soit des relais décrochants compressifs, soit des relais décrochants distensifs [figure 2-28] [36 - Junger].

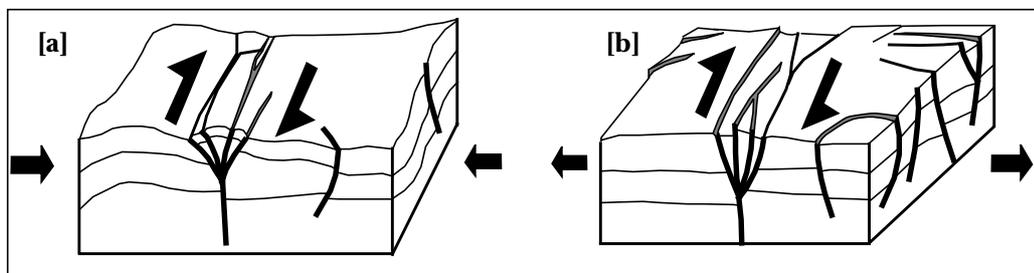


Figure 2-28 : La géométrie des décrochements en 3D peut très complexe (structures en fleur). [a]: relais décrochant compressif (fleur négative); [b]: relais décrochant distensif (fleur positive).

Très fréquemment, la géométrie de ces failles est hélicoïdale. Sur des coupes successives, une même faille peut donc présenter des vergences et des types de rejets dif-

férents [42 - Naylor] [figure 2-29] [a]. Cependant, s'il existe un niveau visqueux (sel) au dessous de la zone décrochante, les failles résultantes sont planes et sub-verticales [45 - Richard].

Le long de tels décrochements, le rejet varie énormément. En effet, le long d'un décrochement des lanières sont découpées et subissent des mouvements verticaux importants [figure 2-29] [b]. Par exemple, sur un tronçon de 10 kilomètres, pour un rejet horizontal de 200 mètres et une déformation modérée dans la lanière, on peut avoir un rejet vertical au centre de 500 mètres.

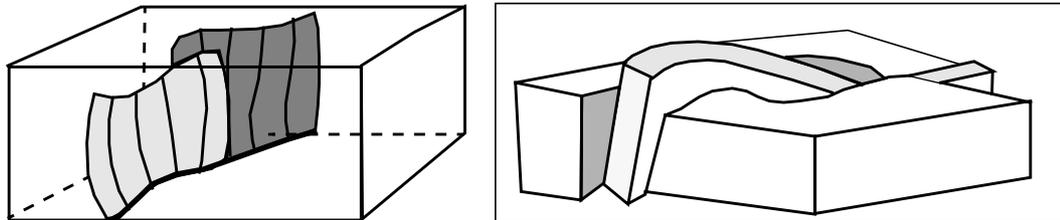


Figure 2-29 : [a]: Forme hélicoïdale fréquemment rencontrée au niveau des failles; [b]: lanière découpée le long d'un décrochement.

#### 1.4 Discussion sur les règles ainsi établies

- La plupart des sources dont ces informations sont tirées sont destinées aux interpréteurs. Une grande partie de ces règles a donc été établie sur des sections sismiques en **domaine temps**. Or, l'étude que nous proposons se place à la fin de la chaîne de modélisation structurale et donc en domaine profondeur. Comme le chapitre suivant le montre, la géométrie des failles en domaine temps et en domaine profondeur peuvent être très différentes. Il faut donc prendre les règles établies précédemment avec beaucoup de prudence. Cependant, les grandes lignes des styles structuraux décrits dans les paragraphes précédents restent valables en domaine profondeur.
- Ces règles sont souvent **très qualitatives** ("la faille a un fort pendage"). Un géologue structuraliste est souvent capable de dire qu'un modèle est non valide, en fonction de ce qu'il connaît de la région et de son expérience. Cependant, il est très difficile d'obtenir des règles systématiques. Ceci est fort compréhensible, du fait que ces géométries dépendent du cas étudié et oscillent généralement entre les styles tectoniques présentés précédemment. Pour utiliser réellement de telles règles, il faut attendre une formulation plus mathématique (un projet est en cours à ce sujet chez Elf Exploration Production).
- **Comment connaître l'état des contraintes?** En effet, dans beaucoup de cas, la détermination du style tectonique fait appel aux directions des contraintes principales (voir "*blocs basculés en extension oblique*"). Dans certains cas, les structuralistes sont incapables de déterminer la direction des contraintes tellement la zone est complexe, surtout lorsque la **tectonique est polyphasée**.

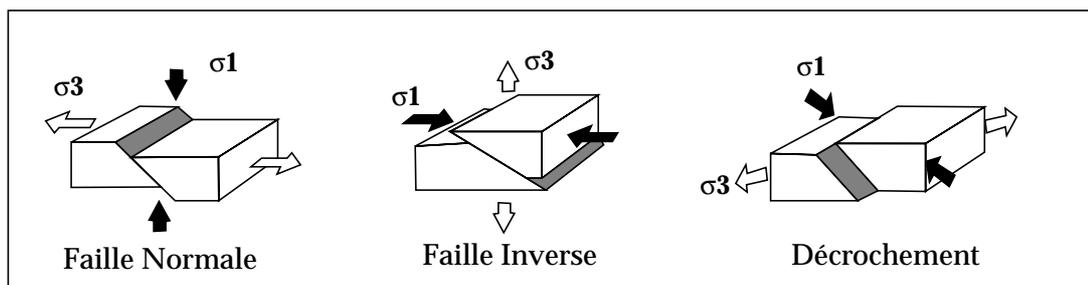


Figure 2-30 : Directions des contraintes principales pour les 3 styles de faille.

- En général, les structures correspondent à une combinaison de plusieurs styles [figure 2-31], [20 - Wernicke], [29 - Gibbs 1].

### Liste non exhaustive ...

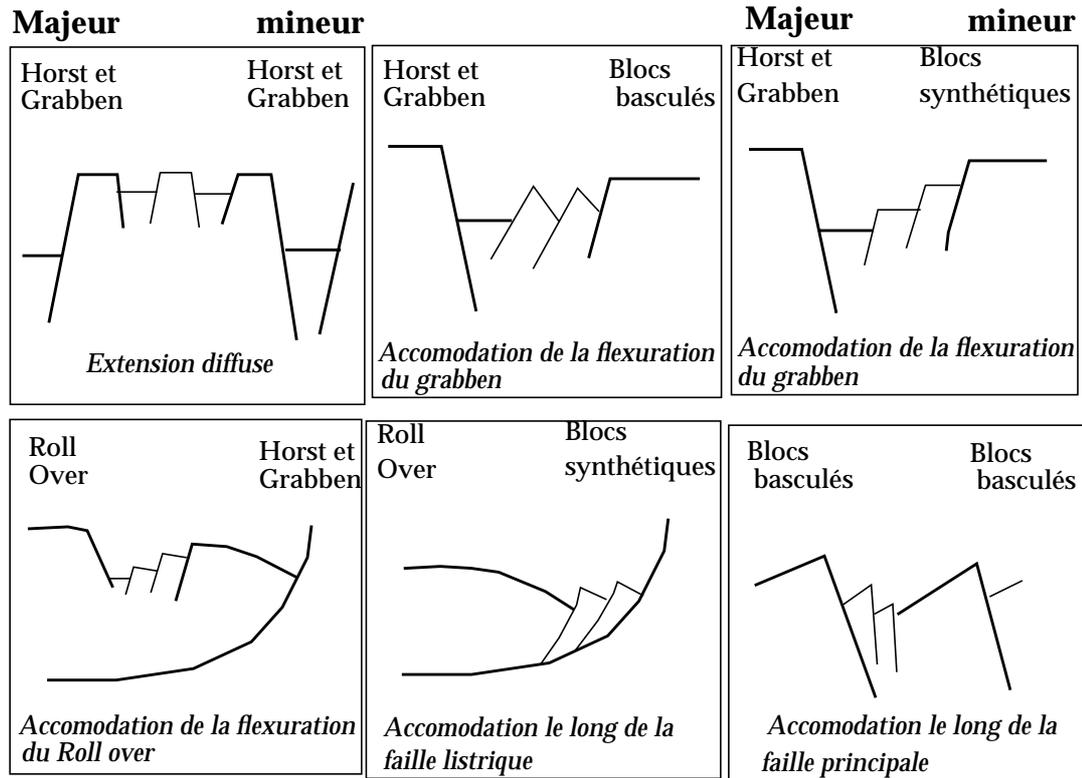


Figure 2-31 : Quelques exemples de combinaisons de deux styles tectoniques (des combinaisons plus complexes sont possibles).

- Influence de l'héritage** lors de tectoniques polyphasées.  
En fonction de la quantité et de la nature des sédiments qui se déposent entre les différents événements tectoniques, l'influence des structures héritées se fait plus ou moins sentir [figure 2-32]. Selon les cas étudiés, il faut donc plus ou moins tenir compte des structures situées plus profondément que la zone étudiée
  - Forte influence des structures héritées
  - Influence des structures héritées plus diffuse (plus de sédiments déposés, couches compétentes)
  - Influence nulle des structures héritées (niveau de décollement)

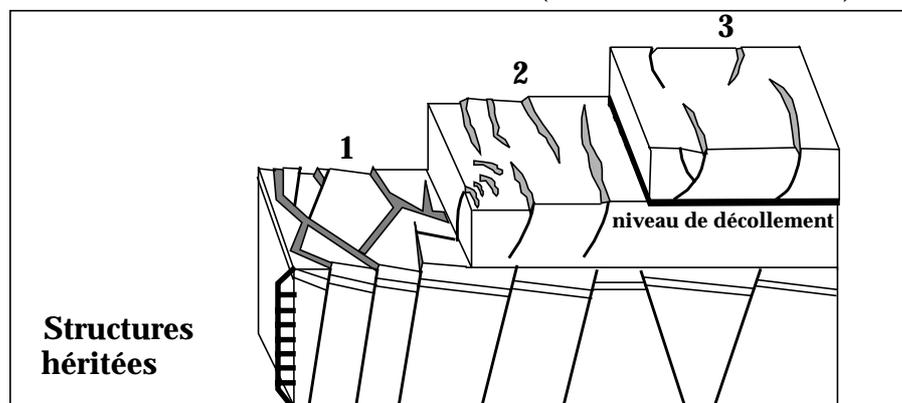


Figure 2-32 : La sédimentation joue un rôle lors de tectoniques polyphasées. [1]: les structures héritées ont une **forte influence** lorsque la **sédimentation** sur les structures réactivées **est faible**; [2]: l'influence est **plus diffuse** si l'épaisseur des sédiments **augmente** ou si les roches sont compétentes; [3]: cette influence **disparaît** au dessus d'un **niveau de décollement**.

- Il faut aussi tenir compte de **la lithologie**. En effet, le style de déformation est directement fonction de la lithologie. *En compression*, un bloc isotrope donnera des failles verticales, alors qu'au niveau d'un empilement de couches, les mouvements horizontaux seront plus importants le long des couches les moins résistantes (niveaux de décollement). *En extension*, la géométrie des failles est conditionnée par la mécanique des roches constituant l'ensemble déformé: un niveau ductile épais entraînera une accommodation de la déformation et donnera des failles normales sans courbure, tandis qu'un niveau ductile mince donnera des failles listriques.
- Enfin, il faut être attentif au fait que l'interprétation à l'échelle du réservoir a ses limites et qu'il faut toujours vérifier que le schéma structural et les déformations impliquées sont cohérents avec les déformations connues régionalement (cohérence cinématique), et que les structures décrites s'intègrent dans un système tectonique d'échelle régionale (cohérence mécanique). L'ensemble de ces études constitue ce qu'on appelle l'étude de la **cohérence régionale**.

## 2. Modélisation de la géométrie des failles

Comme cela a été montré dans la section précédente, le monde des failles est très vaste et la modélisation de leur géométrie à partir des seules données géophysiques est difficilement envisageable, surtout dans le but de réaliser des études d'incertitude.

Nous avons donc pensé qu'en partant d'une première surface de faille "réaliste", il serait éventuellement possible de calculer des statistiques sur les rejets le long de cette faille (ainsi que sur d'autres paramètres), puis d'essayer de les utiliser comme contraintes supplémentaires lors des simulations. Pour cela, il faut être capable de construire une faille correcte à partir du peu de données dont nous disposons, avec un degré de confiance suffisant pour la prendre comme *référence* par la suite.

Depuis quelques années, se rendant compte que l'étude précise de la faille elle-même, et des profils sismiques ne suffisaient pas toujours [20 - Wernicke], un certain nombre de chercheurs ont adapté des techniques utilisées en domaine compressifs et basés sur la *conservation de la masse* de sédiments lors du mouvement dans une coupe parallèle à la direction de l'extension.

Une première construction géométrique simple a ainsi été proposée par Verall en 1981 [*Chevron Construction*], [60 - Verall]. Cependant, cette méthode utilisait des hypothèses trop restrictives: elle ne s'appliquait qu'aux failles listriques, supposait un cisaillement simple vertical, et ne tenait pas compte de la compaction. Elle a donc été reprise par de nombreux auteurs et peu à peu affinée: prise en compte d'un cisaillement incliné, d'une éventuelle compaction, application au cas des failles inverses....

Après avoir décrit les principales hypothèses ainsi que les données nécessaires à l'application de cette méthode, nous décrirons la première version de la méthode puis les améliorations qui ont été apportées à la méthode initiale.

La méthodologie générale proposée est relativement simple, et part de l'hypothèse que *la forme des horizons après déformation et la forme de la faille sont liées*. Elle peut être décomposée en différentes étapes [figure 2-33]:

1. Construire par une méthode qui sera expliquée par la suite une géométrie en coupe de la faille, et ceci pour chacun des horizons pointés.
2. Réaliser cette construction pour différentes valeurs de certains paramètres (compaction ( $\Phi$ ), angle des plans de cisaillement ( $\alpha$ )...)
3. Sélectionner la solution idéale pour laquelle toutes les formes calculées pour la même coupe se superposent (en haut à droite sur la figure).

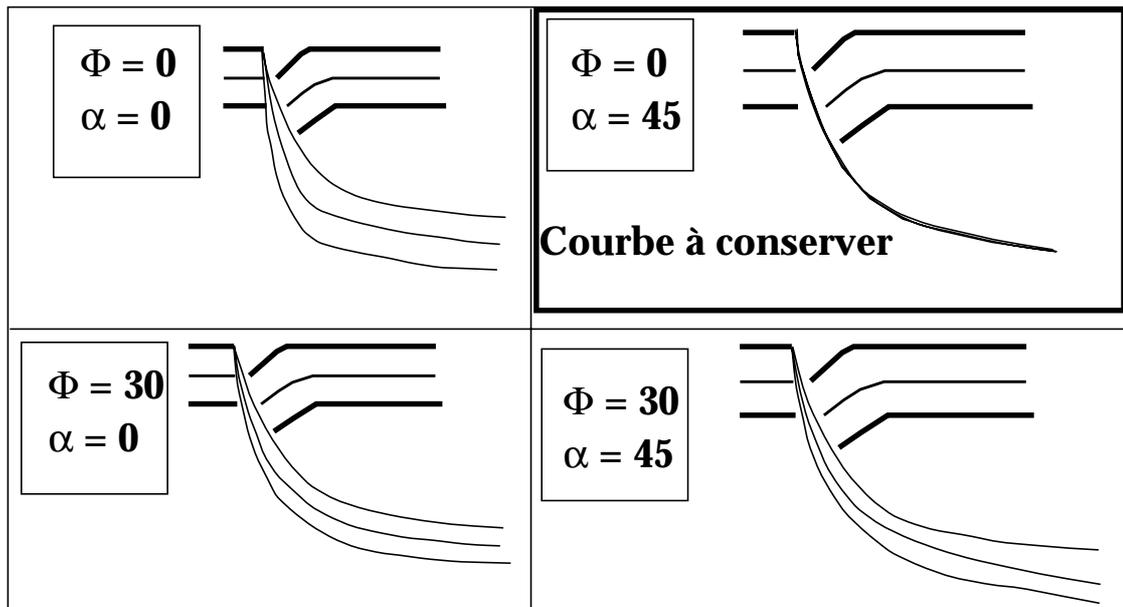


Figure 2-33 : Tenir compte de la forme des horizons pour déterminer la forme de la faille.

## 2.1 Hypothèse et données nécessaires

Le principe de base de l'ensemble de méthodes présentées dans cette section est la conservation de matière sur l'ensemble d'une section, tout au long de la déformation. L'aire à conserver peut être l'aire totale s'il n'y a pas de compaction, et l'aire solide sinon [figure 2-34].

Supposons que la géométrie avant le mouvement soit celle schématisée sur la *figure a* (pour des raisons de simplification, la faille devient plane et horizontale en profondeur, mais ce n'est pas une nécessité). Si maintenant, tous les points du bloc de droite se déplacent d'un même vecteur  $h$ , on obtient la géométrie de la *figure b*.

*Remarque:* dans la première version de la méthode,  $h$  est supposé horizontal.

La section s'étant étendue d'une aire  $A$ , et en supposant qu'aucune partie du mouvement ne se produit en dehors de la section, cette aire est égale à l'aire  $B$ .

Dans la réalité, le vide laissé en dessous du bloc translaté ne peut pas être conservé. Le bloc va donc se déformer afin de combler ce vide, créant un nouveau vide au dessus de lui:  $C$ . Ce phénomène est souvent désigné par le terme: accommodation.

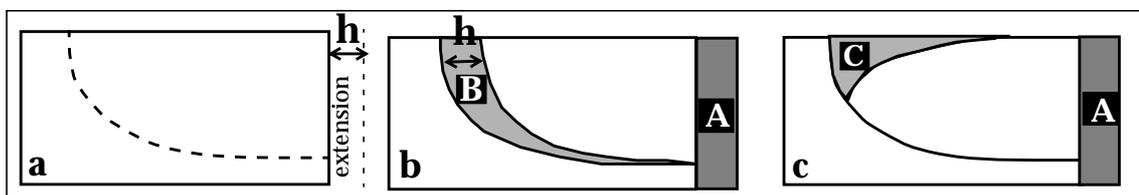


Figure 2-34 : Principe de base pour cette méthode: la conservation de l'aire d'une section lors de la déformation.

### 2.1.1 Hypothèses

Avant de montrer l'évolution et l'amélioration de la méthode, il est utilisé de détailler les hypothèses de base à partir desquelles les formules présentées par la suite ont été établies. I

Les principales hypothèses sont les suivantes:

- **Il n'y a pas de mouvement en dehors du plan de la section étudiée.**

La section doit être parallèle à la direction d'extension. Cependant, cette hypothèse est rarement vérifiable avec les données qui nous sont fournies. De plus, cette hypothèse a été largement critiquée, notamment dans les cas où la diagénèse et les transferts de matière par la dissolution sous pression affectent le volume des différentes couches [56 - Gibbs 2].

- **Un des blocs reste rigide durant la déformation.**

C'est l'hypothèse principale de la majeure partie des modèles étudiés.

D'après les géologues avec lesquels j'ai discuté, cette hypothèse est valable dans la plupart des cas de failles listriques. Elle s'appuie notamment sur des études des tremblements de terre générés par des failles normales. Les mouvements apparaissant dans le bloc supposé fixe sont considérés comme *négligeables* comparés à ceux survenant dans le bloc en mouvement. Cette hypothèse n'est donc valable jusqu'à une profondeur de 10 km, profondeur maximum à laquelle on trouve de tels tremblements de terre. Au-delà, il se peut que des mouvements se produisent dans les deux blocs séparés par la faille. Les formules développées par White [61 - White 1], peuvent toutefois tenir compte d'une éventuelle rotation du bloc supposé fixe ici.

- **La géométrie des sédiments n'a pas été altérée par des effets de compaction.**

Malheureusement, cette hypothèse n'est valable que dans peu de zones. En effet, dès qu'il y a dépôt synsédimentaire, diagénèse, et dans toutes les zones où il y a beaucoup d'eau dans les sédiments, ou dans lesquelles un fort enfouissement permet des transferts de masse par pression fluide, de grandes variations de volume peuvent apparaître au cours du temps.

**Remarque:** White [62 - White 2], propose une solution qui peut tenir compte d'une éventuelle compaction, à condition de pouvoir estimer en plus la valeur  $\phi_0$  de la porosité initiale.

### 2.1.2 Données nécessaires

Dans la dernière version de la méthode proposée par White [63 - White 3], et après de nombreuses améliorations, ces hypothèses sont très succinctes:

- **La géométrie d'au moins deux horizons**
- **La valeur du rejet horizontal  $h$**
- **La géométrie des couches avant déformation, appelée aussi *régionale***

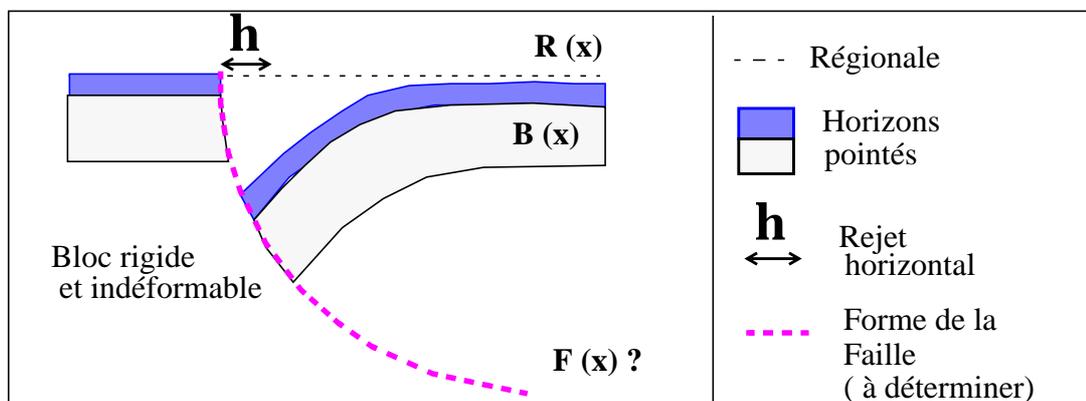


Figure 2-35 : Données utilisées dans la dernière version de la méthode

## 2.2 Construction géométrique

Comme on a pu le voir, les données utilisées dans cette famille de méthodes sont relativement restreintes. Pour arriver à tenir compte d'un cisaillement incliné et d'une éventuelle compaction à partir de si peu de données, la technique géométrique simple proposée par Verrall en 1981 [60 - Verrall], a été beaucoup reprise. Nous proposons de repasser en revue brièvement toutes ces évolutions, afin que la version actuelle de la technique soit plus compréhensible et mieux justifiée.

### 2.2.1 Construction de Verrall (*Chevron construction*)

Cette méthode a été proposée pour construire la forme d'une faille listrique unique à partir d'un profil de "roll-over". Il est supposé dans cette technique que seule la composante horizontale du mouvement doit être utilisée (modèle du *cisaillement simple vertical*).

La méthode géométrique est relativement simple [figure 2-36]:

- calcul de la composante horizontale du rejet au niveau de la faille **h**
- tracé du segment **oA**. Ce vecteur sera le premier segment de la faille.
- a partir du point A, tracé d'une ligne verticale jusqu' à la courbe de la régionale (forme de l'horizon avant déformation). On obtient ainsi le point B.
- a partir de B, on trace un vecteur de longueur h sur la régionale. On obtient C.
- en traçant la verticale passant par C jusqu'à l'horizon, on obtient le point D.
- on reporte le segment **BD** à la suite du segment OA.
- on continue ainsi de suite, jusqu'à obtenir un vecteur *parallèle à la régionale*

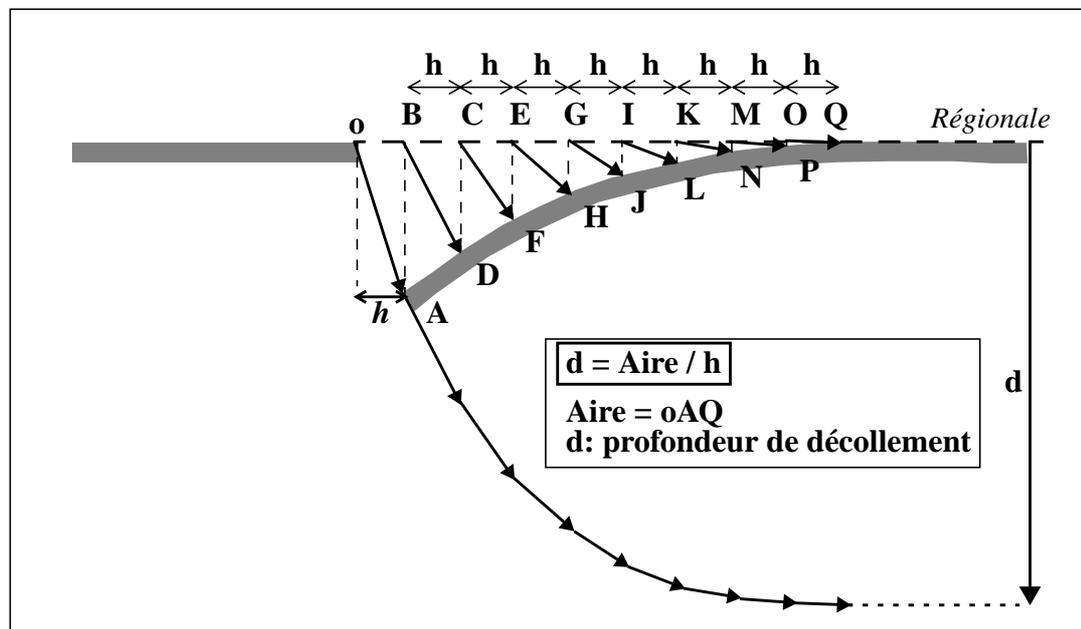


Figure 2-36 : "Chevron construction" de Verrall, 1981

### 2.2.2 Méthode de Davison

Davison a proposé une méthode géométrique similaire, mais qui ne garde pas une longueur constante égale à h (modèle "flexural slip") [57 - Davison]. La méthode reste très proche. Cependant, les opérations ne s'effectuent pas dans le même ordre:

- On trace la verticale passant par A afin d'obtenir le point B et la longueur  $la$ .
- A partir de A, on reporte la longueur  $la$  sur la trace de l'horizon, on obtient C.
- On reporte ensuite le vecteur BC à la suite du vecteur oA sur la trace de la faille.
- On trace la ligne verticale passant par C. L'intersection de cette ligne avec la régionale donne le point D.
- Le segment BD donne la longueur  $lc$  qu'il faut reporter sur l'horizon pour obtenir le point E.
- Dans cette méthode aussi, le processus s'arrête lorsque l'on obtient un vecteur parallèle à la régionale.

Une comparaison des deux méthodes est présentée sur la figure suivante [figure 2-37].

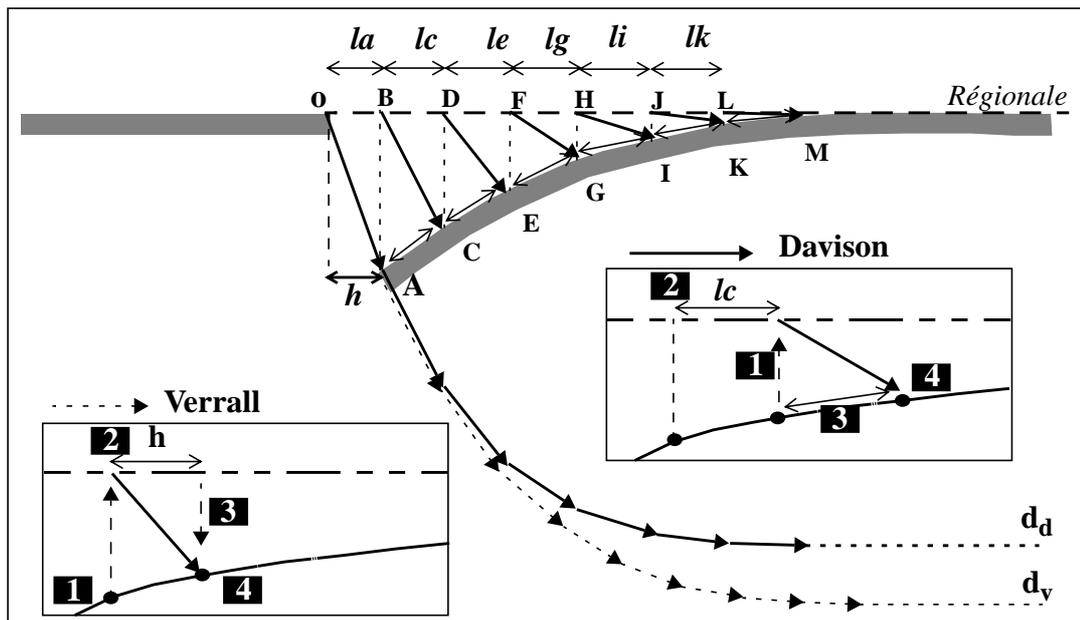


Figure 2-37: Construction de Davison, 1986

Comme on peut le constater, les résultats obtenus sont assez différents. Par exemple, la profondeur du niveau de décollement (i.e.: profondeur à laquelle la faille devient horizontale) est surestimée avec la méthode de Verrall. D'après Gibbs [56 - Gibbs 2], ceci est dû au fait que l'aire d'extension A est rarement un simple rectangle et qu'elle peut avoir une forme plus complexe.

De plus, la méthode de Davison donne une meilleure estimation de cette profondeur, en utilisant la formule suivante:

$$d_{\text{Davison}} = \frac{\text{aire}(oAM)}{lq} \quad (\text{meilleur que: } d_{\text{Verrall}} = \frac{\text{aire}(oAQ)}{h})$$

Malheureusement, ces deux constructions géométriques ne résolvent pas tous les cas, loin s'en faut. De plus, elles donnent parfois des résultats théoriquement faux du fait des hypothèses trop réductrices (notamment sur le modèle utilisé: «*cisaillement simple vertical*» pour Verrall, et «*glissement par flexure*» pour Davison, qui suppose que le bloc de toit possède une stratification parallèle [63 - White 3]).

Les auteurs suivants, après avoir montré les problèmes de ces deux modèles, proposent d'autres techniques, plus analytiques qui permettent de résoudre un plus grand nombre de cas.

### 2.3 Méthode analytique de White (1986)

Dans son article [61 - White 1], White montre qu'il est très important de ne pas cantonner le cisaillement simple à être vertical. D'autant plus que l'observation des petites failles antithétiques montre qu'il est très rare que le cisaillement soit vertical. Une même forme de faille peut être obtenue avec deux angles de cisaillement différents compensés par des extensions différentes [figure 2-38]:

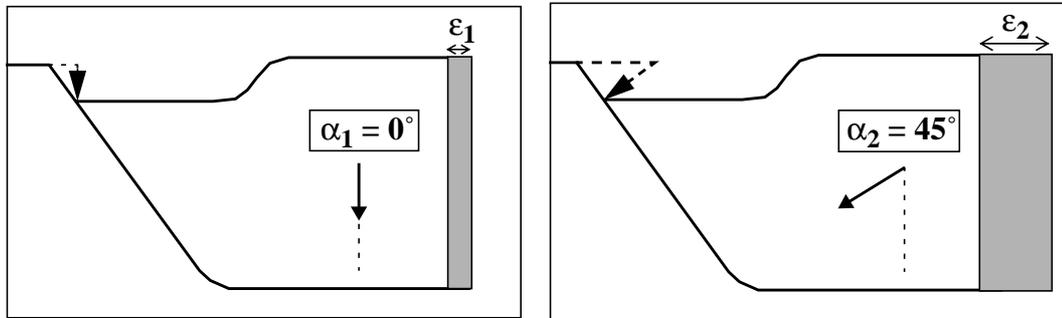


Figure 2-38 : Un cisaillement oblique avec une plus grande extension peut donner le même résultat qu'un cisaillement vertical

Inversement, lorsque l'on fait varier l'angle du cisaillement  $\alpha$ , pour une même extension, on peut voir que cela entraîne de grandes différences sur la forme de la faille obtenue [figure 2-39]. En s'appuyant sur la [figure 2-39], on peut par exemple remarquer que, pour un angle  $\alpha = 45^\circ$ , la géométrie de la faille n'est connue que jusqu'au point **b**, alors que pour  $\alpha = 0^\circ$ , elle peut être dessinée jusqu'au point **c**, à la verticale du point **a**.

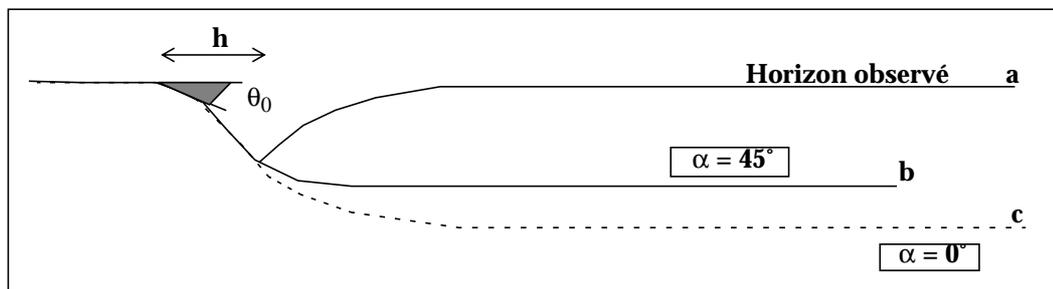


Figure 2-39 : L'angle de cisaillement a une grande importance

La méthode proposée par White [61 - White 1] permet de tenir compte de ces observations. Contrairement aux méthodes précédentes, ce n'est pas une méthode géométrique, mais une méthode analytique. Toutes les coordonnées qui seront données par la suite se situent dans l'espace à deux dimensions muni d'un repère orthonormé  $(x', y')$ , où  $y'$  correspond à la direction du cisaillement [figure 2-39]:

White fait les hypothèses suivantes:

- Les déplacements sont faibles
- Il n'y a pas de mouvement en dehors du plan de la section (déformation plane)
- La déformation se fait par cisaillement simple, c'est à dire que si l'on appelle  $u$  la composante horizontale de la vitesse de déformation, et  $v$  sa composante verticale, les équations suivantes sont à respecter:

$$\frac{\partial v}{\partial y'} = \frac{\partial u}{\partial x'} = \frac{\partial u}{\partial y'} = 0 \quad \frac{\partial v}{\partial x'} = \frac{U_0}{h'} \cdot \frac{d}{dx'} \langle R' - B' \rangle \quad (U_0 = Cste)$$

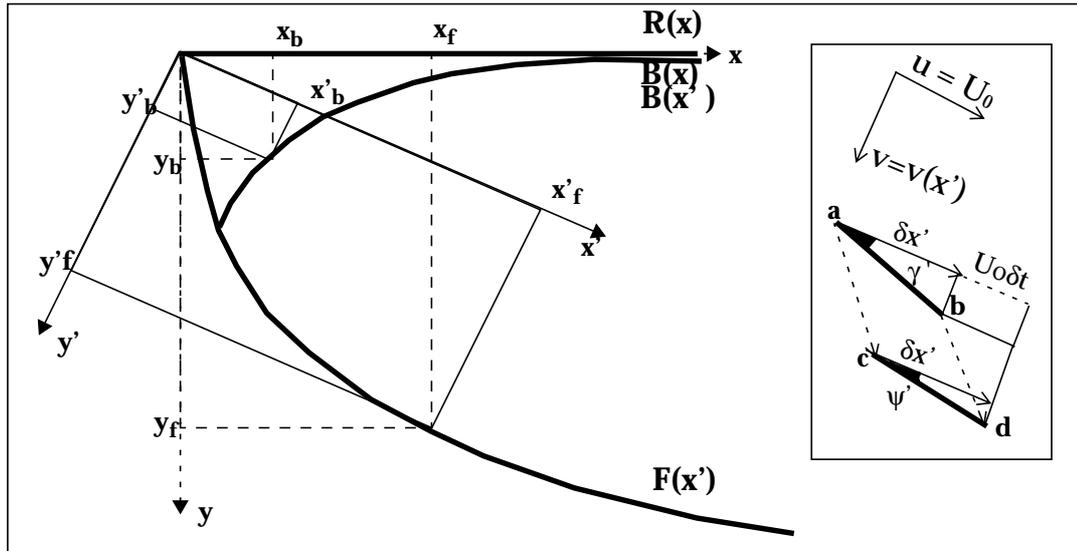


Figure 2-40 : Changement de repère utilisé dans la méthode de White.

- Le bloc situé sous la faille n'est pas déformé<sup>1</sup>
- La géométrie n'est pas altérée par la compaction. Cependant, cette hypothèse disparaît dans les dernières versions de l'équation.

Ainsi, si on utilise la construction précédente, un segment élémentaire  $ab$  est déplacé en  $cd$  [figure 2-40], tel que:

$$a = (x', y')$$

$$b = (x' + \delta x', \delta x' \tan \gamma')$$

$$c = (x' + U_0 \delta x', y' + v(x') \delta t)$$

$$d = (x' + \delta x' + U_0 \delta t, y' + \delta x' \tan \gamma' + v(x' + \delta x') \delta t)$$

avec:

$\gamma'$ : angle entre l'élément  $ab$  et l'axe  $Ox'$

$\psi'$ : angle entre l'élément  $cd$  et l'axe  $Ox'$

En utilisant ces équations, les hypothèses précédentes, ainsi que des règles simples de géométrie, on arrive à la formule suivante, donnant l'équation de la forme d'un horizon à partir de celle de la faille (problème direct):

$$B' - B'_0 = h' \left\langle \frac{d}{dx'} F' - \tan \theta'_0 \right\rangle + R' - R'_0 \quad (1)$$

ou encore, dans le cas qui nous intéresse, l'équation de la forme de la faille à partir de celle de l'horizon (problème inverse):

$$F' = \frac{1}{h'} \int_0^{x'} \{ B' - B'_0 - (R' - R'_0) + h' \tan \theta'_0 \} dx' \quad (2)$$

1. D'après l'auteur et certains structuralistes de Elf E. P., cette hypothèse est la plus importante. Même si elle peut paraître restrictive, elle est vérifiée dans beaucoup de cas (au moins pour les failles listriques).

**Remarque:** En général, on peut supposer que  $R'$  est de la forme suivante:

$$R' = x' \tan \gamma'_0 + D$$

avec

$$D' = cste$$

$\gamma'_0$  : pendage régional dans l'espace  $(x'y')$

De plus, on peut utiliser les relations suivantes:

$$x' = x \cos \alpha + y \sin \alpha \quad (5)$$

$$y' = -x \sin \alpha + y \cos \alpha \quad (6)$$

et

$$x = x' \cos \alpha - y' \sin \alpha \quad (7)$$

$$y = x' \sin \alpha + y' \cos \alpha \quad (8)$$

On peut ainsi imaginer un algorithme assez simple:

1. Passer de l'espace  $xy$  à l'espace  $x'y'$  à l'aide des formules (5) et (6)
2. Calculer pour chacun des horizons une courbe correspondant à la faille en utilisant la formule (2), et en ayant déterminé au préalable les variables  $B'_0$ ,  $\Theta'_0$ ,  $R'_0$  et  $h'$  [figure 2-41].
3. Calculer les coordonnées réelles de cette courbe dans l'espace  $(xy)$  en utilisant les formules (7) et (8)

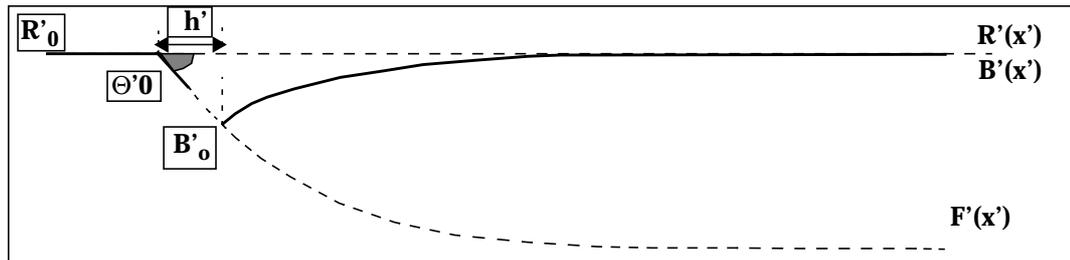


Figure 2-41 : Variables qui doivent être déterminées pour cette méthode

La connaissance des équations de plusieurs horizons permet en théorie d'avoir une meilleure estimation de  $\alpha$ . En effet, la faille ayant la même forme pour tous les horizons, il doit être possible de trouver une valeur de  $\alpha$  commune, qui donne des courbes superposées [figure 2-42].

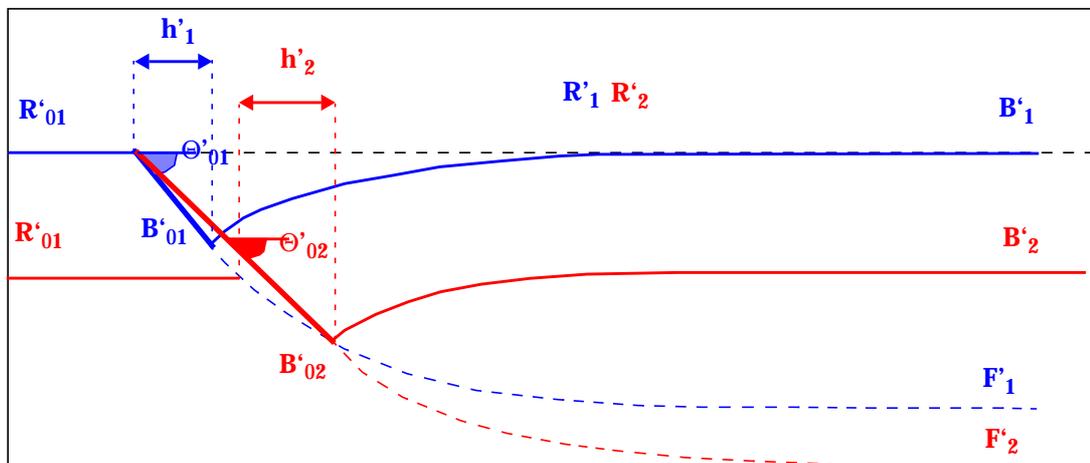


Figure 2-42 : L'utilisation de plusieurs horizons permet d'affiner les résultats.

## 2.4 Discussion de l'application de ces méthodes au sein de nos travaux

### 2.4.1 Hypothèses trop fortes?

Les hypothèses utilisées dans les dernières versions de ces méthodes sont peu nombreuses. Cependant, certaines d'entre elles m'ont paru trop fortes pour être utilisées de manière automatique. Le fait que le bloc de *mur* ne soit **pas déformé** pendant la formation de la faille semble justifié dans le cas des failles listriques mais moins évident pour d'autres styles tectoniques, surtout en contexte extensif où la compaction joue un rôle important [56 - Gibbs 2], [61 - White 1].

De plus, les **données** nécessaires ne sont **pas forcément disponibles**. La sismique étant brouillée au niveau des failles, il paraît difficile d'avoir une bonne estimation de la *distance h* d'une part et des *équations détaillées des différents horizons* d'autre part). Enfin, White remarque qu'il est difficile de trouver des sections sismiques remplissant tous les critères [61 - White 1].

### 2.4.2 Seulement pour les failles listriques en domaine extensif?

Dès la fin des années 80, de nombreuses études de terrain ont montré qu'en extension les failles sont majoritairement des structures planaires. Or ces techniques mises au point au début des années 80 sont difficilement applicables pour de telles structures [59 - Roberts]. Elles sont valables pour la prédiction des failles listriques dues à la gravité comme celles que l'on trouve dans le delta du Niger par exemple [21 - Yielding 1]. Mais il a été montré que des structures en Roll-Over peuvent apparaître avec des failles planes, ou que la formation d'une faille listrique soit compensée par des failles secondaires au lieu d'un Roll Over [56 - Gibbs 2]. Dans ce genre de cas, il serait évidemment très dangereux d'utiliser ces techniques pour déterminer la forme des failles [21 - Yielding 1], [122 - Gibbson].

### 2.4.3 Conservation de la masse dans une section 2D: et la 3D?

Toutes les méthodes présentées dans cette section correspondent à des techniques en deux dimensions. Or nous travaillons en trois dimensions. Il serait envisageable d'utiliser de telles méthodes en "2,5 D" (ou "multi-2D") [58 - Galera]. Cependant, dans des structures complexes, il peut être difficile de trouver un ensemble de sections parallèles à la direction d'extension globale [56 - Gibbs 2]. Quand à l'adaptation directe de ces méthodes en trois dimensions, cela représente en un travail épineux. Nous avons donc préféré nous focaliser sur des problèmes plus proches des objectifs que nous étions fixés au départ, à savoir construire une méthodologie complète pour l'étude des incertitudes sur les failles, allant jusqu'au calcul des volumes.

### 2.4.4 Comment tenir compte de données?

Avec autant de modèles décrivant tant la géométrie des failles courbes, que celle des failles planes, il n'existe pas, en général, une unique solution au problème qui consiste à déduire la géométrie d'une faille à partir d'un nombre limité d'horizons autour de cette faille [21 - Yielding 1], et il est difficile de choisir quel modèle utiliser [63 - White 3]. Il peut paraître plus judicieux d'utiliser toutes les données à disposition (horizons interprétés en trois dimensions, puits, courbes de failles interprétés en domaine profondeur par les structuralistes). L'utilisation de données extérieures que nous tentons de respecter au mieux lorsque nous construisons les modèles géologiques en domaine profondeur ne nous paraissait pas possible avec les méthodes décrites précédemment, puisque ces méthodes utilisent des sections sismiques en domaine temps [61 - White 1].

### 3. Conclusion: les choix qui ont été faits pour ce mémoire

#### 3.1 Faire confiance au travail du géologue structuraliste: faille de référence utilisée comme une donnée.

Lorsque l'on parle d'une faille, il est difficile de savoir à quoi l'on fait référence: les structures peuvent être très complexes et sont très dépendantes de l'histoire de la région, de la lithologie....

Il semble très **difficile d'automatiser la construction d'une faille** géologiquement correcte à partir des seules données géophysiques (horizons migrés) et des données de puits, d'autant plus que les règles avancées ne sont pas encore établies et sont souvent *remises en cause* par de nouvelles études. Il est indispensable, à mon avis, de faire confiance au géologue structuraliste qui travaille en fonction de ses connaissances de la région et de son expérience.

C'est pourquoi par la suite, nous avons considéré que la faille «la plus plausible» était une **donnée fournie par le structuraliste** et que nous nous sommes attachés à nous écarter le moins possible de cette *solution de référence* lors des simulations. En effet, même si cette interprétation contient une part d'incertitude, puisque l'état actuel des connaissances sur la géométrie des failles est encore modeste et que les interpréteurs n'ont pas encore d'outils analytiques à leur disposition [34 - [Hardman](#)], il faut s'en contenter, tout en gardant à l'esprit que ce n'est qu'une interprétation et que la géométrie peut légèrement varier autour de cette géométrie de référence.

Toutefois, il paraît important d'avoir de bonnes notions sur les différents styles tectoniques et de garder à l'esprit les grandes lignes de ce qui a été décrit dans ce chapitre, afin de rester cohérent avec la solution initiale et de respecter au mieux l'idée que le géologue structuraliste a en tête en fonction des connaissances actuelles de la zone considérée.

#### 3.2 Travaux envisageables

##### 3.2.1 Utilisation de bibliothèques de formes

Il a été envisagé au début de ces travaux de construire une bibliothèque de formes à partir de laquelle le structuraliste pourrait choisir une interprétation. Cette interprétation serait ensuite éditée en fonction des données disponibles. Par manque de temps, cet aspect n'a pas été abordé, mais il pourrait être intéressant de pousser dans cette voie afin d'aider les structuralistes.

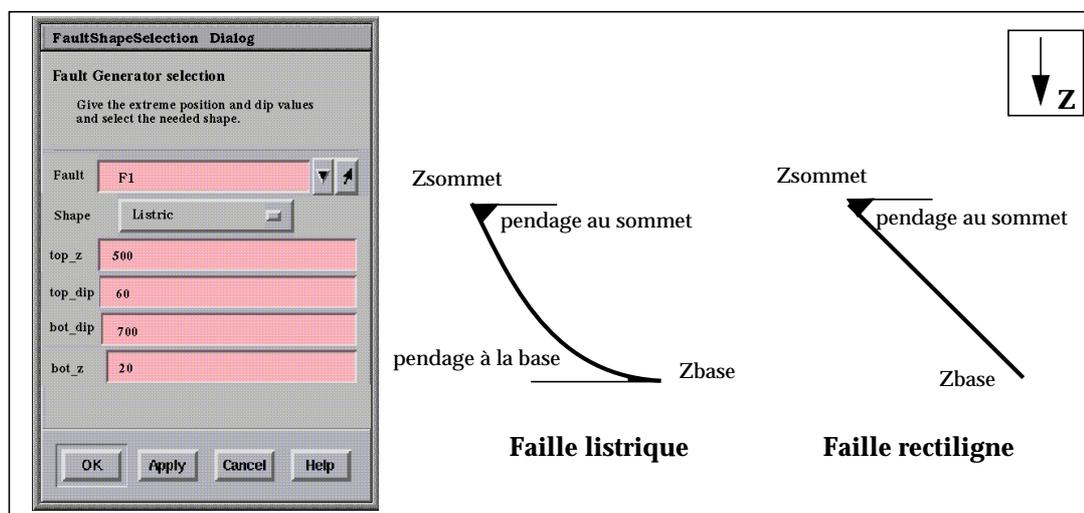


Figure 2-43 : Boîte de dialogue envisageable pour construire automatiquement la forme de la faille en coupe.

### 3.2.2 Utilisation des techniques précédentes: travaux de Cyril Galéra

De la même façon, les techniques de détermination de la forme des failles en coupe à partir de la forme des horizons n'ont pas été implantées dans le cadre de cette thèse. Ces techniques étaient critiquées pour certains types de failles, nous avons en effet préféré concentrer le travail sur la partie "simulations" plutôt que sur la partie "aide à la construction de la surface de référence". Il existe toutefois des cas de figure où ces techniques donnent de très bons résultats, et où leur utilisation apporterait beaucoup.

Un étudiant en thèse du L.I.A.D travaillant en collaboration avec l'Institut Français du Pétrole a déjà implanté de telles méthodes en 2,5D dans le logiciel gOcad [58 - Galera]. Dans les cas où ces techniques s'appliquent, il sera donc possible de les utiliser pour construire la surface de référence.

En attendant que de telles techniques soient disponibles de manière fiable, de nombreuses incertitudes persistent en ce qui concerne la modélisation des failles. Le chapitre suivant décrit comment les failles sont modélisées dans les compagnies pétrolières, et quelles sont les incertitudes rattachées à chacune des étapes.

---

---

## Chapitre 3 Bilan des incertitudes sur les failles:

### De l'acquisition des données à la modélisation géologique.

---

---

Ce troisième chapitre correspond à une partie du stage DEA ayant précédé cette thèse et qui s'est déroulé à Pau dans le Projet Incertitudes [81 - Lecour 1]. Cette partie du stage a permis de dresser un premier bilan des incertitudes intervenant dans les données fournies pour la modélisation des failles. Elle s'est passée essentiellement à suivre différentes études avec des géophysiciens et des géologues spécialisés dans l'interprétation, tant de coupes sismiques que de données de puits telles que la pendagemétrie.

Ce chapitre est très largement inspiré du chapitre 4 du mémoire d'habilitation à diriger les recherches présenté par Pierre Thore en 98 [70 - Thore 1], ainsi que du livre de géophysique écrit par Henry [65 - Henry].

Dans un premier temps, les différentes étapes qui permettent de passer de l'acquisition des données à la modélisation sont brièvement décrites. En effet, le travail de modélisation intervient assez loin dans la chaîne de traitement des données sismiques et géologiques. C'est pourquoi il peut paraître utile de retracer succinctement les différentes étapes permettant de passer des données brutes aux données utilisables par les logiciels de modélisation.

Un second paragraphe montre les multiples sources d'incertitude qui apparaissent au cours de cette chaîne, depuis les phases d'acquisition des données géophysiques et géologiques jusqu'à la modélisation proprement dite.

## 1. Les différentes étapes précédant la modélisation

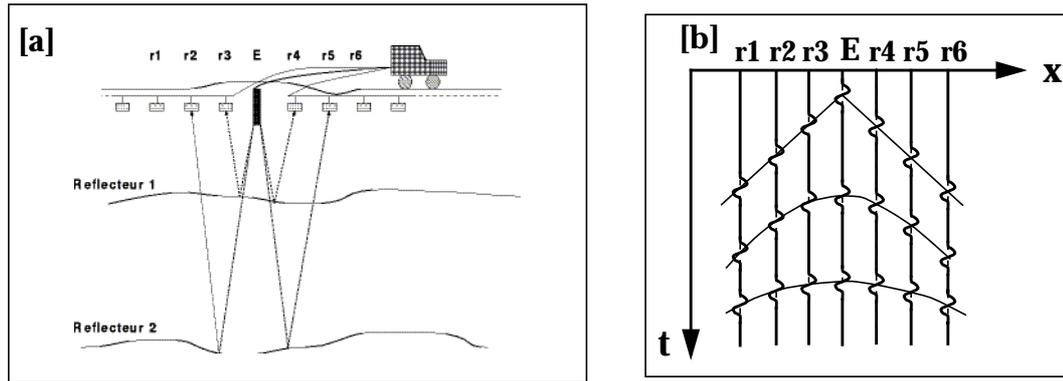
### 1.1 Acquisition des données

#### 1.1.1 Campagnes sismiques

La sismique pétrolière se donne pour but la recherche de la topographie des couches du sous-sol. On utilise principalement la *sismique réflexion*, avec un dispositif assez serré qui permet une visualisation en continu des couches.

Des ébranlements sont provoqués près de la surface, créant un système d'ondes élastiques qui se propagent dans le sol et qui se réfléchissent, se réfractent ou se diffractent en fonction des formes qu'elles y rencontrent.

De nombreux capteurs placés en surface recueillent les réponses dont le géophysicien déduit une image de la structure du sous-sol. L'enregistrement est ainsi constitué par la réunion des traces enregistrées à chaque récepteur, en fonction du temps de réflexion (ou "temps double") [figure 3-1].



**Figure 3-1 : Principe de l'acquisition des données sismiques (d'après Henry).** Des ébranlements sont provoqués en surface à l'aide d'un émetteur (E) [a] et de nombreux récepteurs ( $r_i$ ) placés en surface recueillent les réponses [b].

### 1.1.2 Formation des sections stacks

Le document de base constitué par la trace sismique à l'abscisse  $x$  en fonction du temps  $t$  est largement polluée par le bruit. On utilise donc un enregistrement surabondant afin d'obtenir une mesure "moyenne" moins bruitée (*couverture multiple*).

Puis, une sommation de toutes les traces intersectant une cellule donnée s'effectue suivant une grille régulière (les cellules faisant 12.5m ou 25m de côté). Cette opération de sommation a comme intérêt, outre la diminution du bruit, la grande condensation des données et donc la réduction du temps de calcul par la suite. Elle est primordiale, puisque c'est durant cette étape que l'image du sous-sol se forme.

Mais la section obtenue (*section stack*) est loin d'être interprétable et doit subir de nombreux traitements ultérieurs.

## 1.2 Traitement des données

### 1.2.1 Préparation des données

Les traitements appliqués à la section stack sont de différents types.

Tout d'abord, une *correction d'amplitude* est appliquée pour contrebalancer, entre autres, l'atténuation du signal en profondeur. Elle se fait par l'application d'un facteur multiplicatif  $G(t)$  à l'ensemble de la trace. Ensuite, les bruits résiduels sont identifiés et réduits au minimum par l'utilisation de filtres adéquats.

Cette série de traitements permet ensuite le passage de la sismique à la géologie (c'est à dire à une section interprétable). Ce passage se fait par une opération essentielle: **la migration**.

### 1.2.2 Migration

Ce traitement permet d'éliminer les *principaux artefacts*, et notamment ceux dus aux points diffractants constitués par l'interruption d'un horizon au niveau d'une faille. Elle permet en outre de restituer la *position réelle* occupée par les différentes réflexions dans l'espace à deux dimensions.

Généralement, le point correspondant au réflecteur est pris à la verticale du point milieu entre l'émetteur et le récepteur. Dans le cas de couches pentées, cette hypothèse a pour conséquence de déformer l'image réelle du sous-sol [figure 3-2].

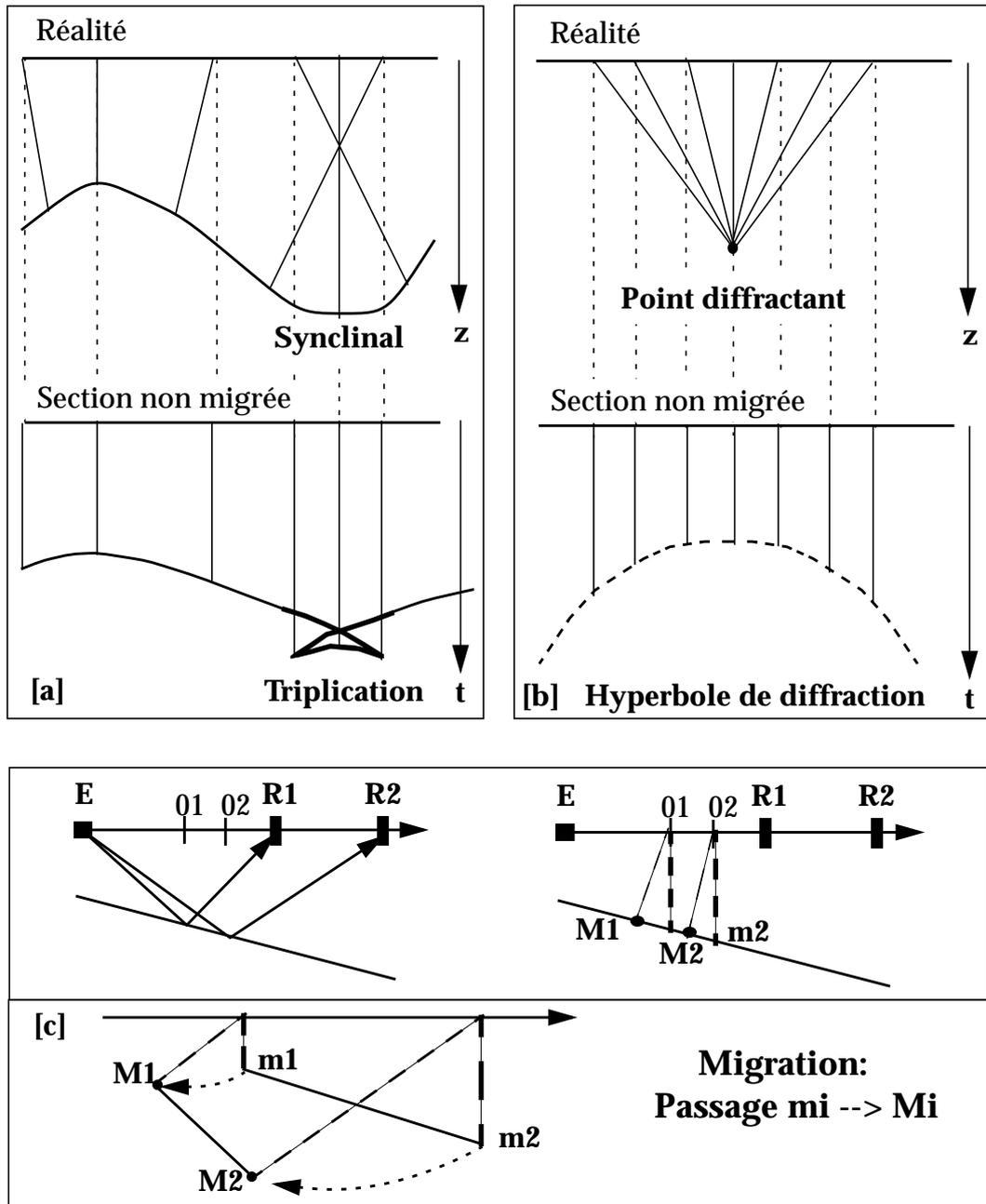


Figure 3-2 : Deux cas où la migration est indispensable: [a] synclinal et [b] point de diffraction. Le principe de la migration est montré en [c] (d'après [65 - Henry]).

Pour des synclinaux, par exemple, on peut voir des *triplifications* se former sur les sections non migrées et rendre l'interprétation impossible ([figure 3-2] [a]). De la même manière, ces sections non migrées peuvent être perturbées par des *points diffractants* (notamment dans les zones faillées). A ce niveau, l'énergie sismique est renvoyée dans toutes les directions de l'espace. Sur la section sismique, les trajets initialement obliques sont reportés verticalement et s'alignent sur une hyperbole (*hyperbole de diffraction*) qui parfois s'étend très largement sur la section et perturbe l'interprétation ([figure 3-2][b]).

Dans ces deux cas de figure, la migration permet de rétablir une image plus réaliste du sous-sol ([figure 3-2] [c]). Cette opération devient donc indispensable dès que l'on passe en domaine géologique complexe.

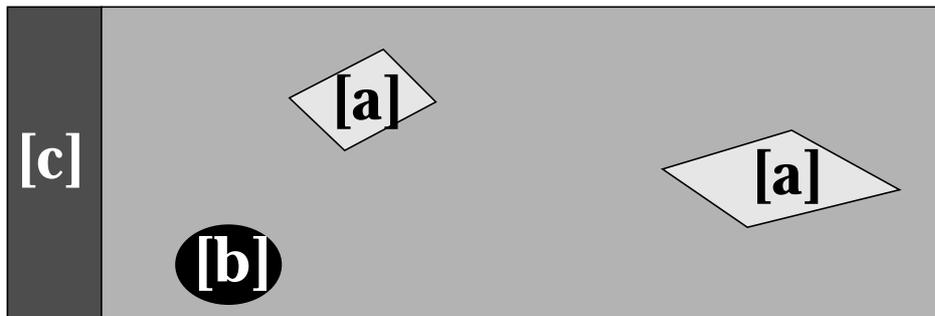
**Remarque:** Il existe 2 types de migration: la migration temps qui se contente de déplacer (migrier) les horizons et la migration profondeur qui en plus intervient sur l'axe Z et fait intervenir la vitesse des ondes ( $Z = T * V$ ).

Malheureusement, les vitesses qui servent à la migration sont mal connues et lorsqu'elles sont mal estimées, l'image résultante peut être déformée. C'est pourquoi on utilise préférentiellement des *sections migrées temps* pour l'étape suivante (*l'interprétation*). Celle-ci est en effet plus robuste que la migration profondeur, très dépendante du champ de vitesses utilisé. Par contre, nous verrons plus tard que faire l'interprétation en domaine temps rend d'autant plus difficile la modélisation des failles en domaine profondeur (Voir « Validation de l'interprétation », page 61.).

### 1.3 Interprétation des données

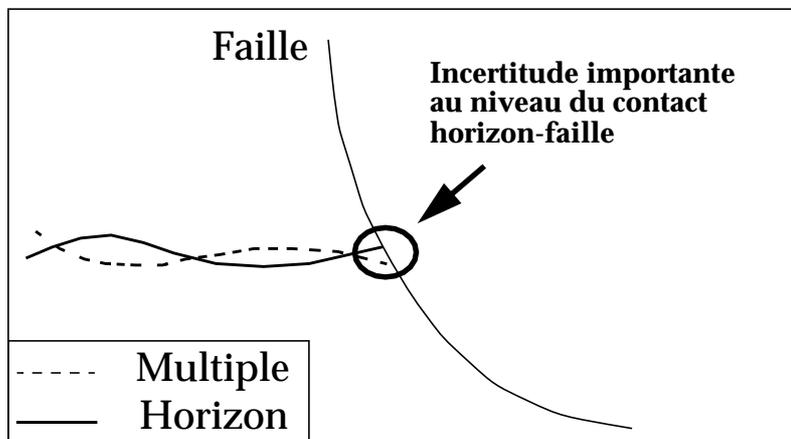
#### 1.3.1 Création d'un premier maillage lâche.

L'interprétation des données sismiques commence par une phase de *critique des données* qui permet de repérer les différents types d'artefacts présents dans la zone et d'en tenir compte par la suite. Seules quelques coupes sismiques sont interprétées afin de repérer les principaux accidents [figure 3-3].



**Figure 3-3 :** Carte d'amplitudes utilisée pour repérer les zones où l'interprétation sera difficile: [a] problèmes de traitement qui donnent une amplitude beaucoup trop forte par rapport au reste de la zone; [b] plate-forme pétrolière déjà en place et qui empêche l'enregistrement de données à ce niveau; [c] superposition des données de cette campagne avec celles d'une ancienne campagne qui donne des amplitudes beaucoup trop fortes.

Remarque: C'est aussi durant cette phase de critique des données que les principaux *multiples* sont repérés (ils sont en partie responsables des incertitudes concernant les relations horizon-faille) [figure 3-4].



**Figure 3-4 :** Les multiples augmentent les incertitudes au niveau des contacts horizon-faille.

### 1.3.2 Pointé des différentes structures dans les 3 directions de l'espace

Une première phase consiste à caler la sismique aux différents puits (passés dans le domaine temps). Cette phase permet d'associer un marqueur géologique (horizon remarquable ou faille par exemple) à un marqueur sismique.

Ensuite, l'*interprétation* proprement dite peut être réalisée. Elle se pratique le plus souvent dans le domaine "temps" pour des raisons de robustesse (Voir « Migration », page 56.). L'interprétation s'effectue sur des sections qui sont sélectionnées par l'interpréteur dans les trois directions de l'espace [figure 3-5]: les sections "inlines", les "cross lines", les "time-slice" (coupes à temps constant), les "random" (sections verticales mais parallèles ni à l'axe x, ni à l'axe y).

Après avoir suivi les marqueurs choisis au préalable sur les différentes coupes sismiques, le réseau de faille est déterminé en accord avec les géologues structuraux, les horizons sont interprétés plus finement par la suite, une fois le réseau de failles interprété.

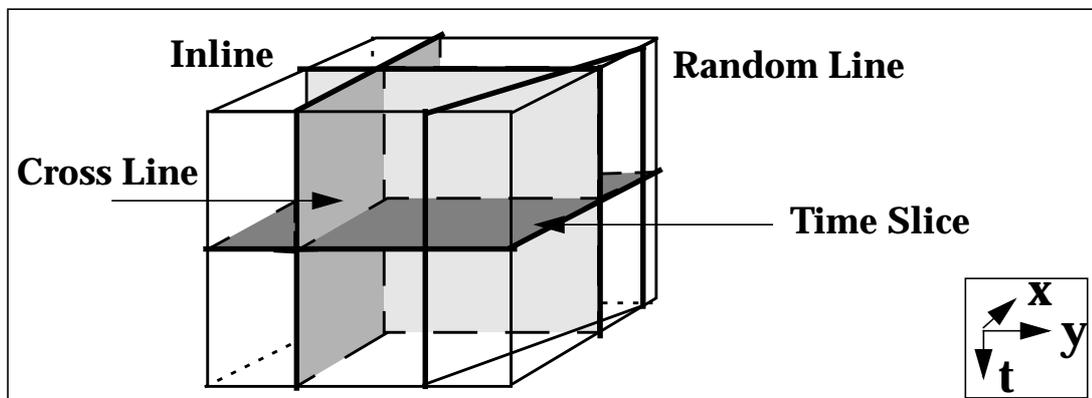


Figure 3-5 : Les différents types de sections interprétées.

Remarque: On peut remarquer que, pour la détermination du réseau de failles, toutes les coupes ne sont pas bonnes à interpréter et que ce sont surtout les coupes perpendiculaires aux failles soupçonnées qui sont interprétées.

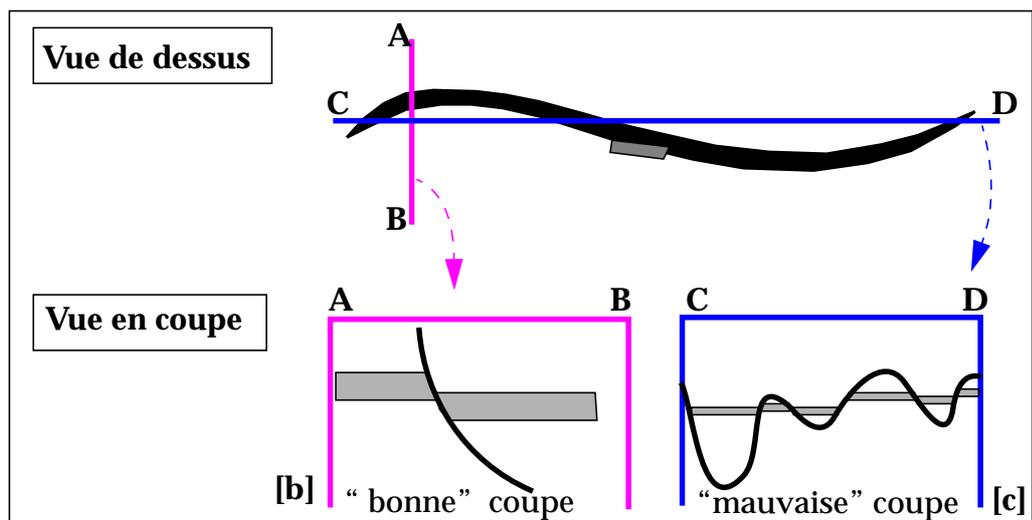


Figure 3-6 : La sélection des coupes sismiques à interpréter est une étape importante et dépend de l'orientation des éléments structuraux présents dans la zone [a]. [b] les coupes perpendiculaires aux failles donnent plus d'informations que les coupes parallèles aux accidents [c].

Après la création de ce premier maillage assez lâche, celui-ci est peu à peu affiné par différentes méthodes.

Les paragraphes suivants concernent plus particulièrement les *failles*. C'est en effet sur les incertitudes qui les concernent que notre attention s'est portée au cours de ce travail.

### 1.3.3 Vérification de la cohérence

La première méthode consiste à vérifier l'alignement des traces d'une même faille repérées sur une série de coupes parallèles (coupes en pointillés fins [figure 3-7]), sur une coupe qui leur est perpendiculaire (trait plein perpendiculaire aux précédents). Si l'une des projections ne concorde pas avec le reste de la distribution, la coupe sur laquelle elle a été repérée est révisée (coupe de la première série montrée par une flèche) [figure 3-7].

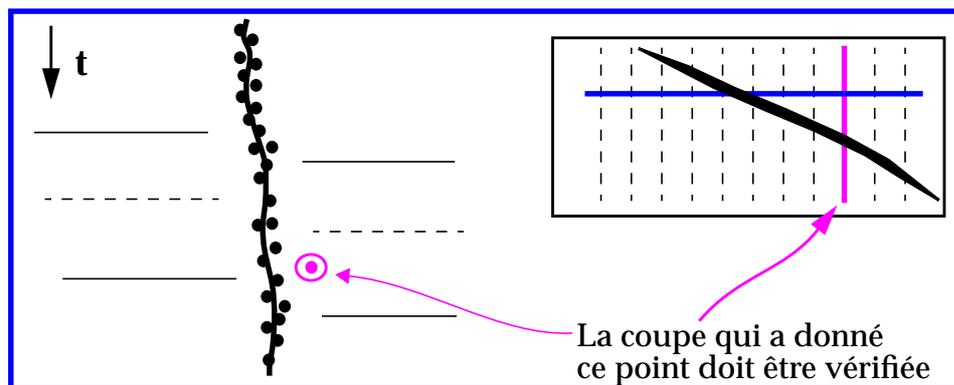


Figure 3-7 : Vérification de la cohérence sur une coupe perpendiculaire à la faille.

Une autre façon de procéder est de regarder la distribution des projections des traces de faille obtenues sur des coupes qui lui sont perpendiculaires (en pointillés) sur une coupe qui lui est parallèle (en trait plein).

On arrive ainsi à améliorer la trace d'une faille dans un plan qui lui est parallèle, en utilisant les informations obtenues dans des plans qui lui sont perpendiculaires et donc nettement meilleures [figure 3-8].

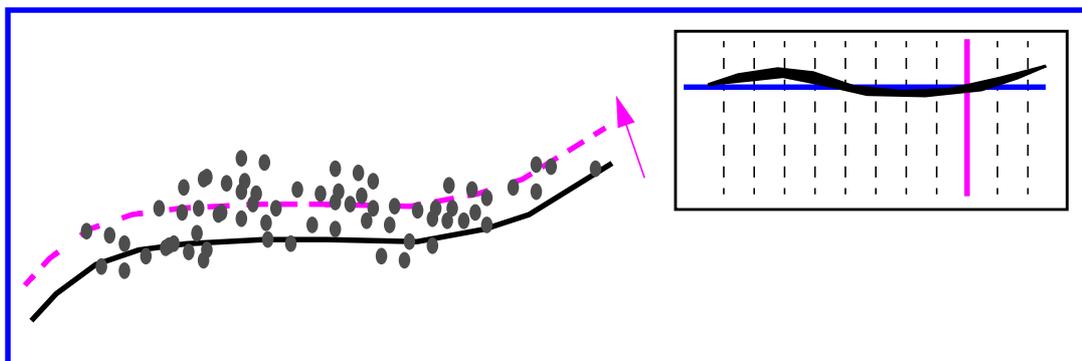


Figure 3-8 : Vérification de la cohérence sur une coupe parallèle à la faille. D'après les traces de failles repérées sur les coupes perpendiculaires à celle-ci (dessinées en pointillés), il vaudrait mieux remonter la trace de la faille sur cette section.

### 1.3.4 Utilisation de cartes d'attributs

Les interpréteurs disposent de nombreuses *cartes d'attributs* qui peuvent guider l'interprétation et permettre de repérer d'hypothétiques accidents. On peut citer par exemple:

- \* cartes d'amplitudes [figure 3-3]
- \* cartes de sommes de valeurs absolues d'amplitudes (au niveau des accidents, la sismique est dégradée et la somme des valeurs absolues des amplitudes chute énormément).
- \* carte des pendages locaux.
- \* .....

Ces cartes permettent entre autres de *sélectionner les coupes* à examiner plus précisément et d'avoir une première idée sur la *hiérarchie* des failles. Mais il faut les manier avec beaucoup de précautions car de nouveaux *artefacts* particuliers à ces cartes peuvent apparaître (certaines zones peuvent faire croire à la présence d'une faille, alors qu'il n'en est rien).

Toutes ces étapes de l'interprétation conduisent à la création de cartes "temps" permettant ensuite de dessiner les traces des différentes failles sur les schémas structuraux [figure 3-9].

**Remarque:** il est parfois possible d'obtenir des zones d'incertitude autour des failles. Ces cartes pourront être très utiles lors de la détermination du domaine d'incertitude à l'intérieur duquel les différents scénarios de faille seront générés. Nous n'avons malheureusement pas eu l'occasion de travailler sur des données de ce type.

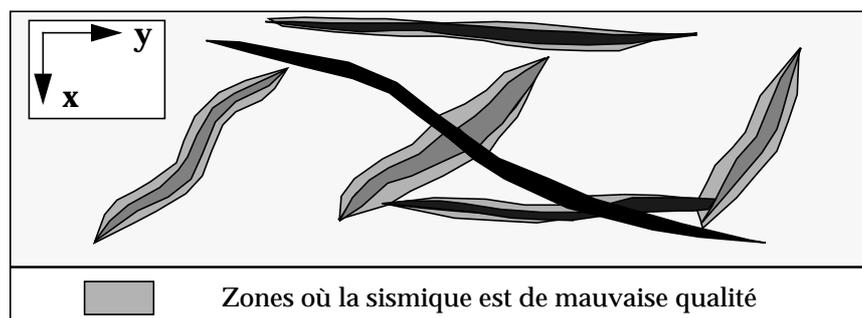


Figure 3-9 : Création d'une carte "temps". Les différentes familles de failles ainsi que les zones de mauvaise qualité sismique sont dessinées à partir des sections interprétées.

### 1.3.5 Validation de l'interprétation

En fin d'interprétation, des cartes temps sont obtenues en effectuant un krigeage à partir des différentes sections interprétées. Le résultat de l'interprétation comporte des schémas structuraux à différents temps et de nombreuses coupes sismiques interprétées dans les trois directions de l'espace.

Une fois l'interprétation terminée et **validée en temps**, les cartes isochrones sont converties en profondeur afin de créer les données servant à construire le modèle. Dans certains cas, *10% des points* correspondant à des interprétations de failles non satisfaisants du point de vue géologique *doivent être éliminés*. Les incertitudes rencontrées durant cette étape jouent un rôle prépondérant (jusqu'à 50% dans certaines zones) dans l'estimation des volumes<sup>1</sup>.

1. Thore P., Lecour M., Shtuka A. (2000), Structural Uncertainties: Determination, Management and Applications, Geophysics (article soumis)

Ensuite, l'interprétation est **validée en profondeur** par un passage à une modélisation en trois dimensions et la vérification de la cohérence du modèle obtenu (observation à la loupe de certains endroits clés et création de coupes verticales à travers tout le modèle).

Malheureusement, *le passage en profondeur ne peut pas s'appliquer directement sur les failles*. En effet, cette opération se déroule relativement bien lorsque les couches sont quasi-horizontales, ce qui n'est bien entendu pas le cas général pour les failles. Les seules données disponibles lors de la création du modèle en domaine profondeur sont donc les puits avec quelques marqueurs (d'horizons et éventuellement de failles) et les pendages associés, et des ensembles de points représentant les horizons. Il faut donc un travail de reconstruction conséquent au moment de la modélisation en profondeur, pour retrouver le pendage réel des failles par exemple [54 - Walsh 1].

C'est à ce niveau qu'intervient le *géomodeleur*, donc assez loin dans la chaîne qui permet de passer de l'acquisition des données à un modèle géologique. C'est pourquoi les sources d'incertitude sont très nombreuses à ce niveau. Le paragraphe suivant donne un bilan des différents artefacts et problèmes rencontrés durant cette succession d'étapes.

## 2. Bilan des sources d'incertitude

Les différentes études que j'ai pu suivre ainsi que la lecture de certains rapports [70 - Thore 1], [71 - Thore 2] m'ont permis de repérer différentes sources d'incertitude. Après avoir donné quelques définitions précisant les termes liés à l'étude des incertitudes, un bilan des incertitudes apparaissant à chacune des étapes précédentes est donné en fin de chapitre

### 2.1 Quelques définitions

Comme nous allons le voir dans le chapitre 6, l'étude des incertitudes au niveau des failles est fortement liée aux statistiques. Ces quelques définitions précisent le vocabulaire qui sera employé par la suite.

**Incertitude:** *Zone de valeurs possibles* que peut prendre un paramètre. Pour un paramètre monodimensionnel, cette incertitude peut facilement être représentée par un *vecteur* (écart type  $\sigma$ ) autour de la mesure effectuée. Il est beaucoup plus difficile de représenter l'incertitude pour des objets tridimensionnels. Dans ce cas l'incertitude peut être représentée par un *ellipsoïde*.

Bien que travaillant sur des failles (objet tri-dimensionnels), nous verrons par la suite que les incertitudes que nous représentons se ramènent eux aussi à une mesure d'écart type dans la plupart des cas (cf Chapitre 6).

**Hypothèse:** passage d'une *approche probabiliste* à une *approche discrète* de l'étude des incertitudes, lorsque la mesure peut prendre un nombre limité de valeurs. Nous utilisons ce passage du domaine probabiliste au domaine discret lorsqu'un certain nombre de modèles considérés comme "représentatifs" de l'ensemble des simulations est choisi pour les simulations d'écoulement.

**Corrélation:** elle mesure la *dépendance d'une mesure* (et donc de l'incertitude associée) par rapport aux *estimations voisines* (corrélation spatiale). Elle permet donc de déterminer de combien l'incertitude locale d'une mesure est diminuée par l'estimation de la même mesure sur des points plus ou moins voisins.

**Confiance:** critère *qualitatif* qui précise si la mesure est bonne, moyenne ou mauvaise. Des cartes précisant le degré de confiance que l'on peut accorder aux données sismiques peuvent être fournies dans certains cas.

## 2.2 Incertitudes rencontrées durant la préparation des données

### 2.2.1 Acquisition et prétraitement

Durant ces premières étapes, les incertitudes sont *quasiment nulles* du point de vue structural et concernent essentiellement la qualité de l'image sismique.

### 2.2.2 Formation du stack (sommation de plusieurs traces)

C'est une étape importante puisque c'est à ce niveau que l'*image se forme*. Là encore, l'effet des incertitudes concerne surtout la netteté de l'image sismique. Cependant, c'est à ce moment que les multiples sont traités (suivant les choix effectués sur les vitesses, des horizons peuvent apparaître ou être remplacés par des multiples). Cette étape peut donc avoir une influence sur l'aspect structural de l'image.

### 2.2.3 Migration

La migration temps permet de replacer les événements dans leur position "réelle". Elle a donc un **rôle primordial** dans la détermination des formes structurales. Malheureusement, on rencontre *différentes sources d'incertitude* à ce niveau. La principale d'entre elles reste celle liée au champ des vitesses. Différentes études ont été effectuées à ce propos pour le Grand Projet Incertitudes ([66 - Faily], [67 - Juliard]) et sont reprises dans [70 - Thore 1].

Une autre source d'incertitude est due à *l'intensité des amplitudes*. Lorsque les amplitudes sont trop fortes, certaines failles peuvent être masquées [figure 3-10].

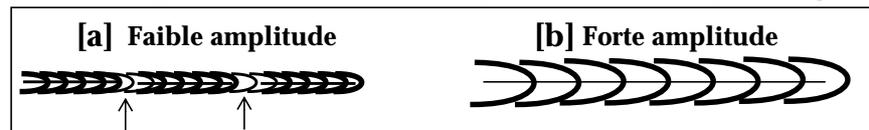


Figure 3-10 : Lorsque l'amplitude est trop forte [b], les failles, visibles en [a], peuvent être masquées.

Lorsque la *migration est mal réalisée* (notamment lorsque le champ des vitesses est connu de manière imprécise), les sections migrées peuvent poser de sérieux problèmes d'interprétation pour le néophyte [figure 3-11] [b] et [c]. Un œil expert arrive cependant à placer correctement la trace de faille en éliminant les formes "non géologiques" et en utilisant les vérifications de cohérence.

Toutefois, les erreurs sur l'estimation des vitesses croissant avec la profondeur d'une part et le pendage d'autre part, on comprend que l'interprétation des failles *en profondeur* soit un but difficile à atteindre.

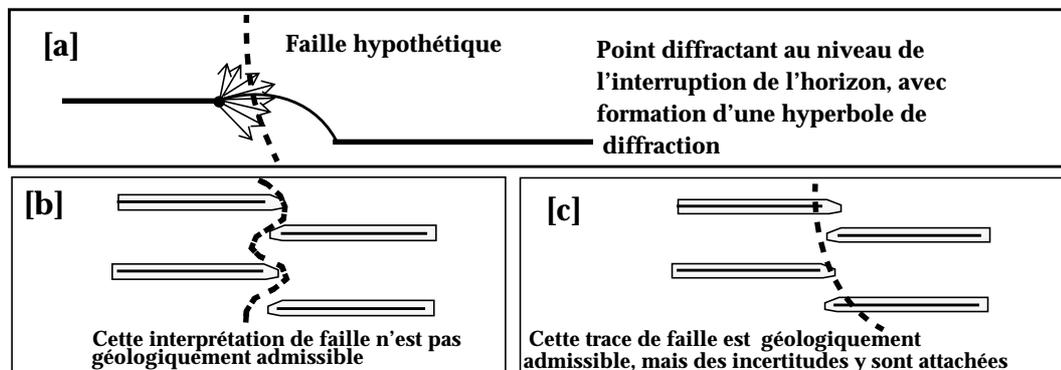


Figure 3-11 : [a] Sans la migration, les points diffractants posent d'importants problèmes; Une migration ratée ne fait pas disparaître complètement cette parabole et entraîne des incertitudes importantes pour l'interprétation des failles lorsque les horizons se chevauchent [b] et [c].

### 2.3 Incertitudes rencontrées durant la phase de calage sismique-puits

Cette étape s'effectue dans le domaine des temps migrés (les puits étant anamorphosés en temps) et suppose que la sismique migrée temps est bien positionnée. A ce stade, les *sources d'incertitudes sont nombreuses* et sont prises en compte *globalement*:

- \* la sismique peut avoir été mal migrée
- \* les puits ne sont pas forcément bien positionnés (surtout dans le cas des puits déviés)
- \* la présence de multiples peut perturber l'opération
- \* ...

**Remarque:** Dans certains cas, en dernier recours, on peut même être amené à *reje-ter un puits* s'il diffère trop des puits voisins par exemple. Dans ce cas, il est important de le spécifier afin que l'incertitude au niveau de ce puits ne soit pas considérée comme nulle en domaine profondeur.

### 2.4 Incertitudes rencontrées durant la phase d'interprétation

Cette phase est *cruciale* puisque c'est la dernière étape qui s'appuie réellement sur les données sismiques. Mais du fait de sa nature même, la détermination des incertitudes la concernant sont très difficiles à estimer.

Cette phase étant spécifique à chaque compagnie pétrolière, il existe peu de littérature la concernant. Ce paragraphe décrit succinctement les problèmes et le savoir que nous avons pu recueillir auprès des interpréteurs d'Elf Exploration Production. Cette description reste *très qualitative* et ne permettrait pour l'instant que d'avoir une estimation de la *confiance* que l'on peut accorder à telle ou telle zone sismique, d'autant plus que ce procédé n'est pas encore systématique.

De plus les sources d'incertitudes dépendent énormément du cas traité et *seul l'interpréteur* est à même de fournir cette information, pour l'instant du moins.

#### 2.4.1 Incertitudes liées à la présence même des failles

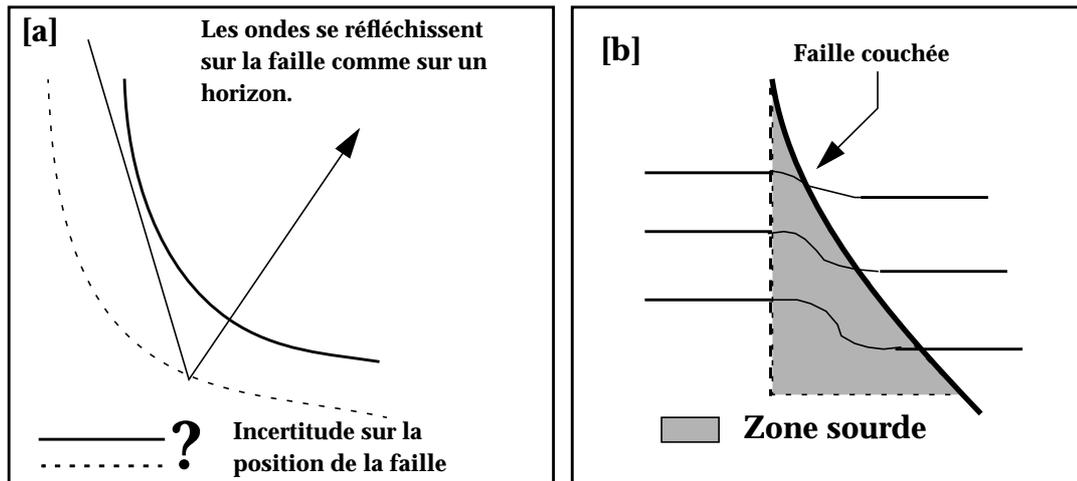
Les zones faillées sont particulièrement affectées par les incertitudes. En effet, elles ne sont la plupart du temps pas visibles directement sur la sismique, mais repérées par une *absence de marqueur*. Cependant, cette interruption du marqueur n'est pas forcément représentative de l'épaisseur réelle de la faille:

\* lorsqu'une **faille est très inclinée**, les rayons s'y réfléchissent comme sur un horizon, mais elle est reconnaissable à la grande variabilité des vitesses le long du plan de faille [47 - Sheriff]. Cela a pour effet de créer une zone sourde sous la faille. Dans cette zone, il devient extrêmement difficile de déterminer où s'arrête l'horizon et où commence la faille, donnant une zone d'incertitude parfois étendue [figure 3-12].

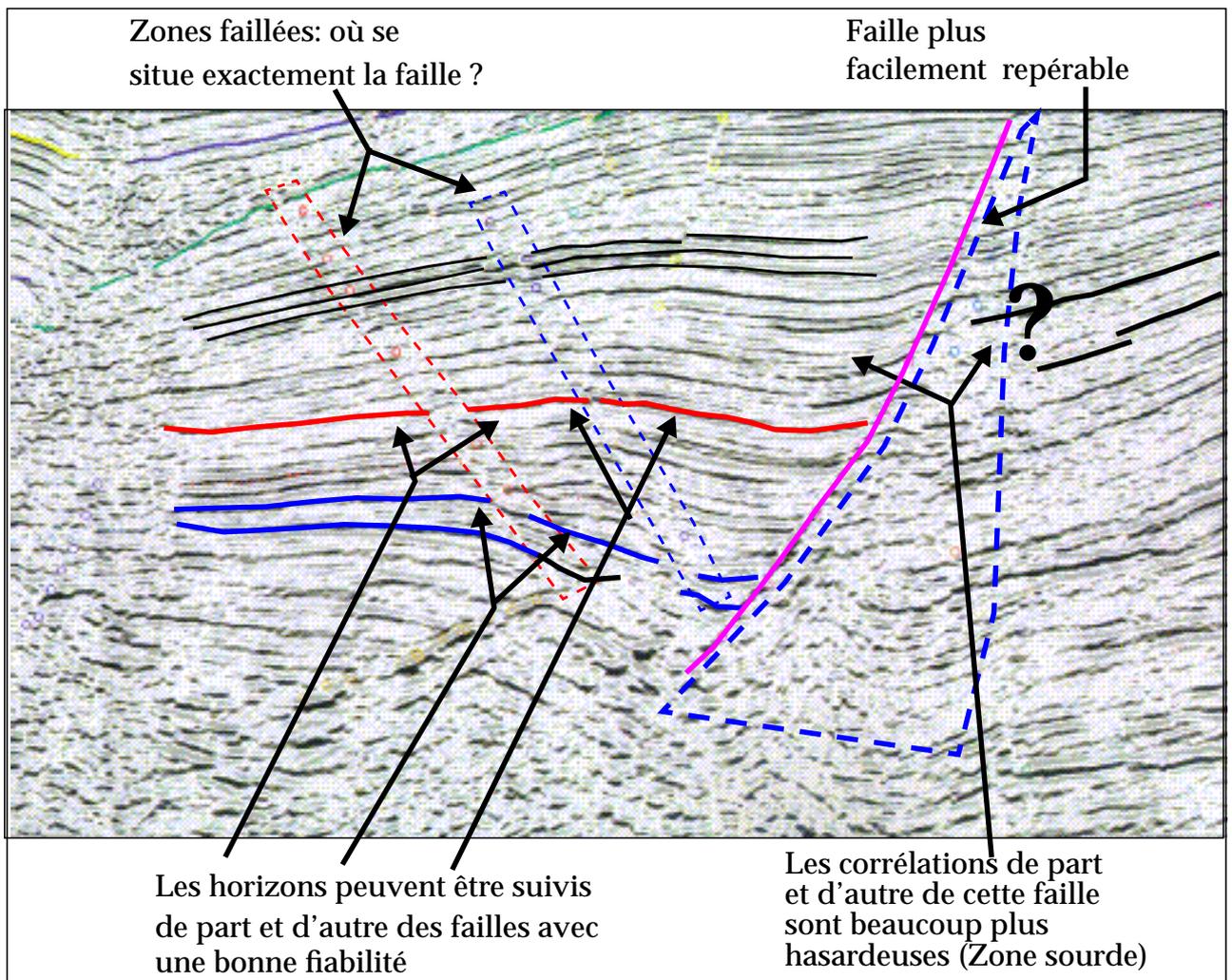
\* la faille constitue une **zone de broyage** avec de nombreux points de diffraction et la qualité de la sismique y est souvent très mauvaise [figure 3-11] [a] et [figure 3-13].

A cela s'ajoute le fait que, lorsque *plusieurs failles* s'intersectent, il est très difficile de trouver un plan d'interprétation perpendiculaire aux structures [figure 3-6].

Enfin, lorsqu'à cause d'une faille, l'interpréteur *change de panneau*, il est parfois amené à faire un choix pour retrouver le même marqueur de l'autre côté de la faille. Cependant, l'incertitude qui en découle est quasi nulle si le marqueur considéré est bien repérable (la BCU, ou " *Base Cretaceous Unconformity*", en est un exemple) [figure 3-13].



**Figure 3-12 : Incertitudes liées à la présence de failles couchées. [a]** les ondes se réfléchissent comme sur les horizons et créent une **zone sourde** où l'interprétation est rendue difficile **[b]**. Ce phénomène est encore amplifié lorsque des **hydrocarbures** sont situés d'un côté de la faille (forte amplitude --> forte réflexion).



**Figure 3-13 : Incertitudes liées à un changement de panneau (d'après [70 - Thore 1]).**

### 2.4.2 Incertitudes liées à la lithologie

Un autre source d'incertitude est liée à la lithologie et aux éléments contenus dans les roches. Dans les zones constituées de *sel ou d'argile*, qui sont de très mauvais réflecteurs, les traces sont relativement brouillées et rendent l'interprétation difficile (on retrouve le même genre d'images que sous les failles couchées) [figure 3-14].

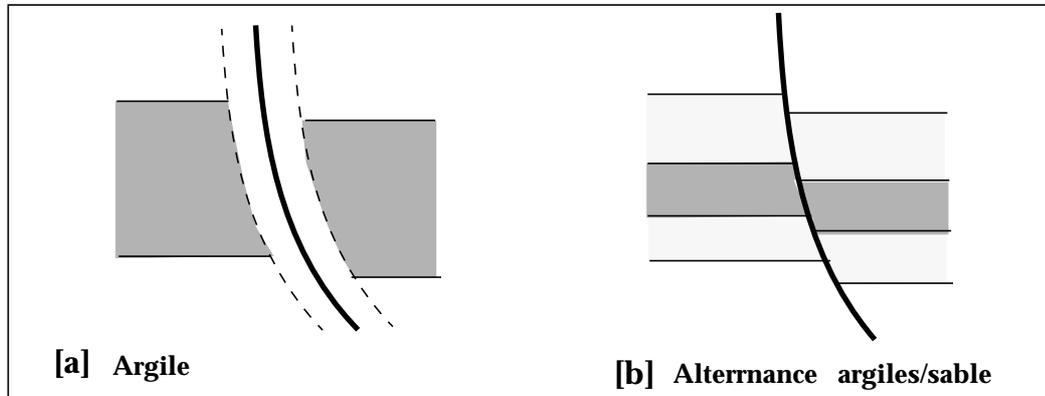


Figure 3-14 : La lithologie joue un grand rôle dans la qualité de la sismique. [a]: les argiles sont de mauvais réflecteurs; [b]: lorsqu'il y a une alternance argile/sable, l'image sismique est meilleure.

De la même façon, les couches des *zones chenalisées* s'entrecroisent et rendent l'interprétation délicate (d'autant plus que le long d'un chenal, la sismique change parfois de caractère, du fait des variations de propriétés physiques des roches, créant ainsi de "*fausses failles*").

Enfin, la *présence de gaz* (qui donne des zones à très forte amplitude) brouille aussi les images et rend difficile l'interprétation des coupes à leur niveau [figure 3-15].

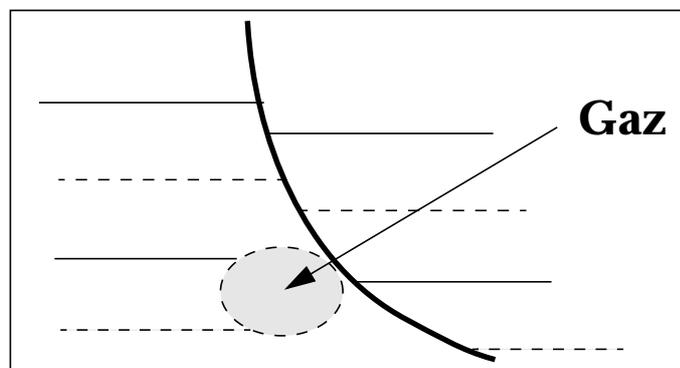


Figure 3-15 : La présence d'hydrocarbures peut détériorer la qualité de la sismique.

## 3. Conclusion sur les incertitudes autour des failles

Le nombre des opérations qui se sont déroulées auparavant étant aussi élevé, on comprend qu'au moment de la modélisation en trois dimensions, le **domaine d'incertitude** soit parfois *étendu* et que le *quantifier* assez précisément ne soit pas toujours très aisé.

En effet, les incertitudes peuvent apparaître à chacune des étapes, mais en fonction du cas étudié, il faut être capable de sélectionner celles qui vont avoir une *influence prépondérante*.

Un tableau récapitulatif permet d'avoir une vue d'ensemble sur les différentes sources d'incertitude apparaissant au cours de la chaîne de traitement, ainsi que du type de problème rencontré à chacune des étapes [figure 3-16]:

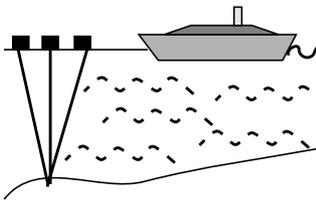
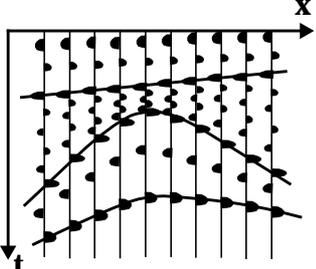
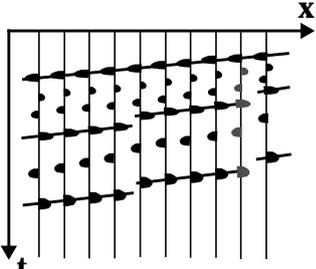
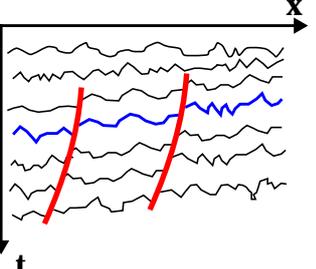
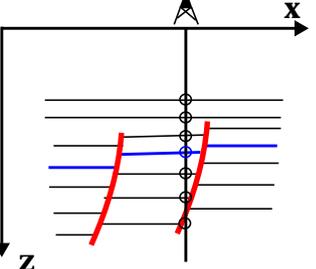
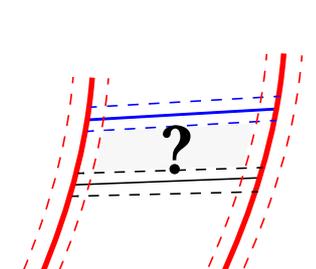
<b>Acquisition</b>	<b>Préprocessing - stack</b>	<b>Migration Temps</b>
		
Positionnement	Résolution, multiples ...	Estimation des vitesses $\Delta V / V$
Globale faible	Globale / Locale faible / (forte)	<b>Locale moyenne</b>
<b>Interprétation (Pointé)</b>	<b>Conversion temps/profondeur</b>	<b>Estimation du volume de roche</b>
		
Rapport Signal/Bruit calage aux puits, interprétation	Manque de puits $\Delta V / V$ du stack	
<b>Multiple forte</b>	<b>Locale faible à forte</b>	<b>Géostatistiques</b>

Figure 3-16 : Bilan incertitudes au cours de la chaîne de traitement (d'après [70 - Thore 1])

Une des sources d'incertitude non illustrée dans ce tableau mais qui n'en reste pas moins primordiale, découle de la distribution des différentes étapes de la chaîne de traitement entre des équipes distinctes. Ainsi, les choix effectués au cours des étapes précédentes ne sont pas toujours pris en compte dans la suite de la chaîne [64 - Basire]. Savoir que la sismique est très mauvaise dans telle zone, ou que les vitesses ont été difficiles à estimer pour telle autre zone permettent de donner plus de libertés lors de la modélisation. Par contre, si les données sont considérées comme sûres, il faut absolument les respecter lors de la modélisation.

Des projets comme **POSC** (the Petrotechnical Open Software Corporation) et **OpenSpirit** se mettent en place actuellement pour améliorer cette communication entre les différentes applications, à la demande de nombreuses compagnies pétrolières.

Un des buts du projet Open Spirit est de pouvoir à terme interagir directement avec des stations d'interprétation ou des bases de données (*Openworks* [base de données Landmark], *Seiswork* [station d'interprétation Landmark] et *IESX* [station d'interprétation de Schlumberger]), sans avoir à perdre du temps pour transformer les données d'un format à l'autre, et ainsi avoir un vrai "Modèle partagé" entre les différentes applications (stations d'interprétation sismique, modeleurs structuraux, mailleurs réservoirs et simulateurs d'écoulement) [16 - Schenck], [17 - Tipee].

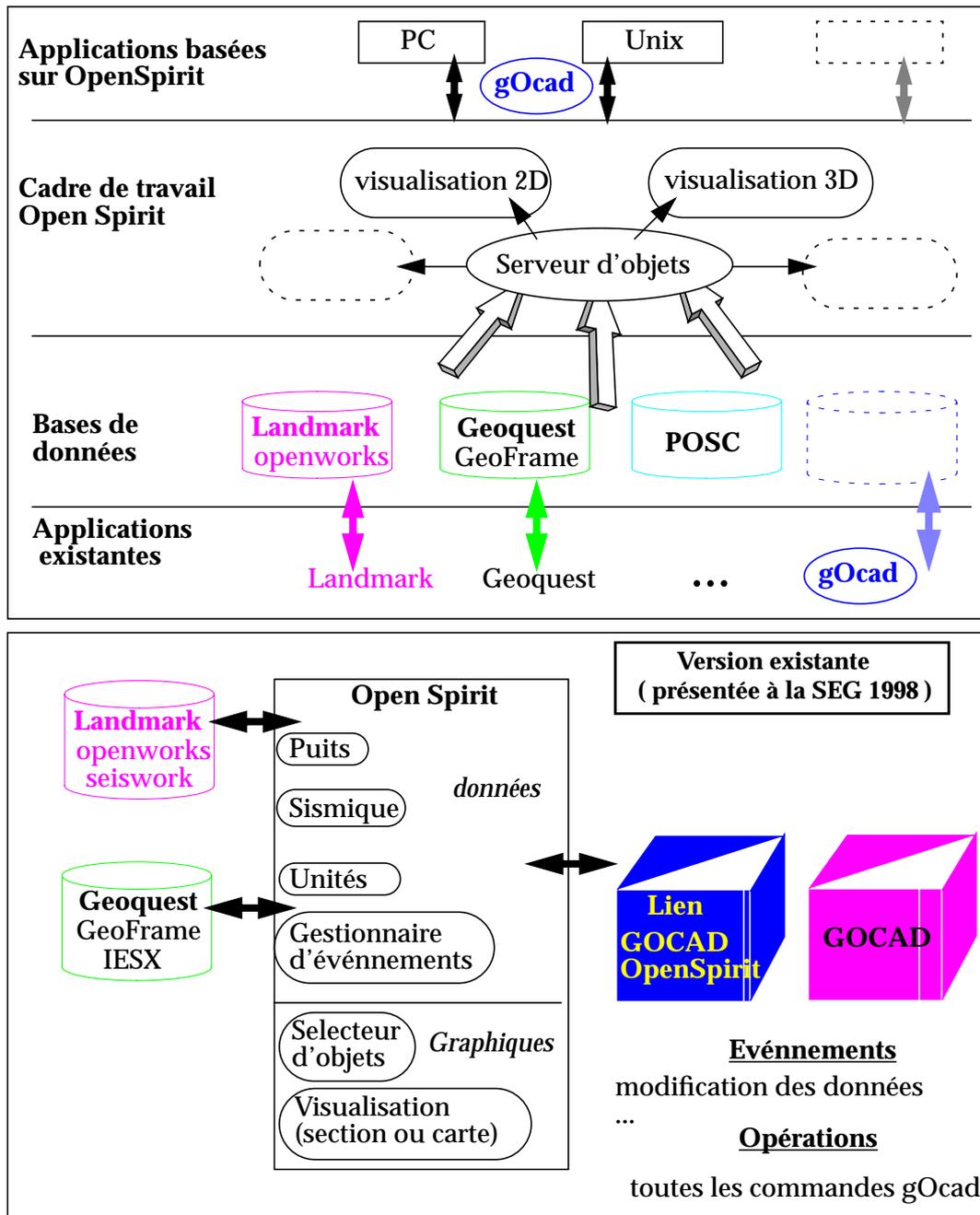
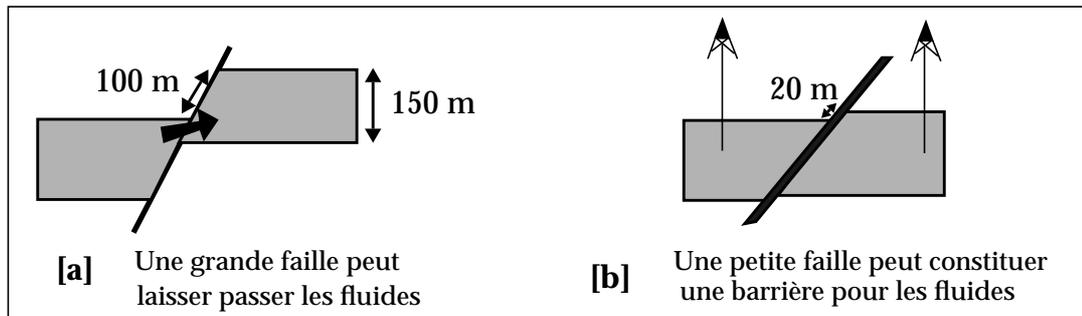


Figure 3-17 : Le projet OpenSpirit permettra de partager le modèle entre différentes applications et différents métiers (d'après [74 - WWW 3]).

Enfin, en fonction du problème étudié, les points auxquels il faut faire particulièrement attention lors de l'étude des incertitudes peuvent être différents. Par exemple, en ce qui concerne les failles:

- pour l'étude du **volume de roches**, les failles auront la plupart du temps un rôle secondaire comparé aux horizons (excepté pour les failles bordières)
- pour l'étude des **réserves** par contre elles ont un rôle prépondérant (puisque en fonction de leurs propriétés physiques, elles peuvent constituer soit des barrières soit des passages pour les fluides) [figure 3-18].



**Figure 3-18 : Les failles jouent un rôle prépondérant dans le calcul des réserves.** En effet, même une grande faille peut correspondre à un couloir de **passage pour les fluides** [a]; au contraire, une faille à faible rejet et donc difficilement repérable sur la sismique peut constituer une **barrière** [b]. Ce type de failles est repérable lorsque les pressions mesurées par 2 puits voisins sont très différentes.

## 4. Techniques existantes pour étudier les incertitudes structurales

Des techniques existent déjà pour étudier l'impact des incertitudes dues aux vitesses mal estimées et au pointé des horizons sur l'évaluation du volume de roche [68 - Samson].

### 4.1 Méthode employée pour les horizons

Cette méthode suppose qu'une surface de référence correspondant à la position la plus probable de l'horizon ainsi que des cartes d'incertitudes sont fournies. Différentes variations de la méthode sont détaillées dans l'article, mais peuvent être résumées comme suit.

- **Estimation de vecteurs d'incertitudes**

Des vecteurs d'incertitude sont attachés aux différents points de la surface de référence. Ils correspondent aux cartes d'incertitudes estimées à partir de la qualité de la sismique par exemple. Dans le cas où l'horizon est faillé (seules les failles verticales sont traitées), ces vecteurs sont pris verticaux afin d'éviter d'avoir à vérifier la cohérence du modèle après chaque simulation.

- **Sélection aléatoire de certains points**

Un certain nombre de points sont sélectionnés aléatoirement sur la surface, mais choisis en fonction d'une longueur de corrélation ( $R$ ) donnée au préalable. Deux points sélectionnés sont distants au minimum d'une distance  $R$ , et au maximum d'une distance  $2R$ .

Pour tous ces points, un nombre aléatoire ( $p$ ) est choisi dans l'intervalle  $[-1, 1]$ . L'erreur de position correspondant à ce point ( $e$ ) est calculée par la multiplication de  $p$  par le vecteur d'incertitude correspondant au point choisi.

- **Génération d'une surface simulée**

Pour tous les points non sélectionnés dans la phase précédente, une valeur d'erreur  $e$  est calculée par interpolation. Pour chaque point de la surface de référence, une position simulée est calculée en additionnant l'erreur  $e$  à la position de référence.

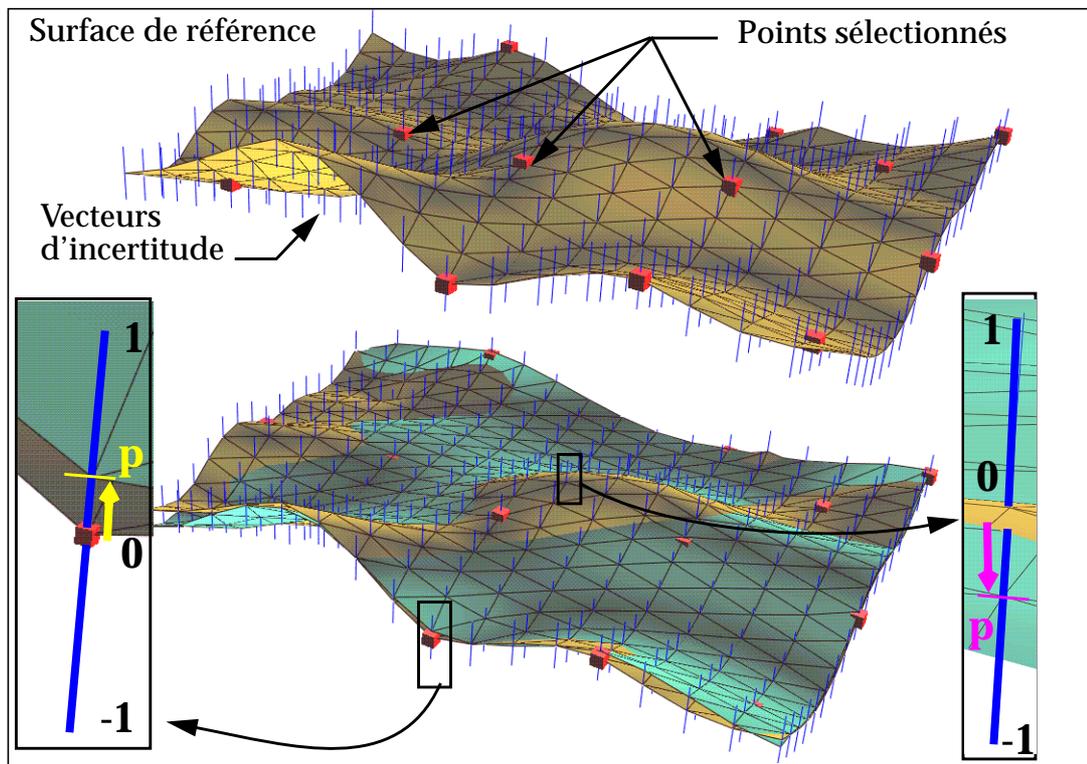


Figure 3-19 : Calcul d'une surface d'horizon "plausible" compte tenu d'incertitudes estimées au préalable et connaissant la surface de référence (la "plus probable").

#### 4.2 Tenir compte des puits

Une amélioration de la méthode a été proposée afin de tenir compte des données de puits [69 - Shuguang]. En effet, si l'on considère que la surface de référence a été calée aux puits de manière satisfaisante, la valeur de l'incertitude au niveau des intersections horizon-puits doit être nulle. Mais les puits donnent beaucoup plus d'information: le fait qu'un puits horizontal traverse complètement une couche, ou qu'au contraire il ne l'intersecte jamais, sont des informations fondamentales sur la position réelle des horizons. Un pré-traitement des vecteurs d'incertitude permet donc de limiter les valeurs issues de l'estimation faites par les géophysiciens à partir des seules cartes de qualité sismique et de tenir compte des informations données par les puits [figure 3-20].

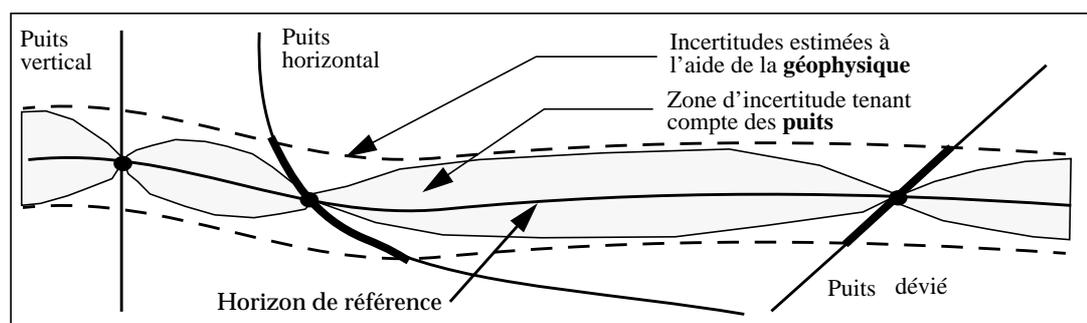


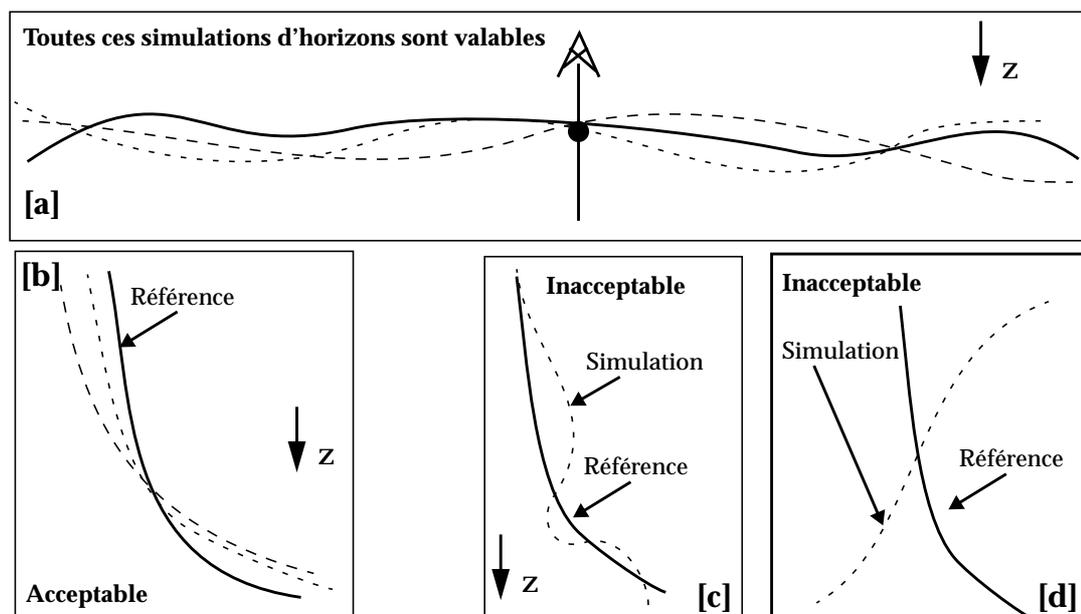
Figure 3-20 : Pré-traitement des vecteurs d'incertitude pour tenir compte des puits (d'après [69 - Shuguang]).

### 4.3 Pourquoi cette méthode n'est pas directement applicable aux failles

Comme cela a été montré dans le chapitre 2, la géométrie des failles est régie par beaucoup plus de *règles mécaniques* que celle des horizons. Même si ces règles ne sont pas toujours exprimées sous forme mathématique, il faut garder à l'esprit que les simulations de failles doivent rester cohérentes du point de vue géologique. Le nombre de simulations acceptable pour les failles étant bien inférieur à celui des horizons [figure 3-21], la méthode précédente, mise au point pour les horizons, doit être modifiée pour simuler les failles.

Les failles étant complètement reconstruites en domaine profondeur, il est en outre quasiment impossible de se reporter aux données géophysiques du départ pour contrôler la validité de la modélisation une fois la faille construite. Il faut tenir compte des incertitudes accumulées tout au long de la chaîne de traitement, mais sans pouvoir s'y référer. C'est pourquoi nous considérerons par la suite que ces **incertitudes** ont été **estimées au préalable** lors des différentes phases de l'interprétation. De plus, comme cela a été montré au début de ce chapitre, les incertitudes au niveau des failles sont plus difficiles à estimer que celles autour des horizons principaux, puisque les zones de failles sont les plus bruitées sur la sismique. L'estimation des incertitudes autour des failles est donc "*moins certaine*" que l'estimation des incertitudes au niveau des horizons. De plus, les méthodes de pré-processing utilisées dans le cas des horizons sont beaucoup plus difficilement utilisables dans le cas des failles. En effet, il est rare que les puits intersectent des zones faillées (les zones de faille sont des zones à éviter en général), et lorsque l'on a la chance d'avoir un puits traversant une faille, il ne la traverse en général qu'une seule fois.

Enfin, si l'on peut étudier l'impact des incertitudes affectant le pointé d'un ensemble d'horizons *non faillés* sur le calcul du volume de roche, il est indispensable de **gérer les contacts horizon-faille** lorsque l'on veut étudier l'impact des incertitudes au niveau des failles sur ce calcul du volume de roche. Cette tâche est loin d'être facile dans le cas général [22 - Abrahamsen], mais beaucoup plus aisée en utilisant le système des contraintes existant dans le logiciel gOcad. Cet aspect sera détaillé dans le chapitre 7 de ce mémoire.



**Figure 3-21 :** Le nombre de simulations acceptables pour les failles [b] est moins important que le nombre de géométries acceptables pour les horizons [a]. Nous avons décidé de les éliminer lorsque la forme est **trop irrégulière** [c] ou que le **pendage s'inverse** par rapport à la faille de référence [d].

C'est pour étudier les zones délicates à interpréter que sont les zones faillées qu'un **objet faille** a été créé dans le logiciel gOcad.

La partie suivante montre comment cet objet a été prévu pour modéliser les failles à partir des données disponibles (sismique migrée temps puis passée profondeur ou directement migrée profondeur; puits; et éventuellement coupes interprétées), ainsi que de pouvoir associer en chacun des points une valeur de l'incertitude (soit fournie par des estimations préalables, soit interpolée à partir de ces dernières).

Ainsi, même si l'estimation des incertitudes n'est pas encore très évoluée en ce qui concerne les failles, une méthodologie permettant de tenir compte de ces incertitudes a pu être mise au point pour les incorporer dans le calcul des volumes de roche. Cette méthodologie est détaillée dans les prochains chapitres. De cette façon, toute la chaîne située en aval est prête et sera utilisable dès que les calculs d'incertitude au niveau des failles seront plus précis.

---

---

## Chapitre 4 Paramétrisation de l’Objet Faille:

### Un objet simple incorporant des données diverses

---

---

Durant la première partie, il a été souligné à quel point la modélisation précise des failles est essentielle lors de la phase d’exploration.

La **modélisation géométrique** des failles n’est malheureusement pas une tâche aisée. Comme cela a été montré durant le chapitre précédent, ceci est dû en particulier à la difficulté de disposer de *données fiables*, non seulement sur la position exacte de la faille, mais aussi sur sa forme [64 - Basire].

En effet, les failles sont reconnues de manière *indirecte*, grâce aux perturbations qu’elles engendrent dans les données sismiques: décalage des horizons, mais aussi zones sourdes ou difficilement interprétables.

Les données utilisées pour modéliser les failles étant beaucoup plus *imprécises* et *moins nombreuses* que celles utilisées pour construire les horizons, les outils classiques utilisés pour ces derniers (surfaces triangulées contraintes par des ensembles de points relativement denses) ne sont pas forcément adaptés à la modélisation des failles. C’est pourquoi la définition d’une **nouvelle structure de données** a été réalisée tout au début de la thèse.

Un premier paragraphe décrit les **données utilisables** pour modéliser les failles. Après avoir constaté que ces données étaient composées essentiellement de *lignes polygonales*, la paramétrisation de cette structure a été spécialement adaptée à ce type de données.

Un second paragraphe décrit plus précisément la **paramétrisation choisie** pour ce nouvel objet “Faille”, dont un prototype performant a été implanté par *Richard Cognot* au démarrage de ce travail. Ce prototype a permis d’avancer plus vite par la suite. Cette paramétrisation a été choisie de façon à ce que la structure soit *précise et flexible* pour incorporer toutes les données, et permettre une édition rapide lorsque de nouvelles données sont disponibles; tout en restant *très légère* afin de pouvoir simuler rapidement un grand nombre de géométries probables.

Outre la modélisation rapide de la position la plus probable de la faille, cette structure de données permet en effet de définir un intervalle d’incertitude variable autour de cette position et d’effectuer plusieurs *simulations de la géométrie d’une faille* donnée.

Le dernier paragraphe décrit plus précisément comment cette structure utilise la notion d’héritage du langage C++ et bénéficie d’un grand nombre de **fonctions d’édition** des lignes polygonales existant dans gOcad<sup>1</sup>.

---

*1. Dans un souci de clarté, les extraits de codes décrits dans ce chapitre ont été simplifiés et dépouillés au maximum. Le fait que le logiciel avec lequel ce travail a été réalisé ait une interface écrite en anglais, explique que certains termes anglais se glissent parfois dans les explications. Cependant, une traduction est proposée la plupart du temps.*

## 1. Données disponibles

### 1.1 Description des données utilisables pour construire les failles

Lors de la phase finale de la construction d'un modèle en trois dimensions, les données utilisables pour modéliser les failles sont de différents ordres [figure 4-2]:

- **les horizons migrés temps**

Ces ensembles de points sont le résultat d'un krigeage des courbes interprétées à partir des données sismiques, puis d'un passage du domaine temps vers le domaine profondeur. Ils donnent donc globalement la position des failles, mais on ne peut considérer leurs bords comme les limites exactes des traces de faille, surtout dans les zones de branchement entre failles. En effet, lorsque la sismique devient trop floue, l'interpréteur prudent stoppe son interprétation à la zone interprétable. Au contraire, les méthodes d'interprétation automatique (*autopicking*) poussent l'interprétation au maximum, allant parfois trop loin dans les zones faillées.

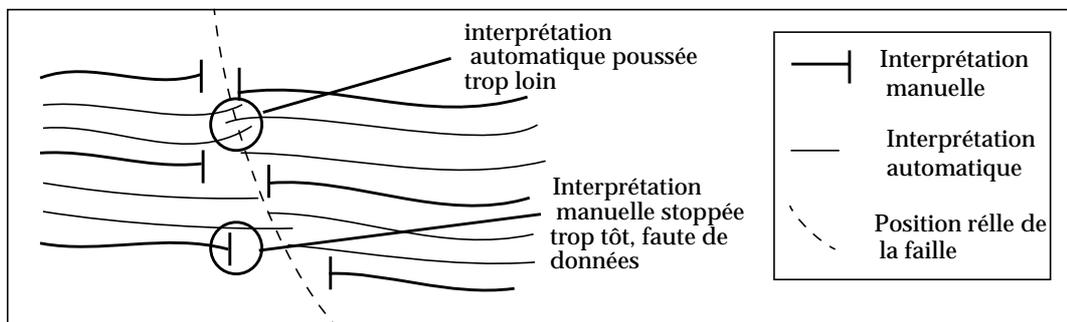


Figure 4-1 : Les bords des horizons interprétés ne correspondent pas forcément aux failles.

- **les polygones de failles**

Ces courbes ne sont pas systématiquement tracées dans toutes les compagnies pétrolières. Elles correspondent à l'interprétation de la position la plus probable de la faille en carte, selon le structuraliste. Ces polygones sont donc plus réduits et a priori *plus fiables* que les vides correspondants dans les ensembles de points.

- **les interprétations de failles**

Ce sont des courbes qui correspondent soit à des interprétations de failles en domaine temps, puis passées en domaine profondeur, soit à des coupes validées lors de la vérification de la cohérence du modèle en domaine profondeur.

Ce sont donc en théorie les *données les plus fiables*. Malheureusement, ces données sont *peu nombreuses* (et non fournies dans la majorité des jeux de données mis à ma disposition).

- **les puits**

Les puits sont placés de façon à intersecter les failles le moins possible, puisque ces zones sont souvent plus délicates à interpréter. Il est donc relativement *rare* d'avoir des indications de pendage données par des puits.

Bien que les jeux de données dont nous disposons ne contiennent pas de telles données, il arrive qu'elles soient disponibles.

- **le concept géologique**

C'est la partie pour laquelle l'*expérience du structuraliste* est indispensable. Bien que difficile à définir, ces données font partie des plus importantes. Notre principal soucis a été de les respecter au maximum durant les simulations.

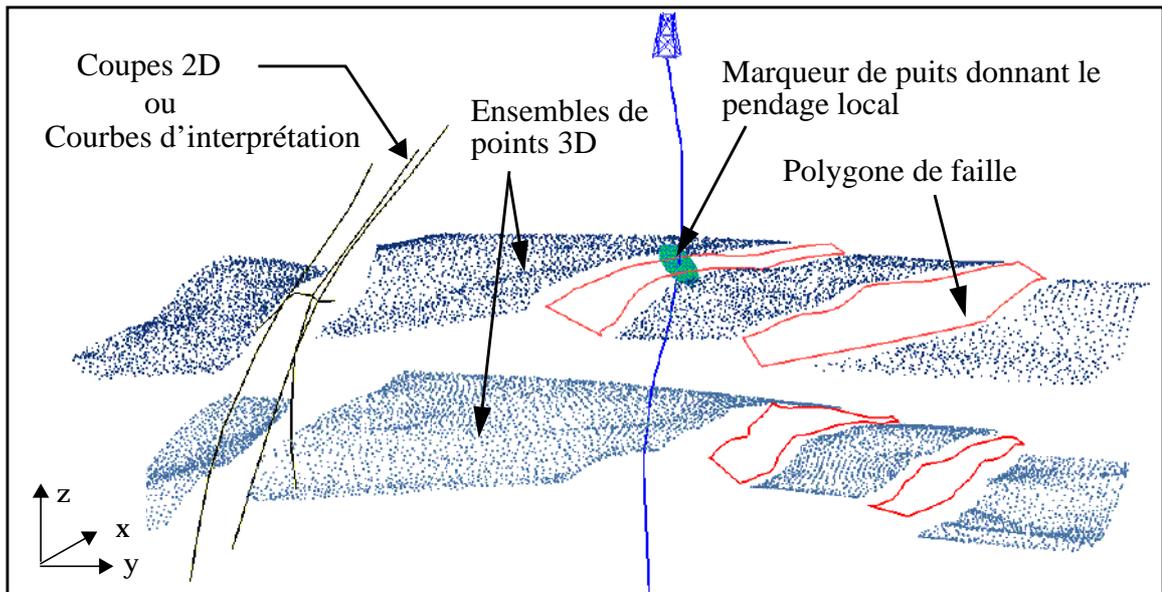


Figure 4-2 : Données disponibles pour construire les failles

## 1.2 Caractéristiques de ces données

Lorsque l'on analyse les données disponibles, on remarque rapidement qu'elles sont composées pour la plupart de **courbes**. Ces courbes sont le plus souvent en trois dimensions (elles sont non planaires, sauf si elles sont issues d'interprétations de coupes). Elles peuvent être allongées soit suivant l'axe horizontal (bords d'horizons, polygones de failles), soit selon l'axe vertical (interprétations de faille).

Il existe un troisième type de courbes (ni sub-horizontales, ni sub-verticales). La dernière partie de ce chapitre montre comment ces courbes sont prises en compte.

De plus, on remarque qu'on ne peut leur accorder la même confiance: les courbes horizontales s'étendent sur la quasi-totalité de la faille mais sont relativement peu précises, tandis que les courbes verticales sont beaucoup moins nombreuses, plus espacées, mais sont plus précises en ce qui concerne la géométrie de la faille. Il existe donc une certaine asymétrie dans les données.

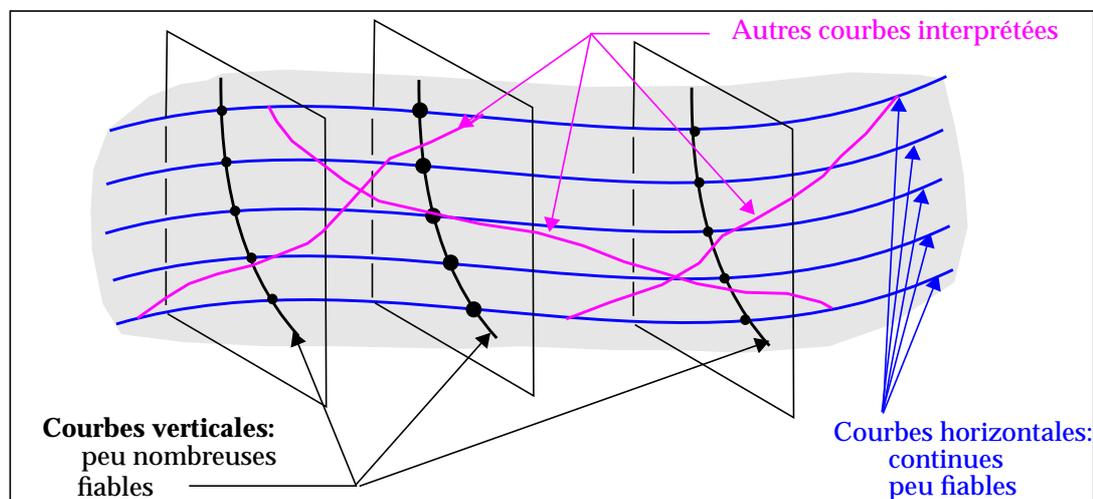


Figure 4-3 : Différentes courbes disponibles pour modéliser les failles

## 2. Présentation de l'Objet Faille

Pour tenir compte de l'asymétrie des données, et incorporer directement ces lignes dans la modélisation, une nouvelle structure a été mise au point par Richard Cognot et Pierre Thore [75 - Cognot 2], puis remaniée durant la thèse.

Cet objet correspond à une paramétrisation très simple de la géométrie des failles, composée uniquement de courbes. Afin de pouvoir utiliser au maximum les fonctionnalités des lignes polygonales de gOcad (cet aspect sera détaillé en fin de chapitre), ces courbes sont de deux sortes [figure 4-4]:

- une **courbe principale** constituant la colonne vertébrale de la faille ("*backbone*")
- un ensemble de **courbes secondaires**, portées chacune par un nœud de la courbe principale ("*génératrices*").

Cette paramétrisation est fortement inspirée de l'objet **GShape** ("Geological Shape") défini par L. Wietzerbin et utilisé dans gOcad pour modéliser les chenaux entre autres [85 - Wietzerbin ]. La différencenotable entre ces deux objets est que pour le premier, les courbes secondaires sont des courbes fermées (sections), alors que pour modéliser les failles, nous utilisons des courbes ouvertes (génératrices) [figure 4-4].

D'autres différences existent (amélioration de la manière dont les points des courbes secondaires sont stockés, afin de tenir compte de failles particulières, ajout de fonctionnalités spécifiques au stockage des incertitudes). Ces points seront développés ultérieurement.

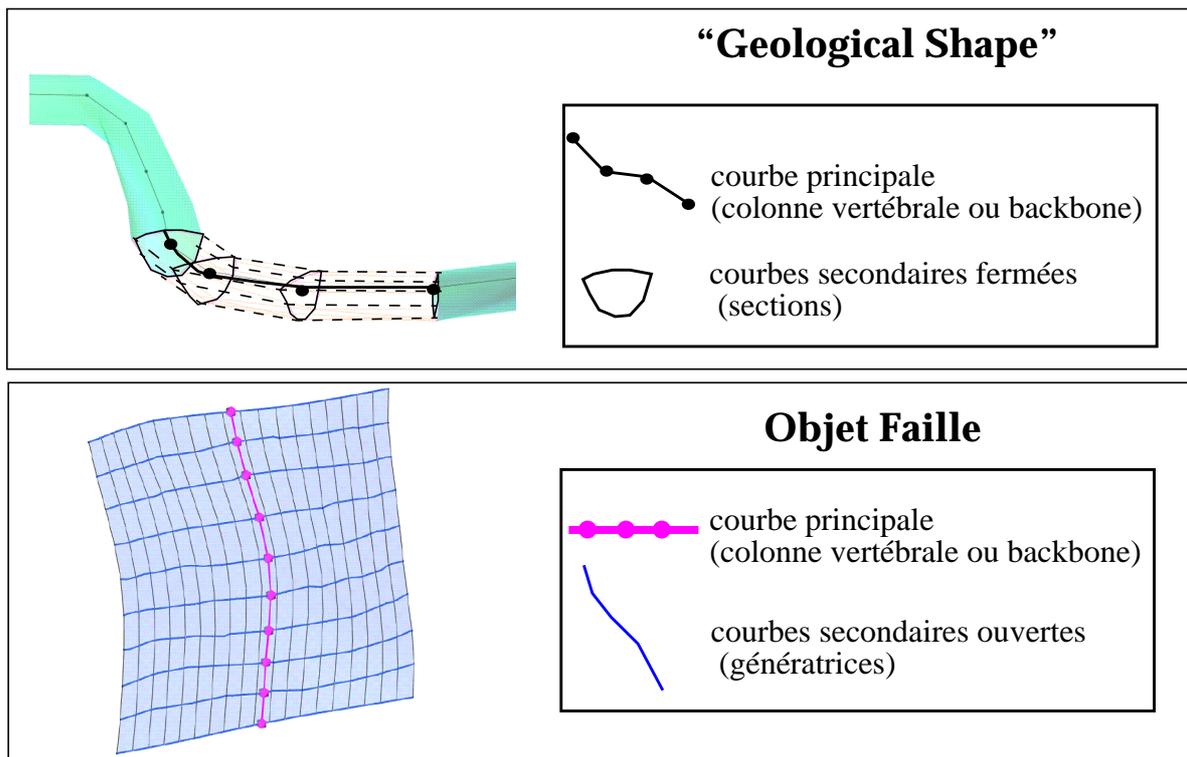


Figure 4-4 : La paramétrisation de l'objet faille est fortement inspirée de l'objet GShape décrit par Liliane Wietzerbin [85 - Wietzerbin ].

**Remarque:** selon que les données sont constituées d'un ensemble de sections verticales interprétées ou d'un ensemble d'horizons, la courbe principale sera horizontale ou verticale.

Il est possible de passer facilement d'une configuration à l'autre pour tenir compte à la fois des deux types de données si celles-ci sont disponibles.

## 2.1 La colonne vertébrale (Backbone)

Pour gOcad, l'objet faille est homomorphe à une ligne polygonale: la colonne vertébrale (ou backbone). Ce type d'objet existe depuis longtemps dans le logiciel et contient un très grand nombre de fonctionnalités que nous pouvons utiliser par héritage.

### 2.1.1 Les lignes polygonales

Comme cela a été brièvement présenté dans le premier chapitre, l'approche du logiciel gOcad, est de définir des objets simples (points, lignes, surfaces....) composés de nœuds connectés par des segments, triangles, etc.

Les lignes polygonales (appelées **PLINE** ou Polygonal Line dans le logiciel gOcad) sont ainsi constituées d'un ensemble de nœuds connectés par des segments. L'objet Faille étant fortement lié à ces lignes, le paragraphe suivant les présente brièvement.

#### a) Un ensemble de nœuds connectés

Pour gOcad, les nœuds constituant le maillage des objets sont appelés "ATOM". Dans le cas général, un ATOM contient un certain nombre d'informations:

- ses **coordonnées géométriques** dans l'espace à trois dimensions ( $x, y, z$ ) (elles sont définies dans la classe POINT3D)
- à **quel(s) objet(s) il appartient** (un même point peut en effet être "partagé" par différents objets). Ceci est très utile pour répercuter les modifications géométriques d'un objet simple sur lequel on peut travailler rapidement vers un autre objet plus complexe (ce stockage d'information provient de la classe VERTEX).
- la **connaissance des nœuds voisins**. Cette notion *topologique* est fondamentale. Elle permet de sélectionner rapidement un certain nombre de voisins par auroles successives (la classe NODE permet de garder cette information).
- les différentes **propriétés** stockées dans l'objet (porosité, perméabilité, distance à un point, erreur d'interpolation par rapport à une contrainte donnée....), ainsi que les **contraintes** qui lui sont attachées. Ces contraintes sont utilisées lors de l'interpolation par D.S.I (c'est la dernière classe, héritant des autres: class ATOM).

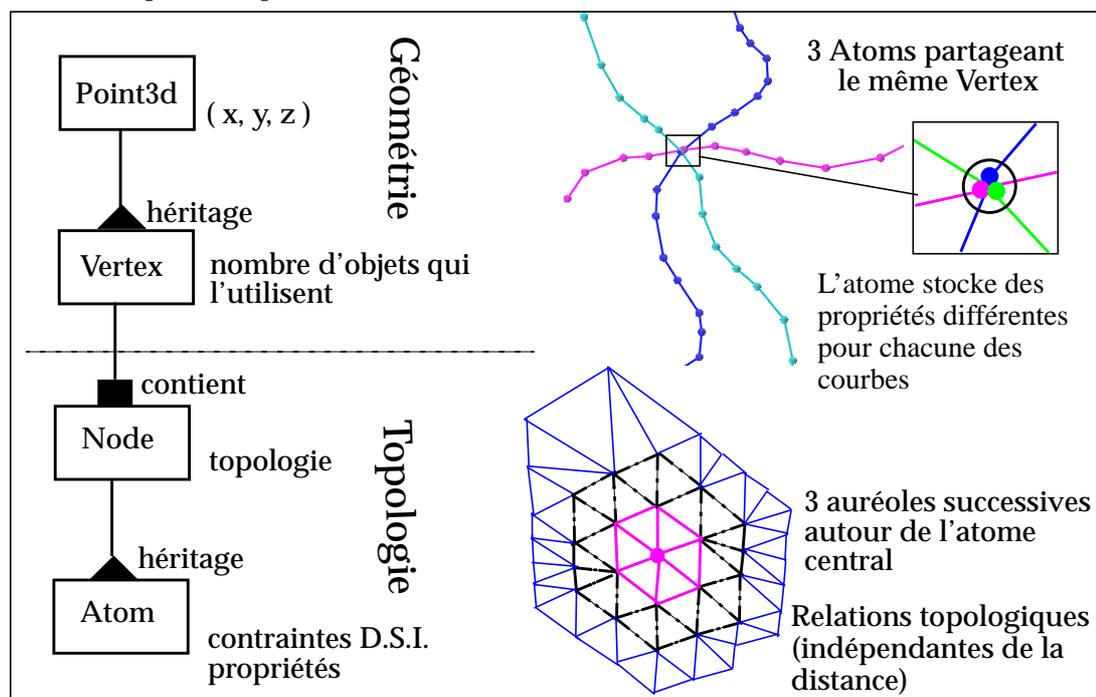


Figure 4-5 : Séparation de la géométrie et de la topologie (Un Node contient un Vertex mais ne dérive pas de la classe Vertex).

Dans le cas où l'objet est une ligne polygonale, le *nombre de voisins* de chacun des atomes est *limité à deux* au maximum. Un atome possédant quatre voisins est un cas non valide. Nous verrons plus tard en quoi ceci est une limitation pour l'objet faille.

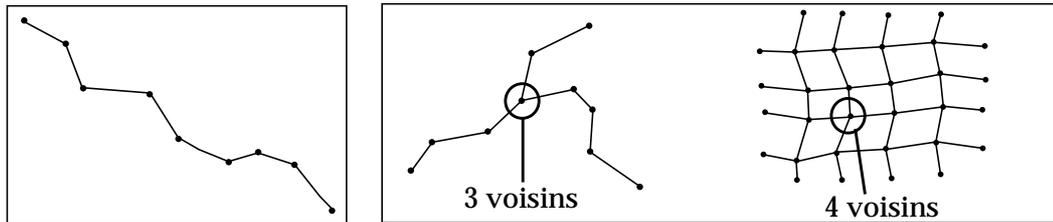


Figure 4-6 : [a]: courbe valide; [b]: topologies non valides (plus de 2 deux voisins pour certains nœuds)

## b) Atomic / AtomicGroup

Comme cela a été souligné plus haut, beaucoup d'objets de base dans gOcad sont constitués d'atomes. Pour cela, ils sont appelés ATOMIC. Un ATOMIC est une collection d'atomes (connectés la plupart du temps).

On distingue pour chaque type d'objet, l'ATOMICGROUP des ATOMICs qu'il contient. Un ATOMIC est constitué d'un ensemble d'atomes connectés, à l'exception des ensemble de points. Cette notion permet entre autres de *regrouper sous le même nom* plusieurs parties qui correspondent au **même objet géologique** (horizon coupé par une faille par exemple) [figure 4-7].

Tableau 1: Relation Atomic / AtomicGroup

Objet	Atomic	Signification	AtomicGroup	Signification
Points	SubVSet	Sub-Vertex Set	VSet	Vertex Set
Courbes	ILine	Isolated Line	PLine	Polygonal Line
Surfaces	TFace	TriangulatedFace	TSurf	Triangulated Surface
Objet Faille	GFault	Geostatistical Fault	GFaultBundle	Geostatistical FaultBundle

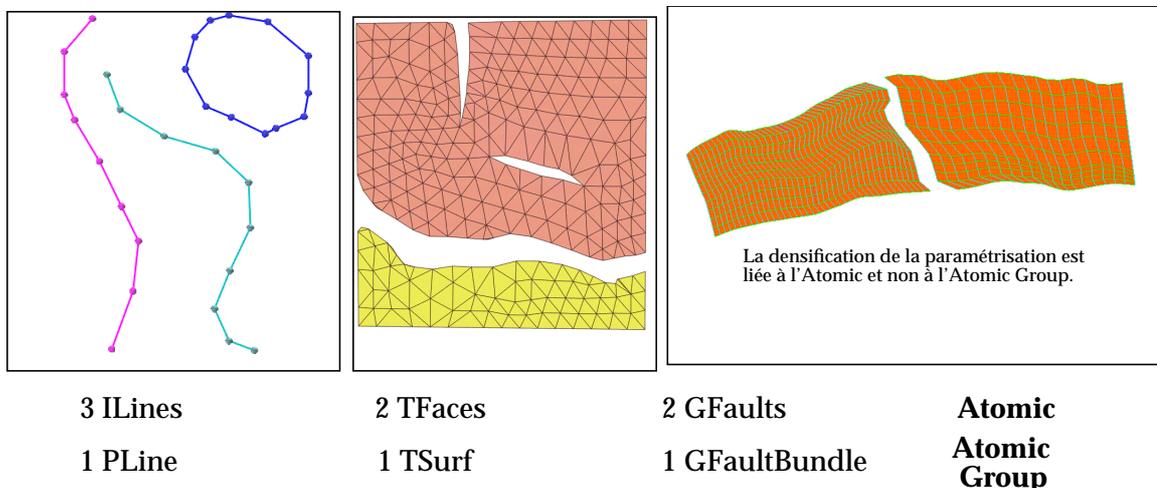


Figure 4-7 : Quelques exemples d'Atomic / AtomicGroup.

Cette classification permet en outre de gérer les **propriétés** de manière *globale* sur les objets. Tous les ATOMICS d'un ATOMICGROUP stockent le *même nombre de propriétés*. Ces propriétés sont sous la responsabilité du même gestionnaire. Ce dernier permet de mettre en relation une propriété, son nom et un index dans la liste de propriétés.

Ainsi, les propriétés stockées sur les différents atomes sont toujours stockées dans le *même ordre* et peuvent être repérées par un *indice* dans la liste des propriétés. Cette particularité a une importance fondamentale pour la gestion de l'objet Faille, comme nous le verrons ultérieurement. Ces propriétés peuvent être scalaires ou vectorielles. Pour tenir compte des propriétés vectorielles deux indices existent: un indice par propriété (*Property Index*) et un indice global (*Property Record Index*).

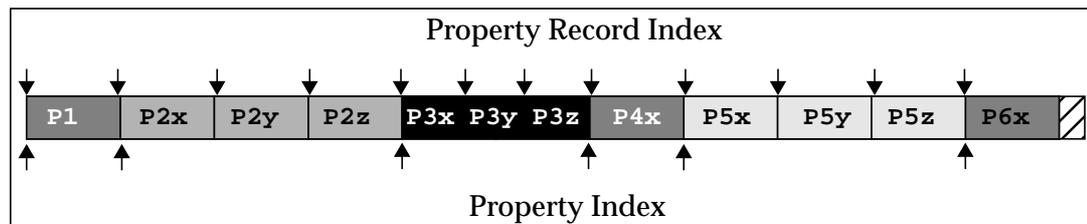


Figure 4-8 : Gestion des indices pour les propriétés

### c) fonctions d'édition

Les lignes polygonales ont, entre autres, un certain nombre de fonctions d'édition de la *géométrie* ou de la *topologie*. Ces fonctionnalités permettent à l'utilisateur de modifier légèrement la courbe en fonction de nouvelles données par exemple.

Parmi ces fonctionnalités, celles qui seront le plus utilisées par la suite sont:

- l'**extension** au niveau d'une extrémité ("*extend Node*")
- la **densification** ("*densify*") : cette fonction permet de fixer la distance maximum acceptable entre deux atomes voisins
- l'**ajout d'un nœud** à une position précise ("*break segment*").

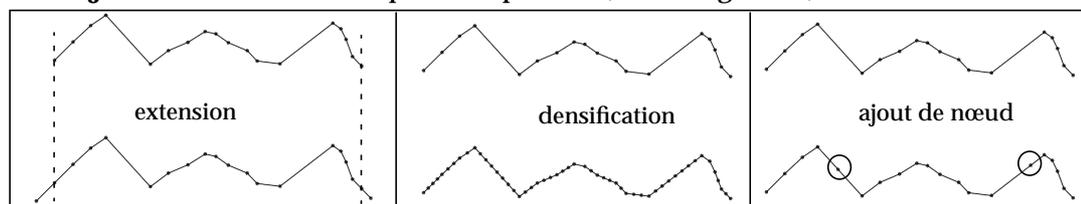


Figure 4-9 : Trois fonctionnalités beaucoup utilisées par l'objet faille

## 2.1.2 Héritage des fonctions d'édition des lignes polygonales

### a) Notion d'héritage

Comme cela a été décrit dans le premier chapitre, l'un des avantages de la programmation orientée objet est la notion d'héritage. Cette notion permet de définir une nouvelle structure à partir d'un objet existant. La structure ainsi créée profite de toutes les définitions (variables et fonctions) définies sur la *classe mère*. Sur la nouvelle classe il est en outre possible de définir de nouveaux champs de types "données" ou "fonctions" qui lui sont spécifiques.

Il est même possible de faire de l'*héritage multiple*. Une classe peut dériver de deux classes différentes, héritant ainsi des fonctionnalités des deux classes mères à la fois. Cette possibilité très forte a été utilisée à plusieurs reprises pour l'objet Faille.

## b) Exemples d'héritages utilisés pour l'objet faille

Les extraits de code ci-dessous montrent en quoi l'héritage permet de ne redéfinir que ce qui est spécifique à la nouvelle structure, tout en pouvant utiliser les fonctionnalités de la classe dont elle hérite.

### *Exemple de code 4.1. Classes PLine, ILine, GFaultBundle et GFault*

```

1  //- PLine
2  class PLine {
3
4      public :
5          // constructeur / destructeur
6          PLine() ;
7          ~PLine() ;
8
9              [...]
9          // Retourne les extrémités
10         bool extremities(PtrList<Node> &list) const ;
11             [...]
12     }
13     -----
14     //- ILine
15     class ILine {
16         public:
17             // constructeur / destructeur
18             ILine() ;
19             ~ILine() ;
20
21                 [...]
21             // Ajout d'un noeud (atom) sur un segment (seg).
22             void break_simplex(SegUse* seg, Atom *atom, SegUse *&nsu) ;
23             // Densification
24             void densify(double max_length);
25             // Extension a une extrémité
26             SegUse* extend_atom(Atom* atom, Atom*& new_atom) ;
27                 [...]
28     }
29     -----
30     //- GFaultBundle
31     class GFaultBundle: public PLine {
32         [...]
33     }
34     -----
35     //- GFault
36     class GFault: public ILine {
37         [...]
38     }

```

L'objet Faille peut ainsi appeler la fonction qui retourne la liste des nœuds situés aux deux **extrémités** de la courbe principale [exemple de code 4.1: ligne 10]. Il en est de même pour les fonctions d'édition qui sont déclarées dans la classe ILINE pour la plupart (*lignes 21 à 26*). De nombreuses autres fonctions que celles montrées dans la [figure 4-9] sont disponibles, et utilisées par l'objet Faille.

La définition de la classe dérivée, GFAULTBUNDLE, ne contient que ce qui est spécifique à l'objet Faille. Parmi ces fonctions, on trouve par exemple la fonction qui initialise le tableau de propriétés. Cette fonction crée trois nouvelles propriétés qui seront stockées sur chacun des atomes. Ces propriétés seront décrites plus loin, lors de la description du stockage des génératrices.

Cette classe définit aussi les différentes manières de créer un objet Faille. Actuellement, il en existe trois [figure 4-10]:

- la méthode la plus utilisée actuellement est la création à partir d'un **ensemble de courbes**. Ces courbes peuvent être soit sub-horizontales (lignes médianes entre les bords des différents horizons) ou sub-verticales (interprétations de coupes). Dans ce cas, un nœud correspondant à l'abscisse curviligne 0.5 est créé pour chacune des courbes. En reliant ces nœuds, on obtient la courbe principale de la faille. Les courbes fournies en données constituent les *génératrices*.
- Si l'on dispose déjà d'une **surface triangulée** correspondant à la géométrie souhaitée pour la faille, une fonction appelée paramétrisation [78 - Eck], [79 - Floater], [82 - Levy 1] permet d'extraire un ensemble de courbes parallèles le long de cette surface. Il ne reste plus ensuite qu'à utiliser la fonction précédente pour créer la faille à partir de ces courbes
- Enfin, il est possible de **créer la faille à partir de sa courbe principale** et d'**assigner** à chacun des nœuds sa **courbe secondaire** par la suite. Actuellement, cette manière de procéder n'est pas utilisée, mais elle pourrait éventuellement servir à vérifier rapidement en 3D l'interprétation des sections sismiques en domaine temps.

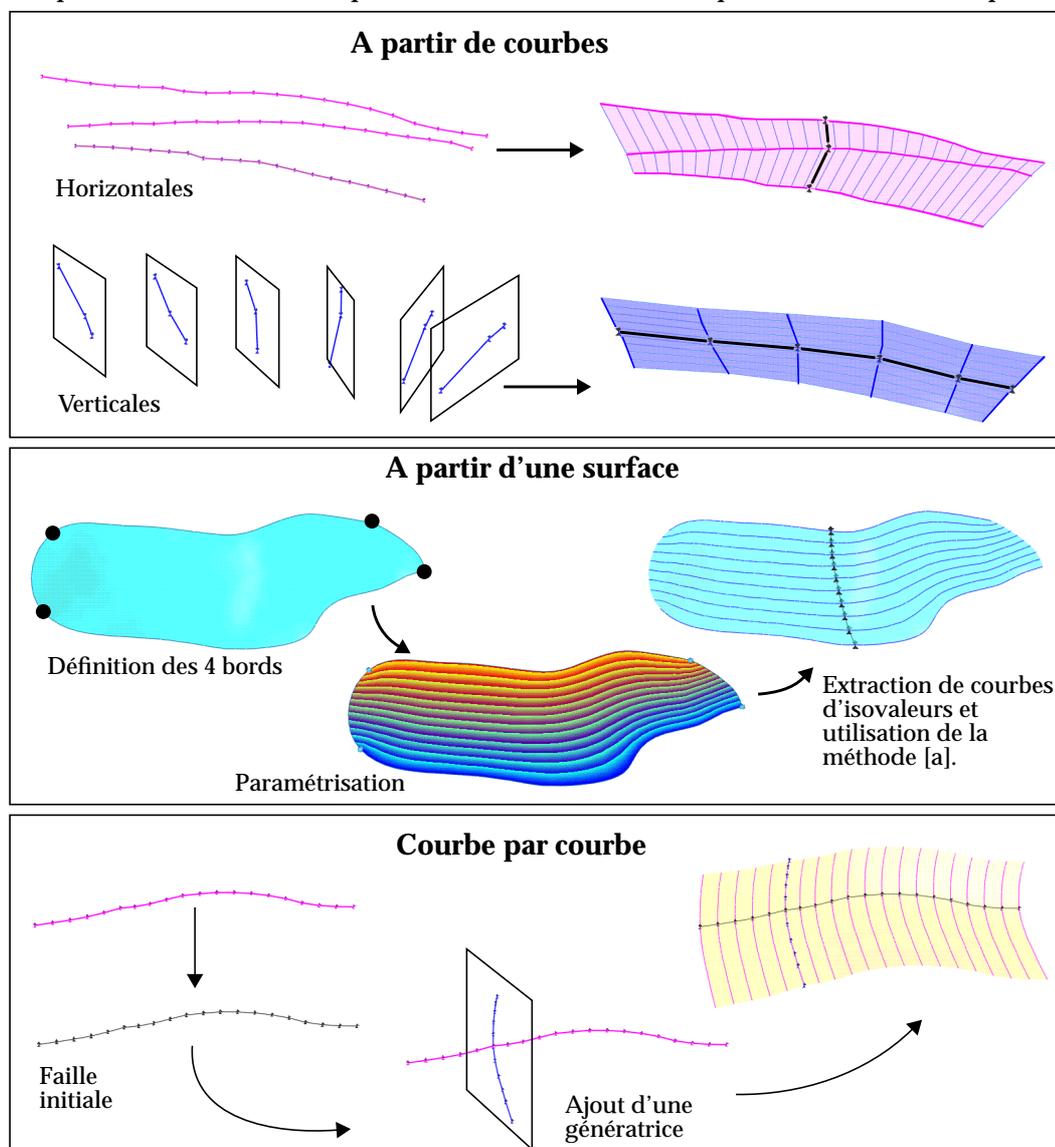


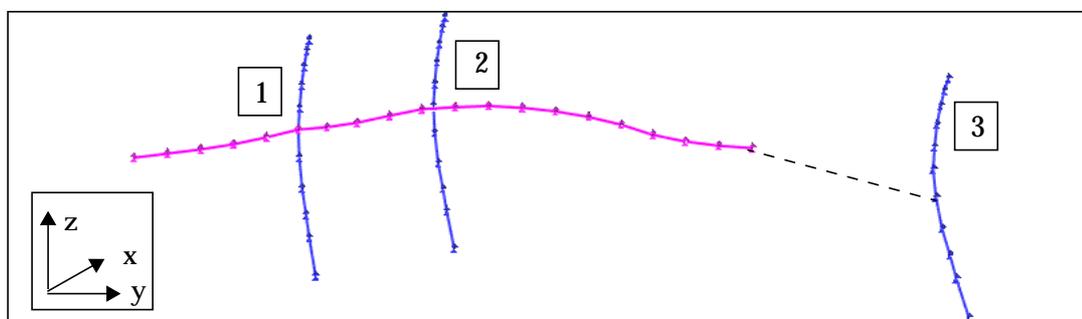
Figure 4-10 : Les trois façons de créer une faille (à partir de courbes, de surfaces, ou étape par étape)

### c) Exemple d'algorithme utilisant l'héritage: ajout d'une génératrice

La dernière méthode utilisée pour créer les failles est un bon exemple de l'utilisation de l'héritage. En effet, l'algorithme qui insère une nouvelle ligne secondaire (génératrice) sur la courbe principale se fait en deux temps.

Le premier temps consiste à déterminer quelle est la configuration. Le nœud correspondant à la nouvelle génératrice se trouve-t-il: près d'un atome existant, sur un segment existant ou en dehors de la courbe principale? [figure 4-11]. En fonction du cas rencontré, la fonction d'édition de la classe parente (ILINE) est appelée:

- si le nœud à créer est très proche d'un nœud existant, on ne crée pas de nouveau nœud, et on affecte juste la nouvelle génératrice au nœud existant.
- si le nœud à créer se situe sur la courbe principale, mais assez loin des extrémités du segment concerné, la fonction "*break\_simplex*" (exemple de code 4.1 ligne 22) est appelée.
- si le nœud à créer se situe à l'extérieur de la courbe principale, la fonction "*extend\_atom*" (exemple de code 4.1 ligne 26) est appelée.

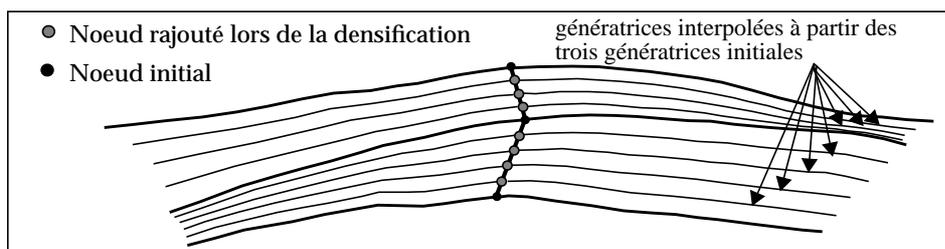


**Figure 4-11 :** La seule question à se poser pour ajouter une génératrice: la courbe tombe-t-elle près d'un nœud [1], à l'intérieur d'un segment existant [2] ou en dehors de la courbe principale [3]?

La deuxième partie de l'algorithme est la même quelle que soit la configuration précédente. Elle consiste à construire une nouvelle génératrice et à l'affecter au nœud qui vient d'être créé durant la première étape.

Remarque: si aucune courbe n'est fournie pour un nœud donné (par exemple lorsque l'on ajoute la première génératrice dans la troisième méthode de construction, un seul atome se voit affecter une courbe "donnée"), une interpolation linéaire est calculée pour tous les autres atomes, en attendant qu'une courbe plus appropriée leur soit éventuellement affectée plus tard.

De la même façon, lorsque l'on veut augmenter le nombre de nœuds sur la courbe principale, afin de raffiner la géométrie de la faille par exemple, la fonction de densification des lignes polygonales est utilisée (exemple de code 4.1 ligne 24). Le même mécanisme de mise à jour des génératrices est alors appelé automatiquement afin de créer les génératrices manquantes.



**Figure 4-12 :** Mise à jour automatique lors d'une densification de la courbe principale.

## 2.2 Les génératrices

Afin d'avoir une paramétrisation complète et de stocker des informations sur la géométrie de la faille dans la deuxième direction principale de la faille, chaque atome de la courbe principale porte une ligne secondaire polygonale appelée génératrice. Contrairement à certaines paramétrisations semblables utilisées par d'autres équipes [84 - Mostad], les génératrices utilisées dans gOcad sont aussi des **courbes polygonales** (et non des segments de droite). Ceci donne une *plus grande flexibilité* à cet objet, très utile lors des simulations, mais aussi beaucoup plus de travail pour leur gestion.

Comme cela a été souligné dans le paragraphe précédent, du fait de la dérivation de l'objet faille à partir des lignes polygonales, les atomes de la courbe principale ne peuvent avoir au plus que deux voisins. Les points de cette génératrice ne pouvant être stockés sous forme d'atomes, ils sont stockés sous formes de propriétés.

### 2.2.1 Stockage sous forme de propriétés

Un atome de la courbe principale stocke donc, en plus de ses coordonnées géométriques (x,y,z) habituelles [A], un certain nombre d'informations supplémentaires [figure 4-13]:

- [B]: l'**abscisse curviligne** de l'atome sur la génératrice. Par défaut, cette propriété est égale à 0.5, mais elle peut être modifiée par la suite.
- [C]: les coordonnées de la **direction principale** de la génératrice et d'un vecteur normal à la fois à ce dernier et à la tangente au backbone.
- [D]: une liste de n propriétés vectorielles à trois champs, n étant le nombre de points par génératrice. Chacune de ces propriétés correspond au stockage des coordonnées d'un des point de la génératrice accrochée sur l'atome en question.

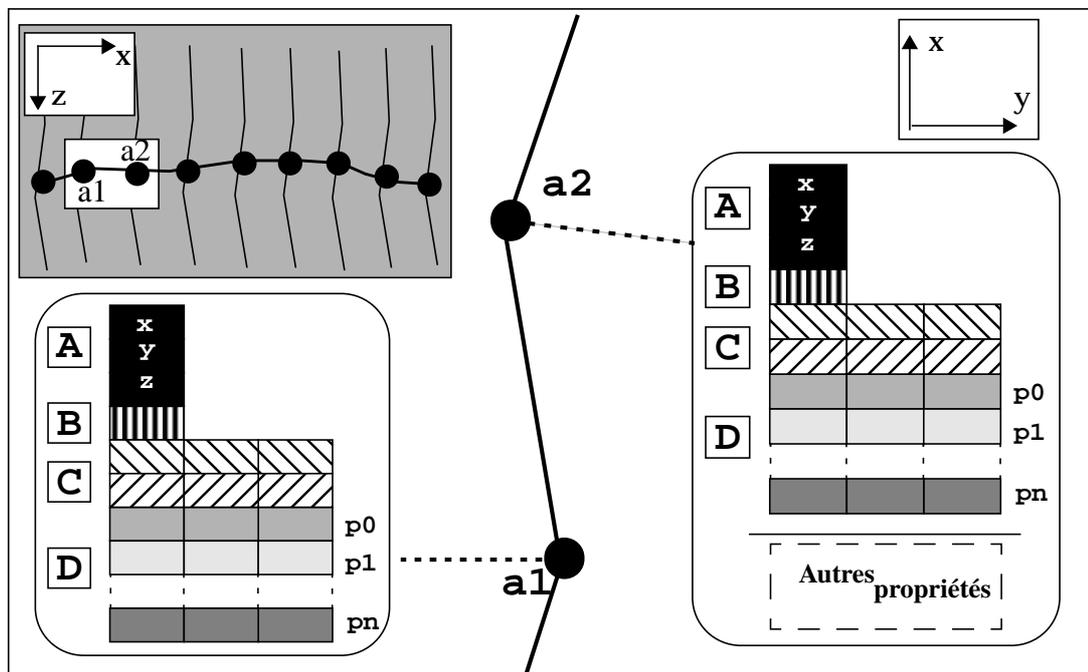


Figure 4-13 : Stockage des génératrices sous forme de propriétés

Par définition, toutes les génératrices d'une faille comportent le **même nombre de points**. Ainsi la gestion des propriétés est aisée, puisque chaque atome de la courbe principale stocke exactement le même nombre de propriétés. Un système d'indices permet de retrouver rapidement les coordonnées du  $k^{\text{ième}}$  point sur le  $j^{\text{ième}}$  atome.

Cependant, ce nombre de points peut être modifié au cours de l'édition, afin d'affiner la géométrie si nécessaire.

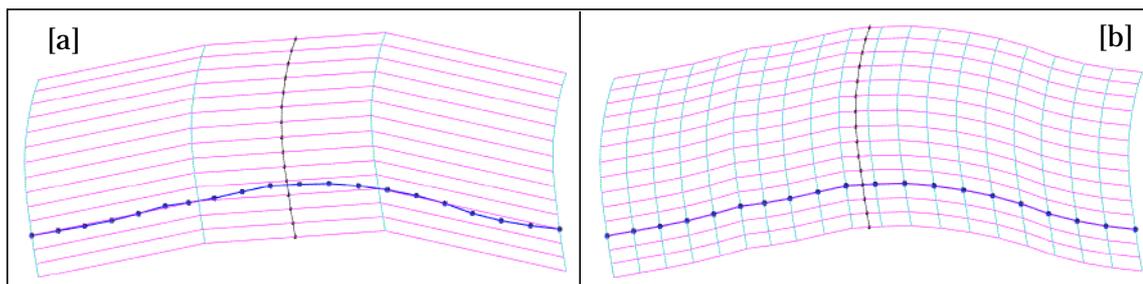


Figure 4-14 : Densifier les génératrices permet d'améliorer la précision de la géométrie. [a]: 4 points par génératrice; [b]: 20 points par génératrice.

Afin que toutes les génératrices contiennent exactement le même nombre de points (nombre choisi par l'utilisateur), les courbes fournies en données sont rééchantillonnées avec un pas régulier. C'est pourquoi, si les courbes contiennent beaucoup de détails, il est important de donner un grand nombre de points afin de les respecter au mieux.

### 2.2.2 Stockage dans une base locale

Les coordonnées des points ainsi rééchantillonnés sont ensuite transformées afin d'être stockées dans une nouvelle base [75 - Cognot 2]. Les paragraphes suivants montrent les avantages et les inconvénients d'un tel stockage.

#### a) Définition de la base locale

Cette base locale, orthonormée, est centrée sur l'atome qui stocke la génératrice. Elle est composée de trois vecteurs orthogonaux, tenant compte de la tangente locale à la courbe principale (*tangent*) et de la direction principale de la génératrice (*main\_dir*). Comme ces deux vecteurs ne sont pas forcément orthogonaux, une fonction permet de calculer une base orthogonale la plus proche possible de ces deux vecteurs, en utilisant un double produit vectoriel (exemple de code 4..2) [figure 4-15].

#### Exemple de code 4..2. Calcul de la base locale

```

1 void GFaultGenerator::get_normal_vectors(
2     const Vector3d& tangent, Vector3d& v, Vector3d& w
3 ) {
4     w = tangent ^ main_dir ;
5     w = w.normalize() ;
6     v = w ^ tangent;
7 }

```

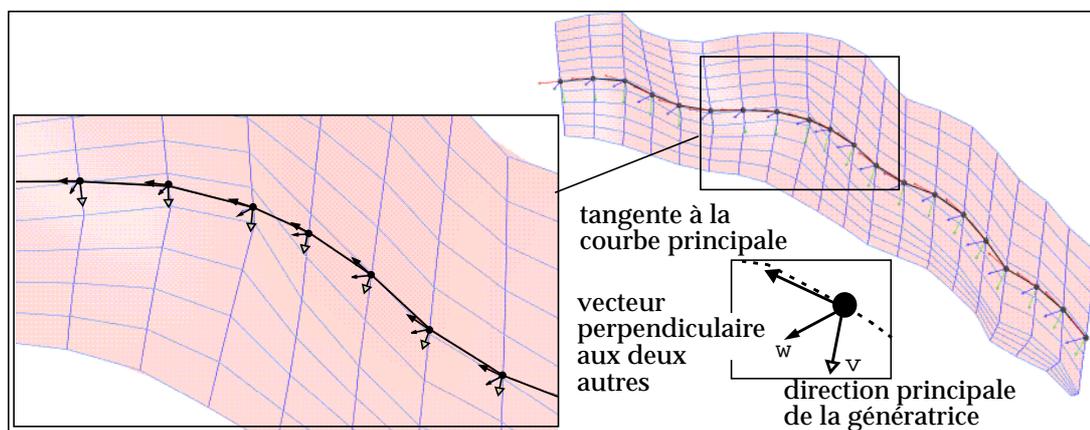
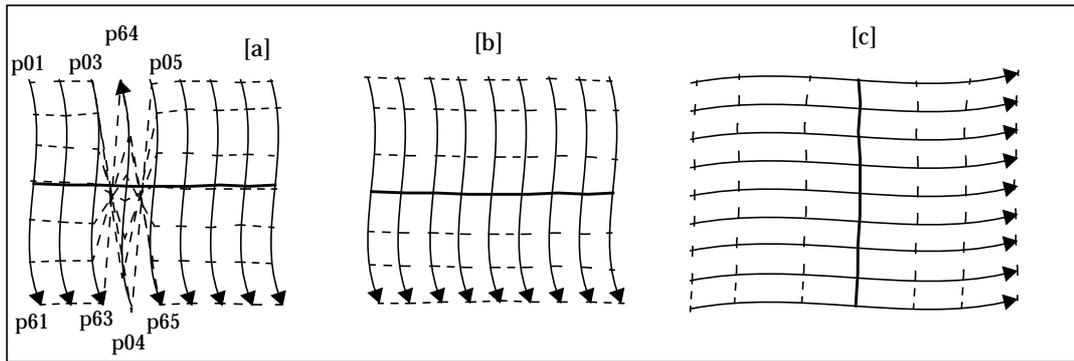


Figure 4-15 : calcul de la base locale pour chaque atome de la courbe principale



**Figure 4-16 :** Il faut orienter les génératrices de manière cohérente pour éviter les croisements. [a]: une génératrice mal orientée crée des problèmes lors de l'interpolation; [b] et [c]: les génératrices sont bien orientées (courbe principale sub-horizontale en b et sub-v verticale en c).

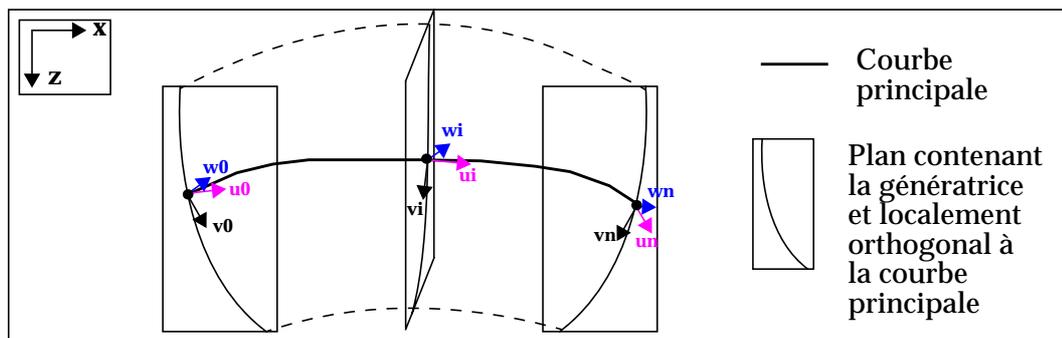
Les différentes bases locales sont toutes orientées de la même manière le long de la courbe principale. Ainsi, le point  $p_0$  d'une génératrice peut être relié aux points  $p_0$  des génératrices voisines sans problème de croisement. Selon que la courbe principale est sub-v verticale ou sub-h horizontale, un critère différent est utilisé pour orienter les bases locales:

- backbone sub-horizontale [b]: les génératrices sont orientées du haut vers le bas (coordonnée  $z$  du vecteur  $v$  négative)
- backbone sub-vertical [c]: les génératrices sont toutes orientées dans la même direction. La direction de référence est celle de la première génératrice ajoutée sur la courbe principale.

## b) Intérêt de la "base locale"

Si l'on construit une faille à partir d'une courbe principale semi-circulaire à laquelle on attache une génératrice contenue dans un *plan vertical et localement perpendiculaire* à la courbe principale, le résultat que l'on veut obtenir est un ensemble de génératrices planes elles aussi, et contenues dans des plans verticaux et localement perpendiculaires à la courbe principale [figure 4-17]. Une des manières faciles pour obtenir ce résultat est de stocker les coordonnées des points des génératrices dans une *base locale* dont deux vecteurs sont contenus dans le plan en question.

De la même manière, lorsque l'on étend une faille à l'une de ses extrémités, il suffit de construire le nouveau nœud et de recopier directement les coordonnées  $(u, v, w)$  des points de la génératrice précédente. Une utilisation similaire de cette base locale pour stocker et interpoler les vecteurs d'incertitude montrera de manière plus évidente l'intérêt de stocker toutes ces variables dans un repère local.



**Figure 4-17 :** Stockage des coordonnées dans une base locale

### c) La classe Transformer3D

Cette classe permet de définir des transformations dans l'espace. Ces transformations peuvent être par exemple des translations, des rotations ou des changements de base. Pour stocker les points des génératrices, nous avons utilisé les fonctions correspondant à la dernière transformation.

Deux fonctions permettent de passer des coordonnées de l'espace réel  $\mathfrak{R}(x,y,z)$  à la base locale  $\mathfrak{S}(u,v,w)$ . L'exemple de code 4.3 montre comment passer de l'espace réel à la base locale connaissant l'origine de la nouvelle base  $R(x_a, y_a, z_a)$  et les coordonnées dans l'espace réel des trois vecteurs définissant cette base.

*Exemple de code 4.3. Changement **Espace réel / base locale**:  $\mathfrak{R} \Rightarrow \mathfrak{S}$*

```

1  const Transformer3d& GFaultGenerator::xyz2uvw() {
2      // Returns the transformer from world coordinates to
3      // GFaultGenerator coordinates.
4      Vector3d u = fault_-> get_local_tangent(atom_) ;
5      Vector3d v, w ;
6      get_normal_vectors( u, v, w ) ;
7      Vertex* origin = atom_->v() ;
8      xyz2uvw_.set_new_basis( origin, v, w, u ) ;
9      return xyz2uvw_ ;
10 }
11
12 void GFaultGenerator::set_uvw_from_xyz_points() {
13     const Transformer3d& transform = xyz2uvw() ;
14     int fnb_points = fault_->nb_points() ;
15     const int *indices = fault_->bundle()->points_indices() ;
16     for( int i=0 ; i<fnb_points ; i++ ) {
17         Point3d p = transform * xyz_points_[i] ;
18         int index = indices[i];
19         atom_->value(index) = p.x() ;
20         atom_->value(index+1) = p.y() ;
21         atom_->value(index+2) = p.z() ;
22     }
23 }
24
```

De la même façon, l'exemple de code 4.4 montre la transformation inverse (la fonction qui applique cette opération aux points successifs n'est pas détaillée).

*Exemple de code 4.4. Changement **Base locale / espace réel**:  $\mathfrak{S} \Rightarrow \mathfrak{R}$*

```

1  void GFaultGenerator::get_xyz_points(Point3d*& xyz_points) {
2      const Transformer3d& transform = uvw2xyz() ;
3      [...]
4  }
5
6  const Transformer3d& GFaultGenerator::uvw2xyz() {
7      // Returns the transformer from GFaultGenerator
8      // coordinates to world coordinates.
9      Vector3d u = fault_-> get_local_tangent(atom_) ;
10     Vector3d v, w ;
11     get_normal_vectors( u, v, w ) ;
12     Vertex* origin = atom_->v() ;
13     uvw2xyz_.set_old_basis( origin, v, w, u ) ;
14     return uvw2xyz_ ;
15 }
```

#### d) Inconvénients des changements de repères

Comme on peut le voir sur les exemples de codes précédents, ces changements de repères sont assez lourds à utiliser. Même si les TRANSFORMER3D sont stockés dans chacune des génératrices, une fois qu'elles sont créées, les génératrices ne sont construites que ponctuellement à chaque fois que l'on a besoin de les modifier ou que l'on utilise l'une d'entre elles pour une opération géométrique. Il faut donc à chaque opération sur les points d'une génératrice, repasser tous ces points en revue afin de leur appliquer la transformation adéquate.

Ces transformations sont appliquées relativement souvent lors de la phase d'édition des génératrices:

- **Espace réel / base locale:**  $\mathfrak{R} \Rightarrow \mathfrak{S}$ :  
lorsqu'un changement de géométrie a été effectué sur la faille, par exemple par application d'une contrainte suivie d'une interpolation par D.S.I.
- **Base locale / espace réel:**  $\mathfrak{S} \Rightarrow \mathfrak{R}$   
Cette opération est utilisée beaucoup plus souvent: pour construire la courbe correspondant aux bords de la faille, pour changer le nombre de points sur les génératrices, pour construire un autre type d'objet à partir d'un objet faille, etc.

On peut donc se demander si une autre structure ne serait pas plus adéquate pour paramétrer la faille.

#### e) Pourquoi ne pas utiliser une autre structure de données?

On aurait pu envisager de définir l'objet faille par une **grille irrégulière bidimensionnelle**, en autorisant les atomes à être reliés à plus de deux voisins. Ceci aurait permis en outre de pouvoir définir plusieurs courbes dans les deux directions principales de la faille, ce qui n'est pas possible actuellement (un seul backbone est autorisé). Cela aurait aussi permis de ne privilégier aucune des deux directions.

Cependant, la structure ainsi définie aurait été beaucoup *plus difficile à gérer*. L'objet actuel ne contient des atomes qu'au niveau de la courbe principale. Ceci permet d'avoir un objet très léger, tout en ayant rapidement accès aux coordonnées d'un point, en connaissant l'indice de l'atome qui le porte, ainsi que l'indice du point sur la génératrice.

Un atome contenant beaucoup plus d'informations que le stockage simple des coordonnées (u,v,w) des points, on aurait perdu en **flexibilité et en légèreté** avec une structure de type «grille irrégulière».

De plus, on n'aurait pas pu utiliser tous les **avantages de l'héritage** des fonctions des lignes polygonales. Il aurait fallu redéfinir toutes les fonctionnalités d'édition, telles que l'ajout d'une courbe, la densification de la grille, l'extension aux extrémités, etc.

Enfin, il faut bien garder à l'esprit la fonctionnalité principale de cet objet: **la génération rapide d'un grand nombre de failles** possibles autour d'une faille de référence.

La phase d'édition de la faille de référence ne se produit qu'au début du processus et reste relativement limitée. Comme nous l'avons montré, l'objet correspond bien aux types de données fournies pour modéliser les failles. Si les données utilisées pour construire la faille de référence sont cohérentes, il y a *peu d'opération d'édition* à utiliser. On peut même envisager des cas où cette phase d'édition est inexistante si la définition d'une faille correcte du point de vue géologique a été réalisée dans une phase préliminaire, et que l'on ne construit l'objet faille uniquement pour étudier les incertitudes sur la position et la forme de celle-ci [figure 4-10] (b).

C'est pourquoi la flexibilité et la légèreté de l'objet ont été privilégiées, et la structure actuelle conservée.

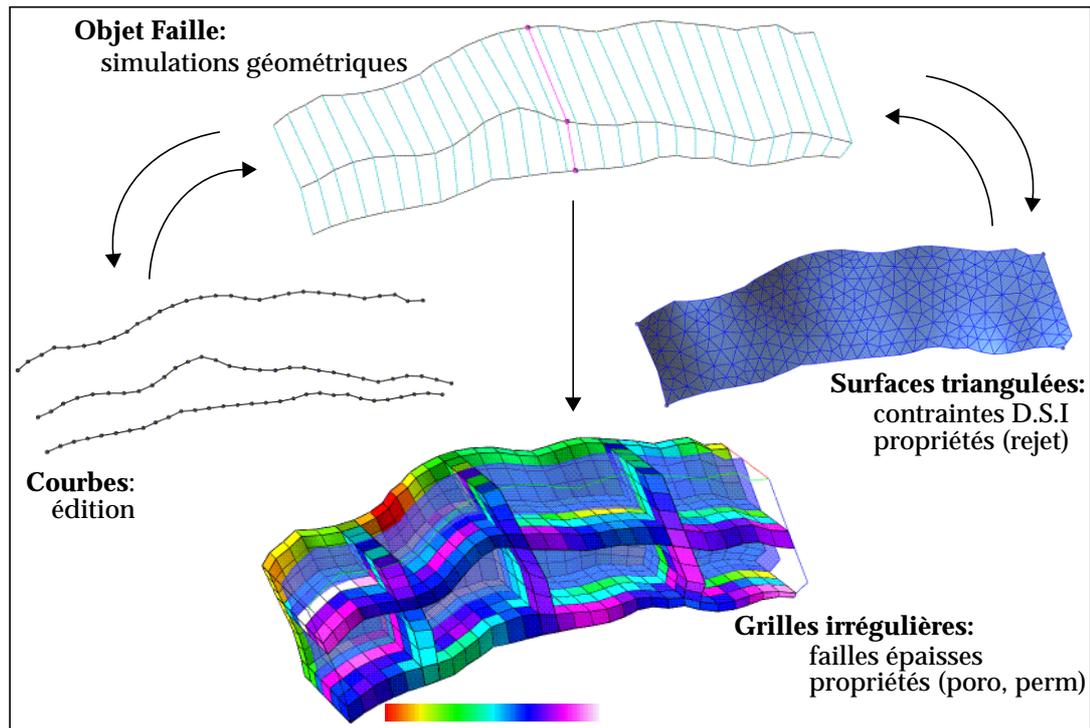
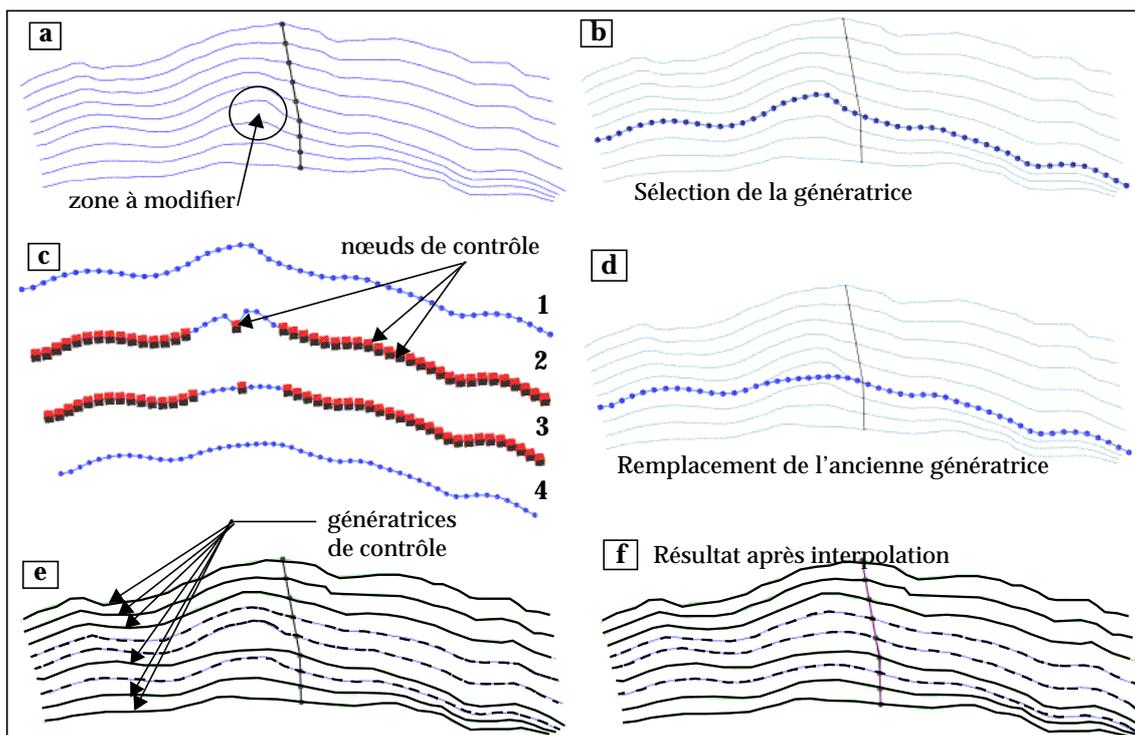


Figure 4-18 : Relations entre l'objet faille et les autres objets gOcad.

### 3. Des relations étroites avec les objets gOcad existants

Afin de garder la structure légère de l'objet faille, nous avons réduit au maximum ses fonctionnalités. Nous avons concentré nos efforts pour définir un objet simple et très facile à manipuler, mais une faille ne se limite pas à sa géométrie. C'est pourquoi, de nombreuses passerelles existent afin de transformer l'objet faille en d'autres objets qui ont leurs fonctionnalités propres. L'ensemble permet de définir la faille de manière plus complète [figure 4-18]:

- **les courbes polygonales:** *modification de la géométrie*  
L'objet faille est capable de gérer les opérations géométriques et topologiques effectuées sur la courbe principale. Cependant, dans la version actuelle, il faut passer par des lignes polygonales pour modifier la forme d'une génératrice. La courbe ainsi modifiée remplace ensuite l'ancienne génératrice (cf § 3.1, page 89).
- **les surfaces triangulées:** *définition de contraintes D.S.I. et visualisation de propriétés*  
La définition de contraintes directement sur l'objet faille afin de prendre en compte de nouvelles courbes d'interprétations n'est pas possible du fait de l'asymétrie de la paramétrisation (ceci sera montré plus loin). C'est pourquoi nous utilisons les capacités des surfaces triangulées pour définir des contraintes géométriques sur l'objet faille. Pour les mêmes raisons, nous n'avons pas défini la gestion des propriétés comme le rejet sur l'objet faille. Afin de le garder le plus léger possible, nous préférons utiliser les fonctionnalités définies sur les surfaces triangulées (cf § 3.2, page 90).
- **les grilles irrégulières:** *failles épaisses et simulations de propriétés*  
De plus en plus, les industriels demandent à ce que les failles ne soient plus modélisées uniquement comme des surfaces, mais que leur épaisseur et leurs propriétés internes soient aussi modélisées. Bien que nous n'ayons pas poussé dans cette direction, il est possible de définir un volume autour de la faille puis de simuler les propriétés de celle-ci. On peut même envisager d'étudier l'incertitude sur l'épaisseur de la faille. Cet aspect sera repris dans la conclusion de ce mémoire.



**Figure 4-19 : Édition de la géométrie d'une génératrice. [a]: faille initiale; [b] transformation en ligne polygonale; [c]: modification de la ligne; [d]: réinjection de la nouvelle courbe dans la faille initiale; [e]: certaines génératrices sont fixées (*Control Generators*); [f]: résultat après interpolation.**

### 3.1 Les lignes: édition de la géométrie

Lorsque l'on n'a aucune information sur la faille, et que l'on n'a que les *ensembles de points migrés* (ce qui est le cas pour la plupart des jeux de données avec lesquels nous avons eu à travailler), on construit la faille à partir des bords de ces horizons. Ces bords étant assez imprécis et inégaux en qualité, il peut arriver que les courbes construites à partir de ces bords doivent être modifiées par la suite, afin d'obtenir une faille mécaniquement correcte [figure 4-19].

Dans ce cas, il suffit de transformer les génératrices qui doivent être modifiées en **lignes polygonales** ([figure 4-19] [b]) et de modifier la géométrie de celles-ci en utilisant des fonctions d'édition simples [figure 4-19] [c]).

Dans l'exemple traité ci-dessus, il a suffit de fixer la majorité des nœuds (**control nodes** marqués par un cube), de modifier la position de l'un d'eux et de le fixer lui aussi en nœud de contrôle, puis de laisser libres les nœuds autour de ce dernier ([figure 4-19] [c] 2). Il ne reste plus ensuite, qu'à **interpoler** cette ligne polygonale à l'aide d'un interpolateur (D.S.I. par exemple) ([figure 4-19] [c] 3), pour obtenir une courbe lisse au niveau de la modification. Une fois la forme correcte obtenue, ces courbes peuvent remplacer les anciennes génératrices dans l'objet faille ([figure 4-19] [d]).

Comme ce changement peut modifier sensiblement la géométrie de la faille initiale, il est en outre possible de fixer un certain nombre de génératrices que l'on considère comme correctes (**Control Generators**), et d'interpoler les génératrices incorrectes à partir des **génératrices de contrôle** ([figure 4-19] [e]). Comme les points sont stockés sous forme de propriétés vectorielles et que l'interpolateur D.S.I. interpole de la même manière les propriétés que les coordonnées  $(x,y,z)$  des atomes de la courbe principale, cette opération est obtenue une fois de plus automatiquement par l'héritage des fonctions des lignes polygonales. Ceci permet d'obtenir une faille qui respecte la nouvelle géométrie, tout en restant relativement lisse ([figure 4-19] [f]).

### 3.2 Les surfaces triangulées: utilisation des contraintes

Comme cela a été montré, des **opérations simples** peuvent être appliquées sur la géométrie de l'objet faille, soit directement sur la courbe principale, soit sur les génératrices, en passant par des lignes polygonales.

Cependant, ces modifications ne suffisent pas toujours à prendre en compte toutes les informations disponibles. Lorsque l'on doit tenir compte d'interprétations de failles qui traversent la faille en biais, on ne peut les intégrer ni comme courbe principale, ni comme génératrice ([figure 4-3]: "Autres courbes interprétées").

Pour tenir compte de ces données, il faut mettre en place un système de contraintes sur l'objet faille, qui l'oblige à suivre la forme de ces courbes sans les intégrer directement dans la paramétrisation. Une telle contrainte est appelée "fuzzy control points" dans gOcad.

#### 3.2.1 Mécanisme général

##### a) Principe des associations

Une association permet de stocker des informations liées à un objet, mais uniquement au moment où ces informations sont utiles. L'objet peut très bien exister par lui-même sans que l'association n'ait été créée. Un objet peut être lié à plusieurs associations, mais une association n'observe qu'un seul objet à la fois. Lorsque la modification de l'objet le nécessite, les informations utiles à l'association sont stockées. C'est un moyen pratique de rajouter ponctuellement de l'information concernant un objet, sans avoir à modifier la classe de l'objet [4 - Cognot 1]. Ce mécanisme, largement utilisé pour la gestion des contraintes géométriques dans gOcad, est fortement inspiré d'un mécanisme de programmation classique: le principe "Modèle - Vue - Contrôleur" [80 - Krasner].

Lorsqu'une modification est appliquée à l'objet, ce mécanisme permet de notifier les associations que l'objet a été modifié et qu'elles doivent se mettre à jour ("update"). Dès ce moment, l'association est marquée comme n'étant pas à jour ou "dirty". Le mécanisme de mise à jour est alors appelé (do\_update). Cette étape peut par exemple correspondre à la destruction des informations erronées, voire de toute la structure de stockage de l'association.

Ce n'est qu'à la prochaine modification faisant intervenir cette association que l'information manquante est calculée et stockée (étape 2). Ainsi, un minimum de choses est stocké entre deux opérations faisant appel à cette association [figure 4-20].

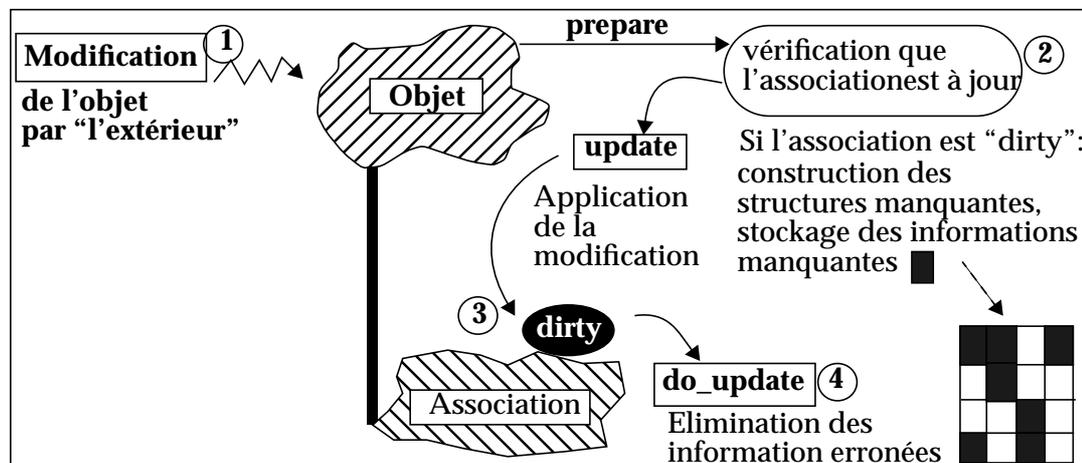


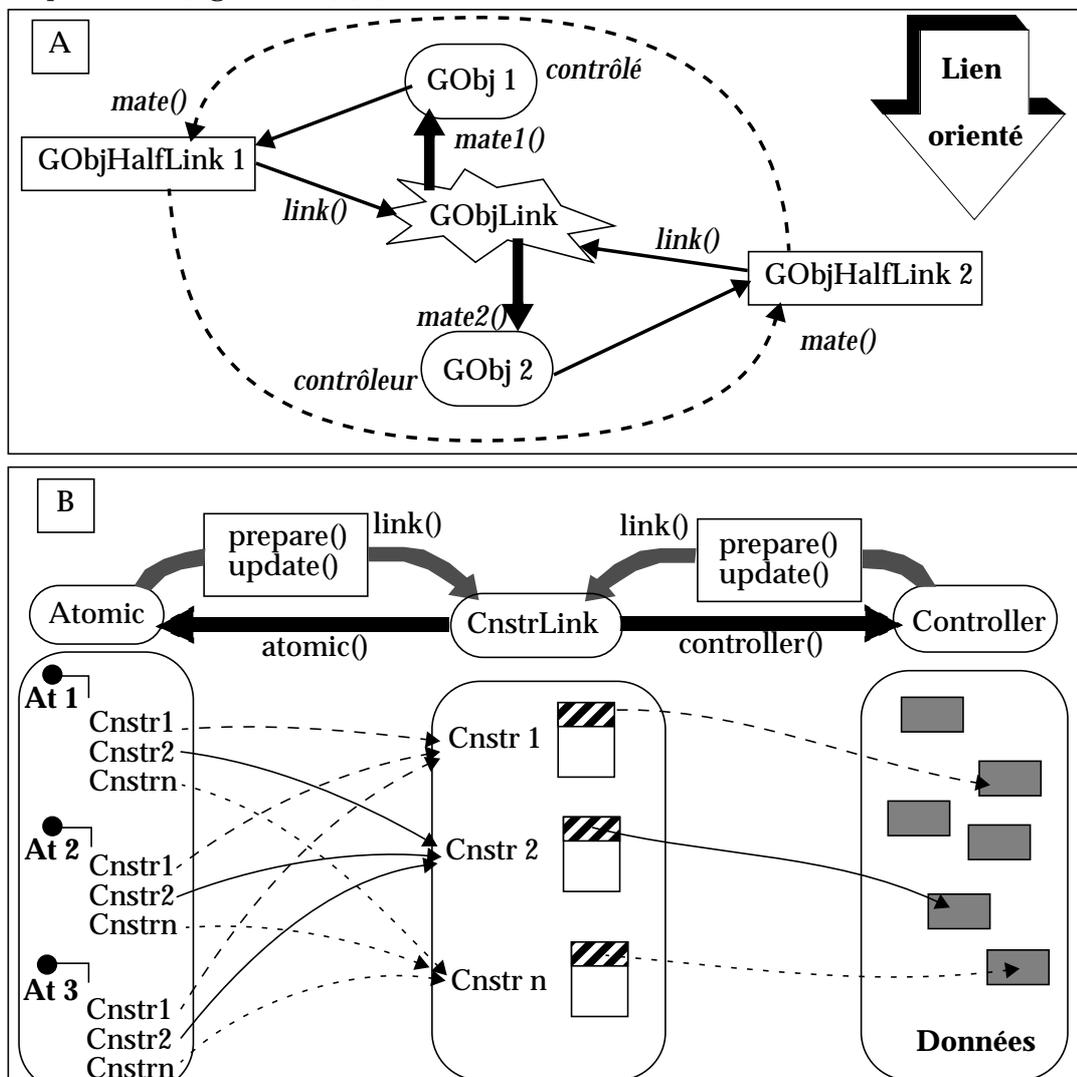
Figure 4-20 : Principe de l'Association (modifié d'après [4 - Cognot 1]).

## b) Le mécanisme des contraintes

Une contrainte géométrique dans gOcad met souvent en relation deux objets géologiques : un objet (*contrôleur*) possède l'information qui va permettre de modifier le second (*contrôlé*). Une contrainte peut faire agir un objet (GObj = Geological Object) sur lui-même, par exemple lorsque les valeurs d'une propriété sont connues en certains points, et que l'on veut calculer une valeur interpolée pour les autres points du même objet. Dans ce cas le contrôleur et le contrôlé sont représentés par le même objet.

Dans le cas général, la contrainte est représentée par une paire d'associations particulières, chacune d'entre elles attachée à l'un des deux objets. La gestion des informations communes se fait par l'intermédiaire d'un lien (GOBJLINK). Ainsi, toute action sur l'un de ces objets peut avoir une influence sur chacune des deux entités. Comme les deux objets ont un rôle différent, le lien est orienté. Lors de sa construction, l'ordre dans lequel les deux objets sont fournis définit quel est le contrôleur et quel est le contrôlé [figure 4-28] [A].

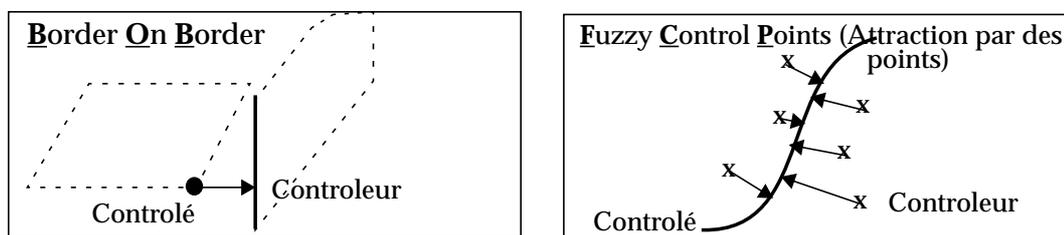
De ces relations binaires entre deux objets ont été dérivées les contraintes (CNSTR-LINK). A chaque fois qu'une nouvelle contrainte est ajoutée sur un objet, un lien de type CNSTRLINK est construit et l'information correspondante stockée au niveau de chaque atome [figure 4-28] [B].



L'information stockée dans l'atome est différente en fonction de la contrainte à laquelle elle fait référence et nous ne rentrerons pas plus dans les détails. Cette information est utilisée par l'interpolateur D.S.I. afin de modifier les valeurs (x,y,z) ou les valeurs de propriétés de l'atome en fonction des coefficients stockés (pour plus de précisions, on pourra se reporter à [4 - Cognot 1], [6 - Mallet 1]). La modification peut éventuellement ne porter que sur certains atomes (atomes du bord des horizons par exemple)

La plupart des contraintes géométriques font intervenir des notions de distance entre le contrôleur et le contrôlé. Elles utilisent des fonctions de projection. Toutes les contraintes utilisant des projections sont des classes dérivées de la classe PRJCNSTR-LINK, elle même dérivée de la classe CNSTRLINK. En fonction du type d'élément sur lequel les atomes du contrôleur vont chercher l'information déterminant les atomes de l'objet contrôlé qu'ils vont influencer, on distingue les SEGPRJCNSTR (la cible est un segment) et les TRGLPRJCNSTR (la cible est un triangle).

#### SegPrjCnstr



#### TrglPrjCnstr

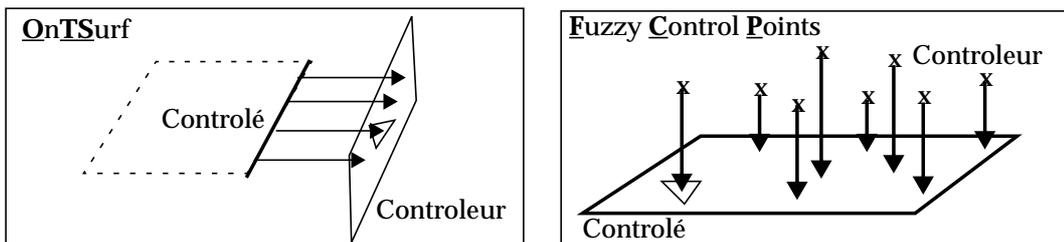


Figure 4-22 : Quelques modèles Atomique-Contrôleur dans gOcad.

Comme on a pu le voir précédemment, le mécanisme est très efficace mais lourd à mettre en place. D'autant plus, que seules les classes de bases sont décrites dans le paragraphe ci-dessus, et que de très nombreuses classes sont à définir pour la mise en place de la gestion des contraintes sur un objet (élément cible, classe qui permet de tirer de l'un des objets sur l'autre....).

Pour gérer les contraintes attachées à l'objet faille, deux options s'offraient à nous:

- **Redéfinir les structures** nécessaires pour les contraintes liées à l'objet faille à partir des structures des surfaces triangulées, très voisines
- Stocker une surface triangulée "cachée" dans l'objet faille et par un mécanisme de double association, **appliquer les contraintes sur la surface cachée**, puis transmettre les modifications géométriques à l'objet faille.

Dans un premier temps, il nous a paru plus facile et donc plus rapide de s'inspirer des structures existantes et de les adapter à l'objet faille. Mais après quelques semaines d'efforts, il s'est avéré que les contraintes ne pouvaient fonctionner sur l'objet faille, et que la deuxième solution était mieux adaptée au problème.

Les travaux nécessaires aux deux solutions sont décrits dans les paragraphes suivants.

### 3.2.2 Application directe à l'objet Faille: cas des points attracteurs (contrainte FCP)

Notre première idée a été de définir les contraintes liées à l'objet faille, directement sur celui-ci, en adaptant les contraintes existant déjà sur les surfaces triangulées. Pour ce faire, il a fallu définir ou redéfinir un certain nombre de classes. Une sélection de certaines de ces classes est présentée par la suite, en s'appuyant sur le cas des FCP.

#### a) Opérations de tirs

Comme cela a été souligné plus haut, beaucoup de contraintes utilisent des projections pour savoir «quels atomes de l'objet contrôlé sont affectés par quels atomes du contrôleur». Pour chaque type de contrainte par projection, on utilise un “tireur” (SHOOTER) [figure 4-23]. Cette classe permet de tracer une ligne dans une direction donnée à partir de l'un des objets vers l'autre objet, et de récupérer les points d'impact, ainsi que certaines informations topologiques. Ces informations sont regroupées dans un élément cible (triangle, segment...). L'objet faille n'étant constitué ni de triangles, ni de segments, un élément cible a été redéfini: la GFAULTCELL.

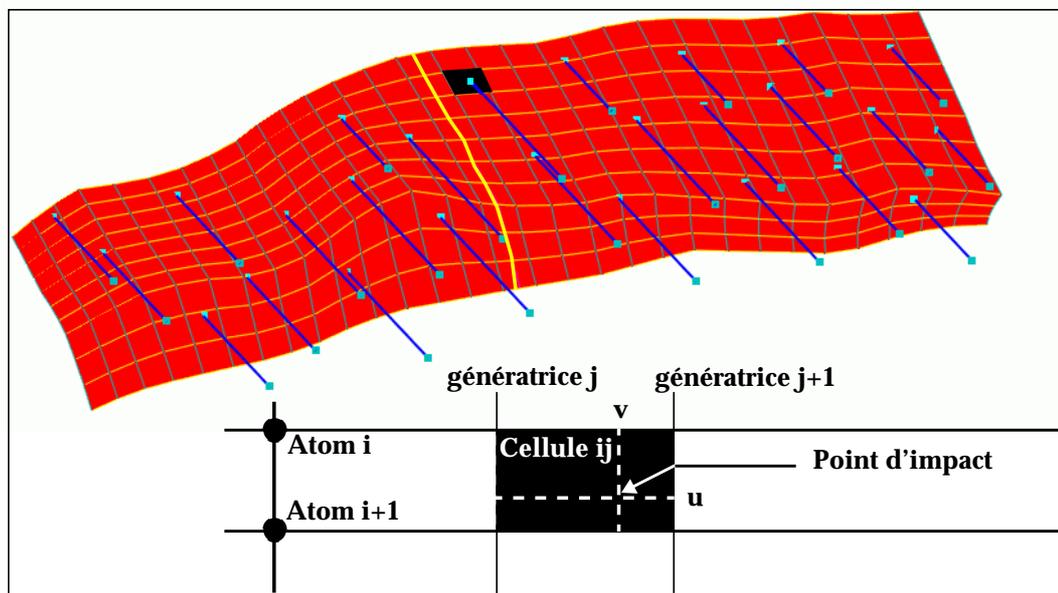


Figure 4-23 : Principe du tir: trouver le point d'impact le plus proche dans une direction donnée et stocker l'élément cible correspondant.

#### b) Élément cible: la GFaultCell

Une GFAULTCELL constitue la plus petite partie géométrique de l'objet faille. Toutes les GFAULTCELLS sont construites dans l'ordre et stockées dans un conteneur, le GFaultCellsContainer. Celui-ci permet de retrouver facilement les huit cellules voisines si elles existent. La numérotation d'une cellule suit la numérotation du premier point qui la constitue [figure 4-24]. Ainsi si ce point est le  $v^{\text{ième}}$  point de l'atome  $u$ :

$$\text{cell\_index}(u,v) = u * nv + v \text{ (avec } nv: \text{ nombre de points par génératrice)}$$

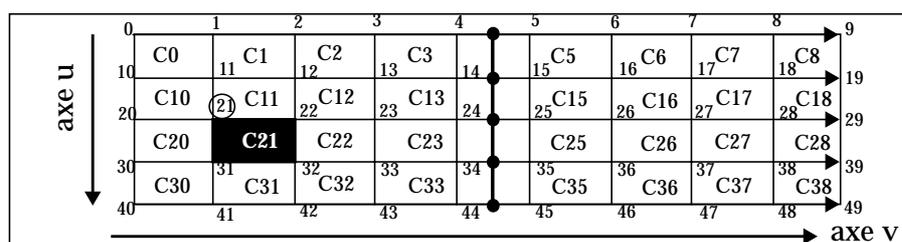


Figure 4-24 : Numérotation des cellules dans le conteneur

Une cellule contient: le segment de la courbe principale auquel elle est attachée, l'indice de la cellule correspondant au conteneur de cellules, les quatre points la définissant ( $p_0, p_1, p_2, p_3$ ), ainsi que l'indice  $v$  du point  $p_0$  dans le tableau de points de la génératrice qui le porte.

### c) Construction de l'arbre octant (*Octree*)

Afin d'accélérer la recherche du point le plus proche, nous avons utilisé la technique de l'OCTREE [83 -Meagher]. Cette structure de données de type "arbre" permet de partitionner de manière "intelligente" l'espace à trois dimensions, souvent représenté par la boîte entourant un objet donné. Ce découpage se fait de manière récursive. L'arbre résultant permet de garder l'historique des découpages et de gérer les relations géométriques entre les différentes cellules.

Cette construction est assez simple. Au départ, l'espace n'est représenté que par la boîte entourant la faille. Cette boîte étant intersectée par des GFAULTCELLS, elle est découpée en huit sous-boîtes (d'où le nom "*octree*"). Pour chacune des sous-boîtes, on recommence l'opération: si elle contient au moins une GFAULTCELL, elle est découpée, sinon, elle reste telle quelle. Le processus s'arrête soit lorsqu'un nombre de boîtes est atteint, soit lorsque le nombre de niveaux de découpages atteint un seuil donné [figure 4-25] (les schémas [a], [b] et [c] sont en deux dimensions pour plus de clarté).

Une fois de plus, nous avons utilisé les bénéfices de l'héritage: il a suffi de décrire l'élément de base correspondant à l'objet faille (GFAULTCELL), et de dériver une classe GFAULTOCTREE à partir de la classe OCTREE existante. Les fonctions de construction de l'OCTREE ont ainsi pu être utilisées directement, ce qui a beaucoup limité le travail.

Le GFAULTOCTREE est une association liée à l'objet faille. Ainsi, dès que la géométrie ou la topologie de la faille est modifiée, le GFAULTOCTREE est remis à jour par l'intermédiaire du GFAULTCELLSCONTAINER: toutes les GFAULTCELLS sont détruites, puis reconstruites à partir de la nouvelle topologie. Enfin un nouveau GFAULTOCTREE est reconstruit à partir du nouveau groupe de GFAULTCELLS.

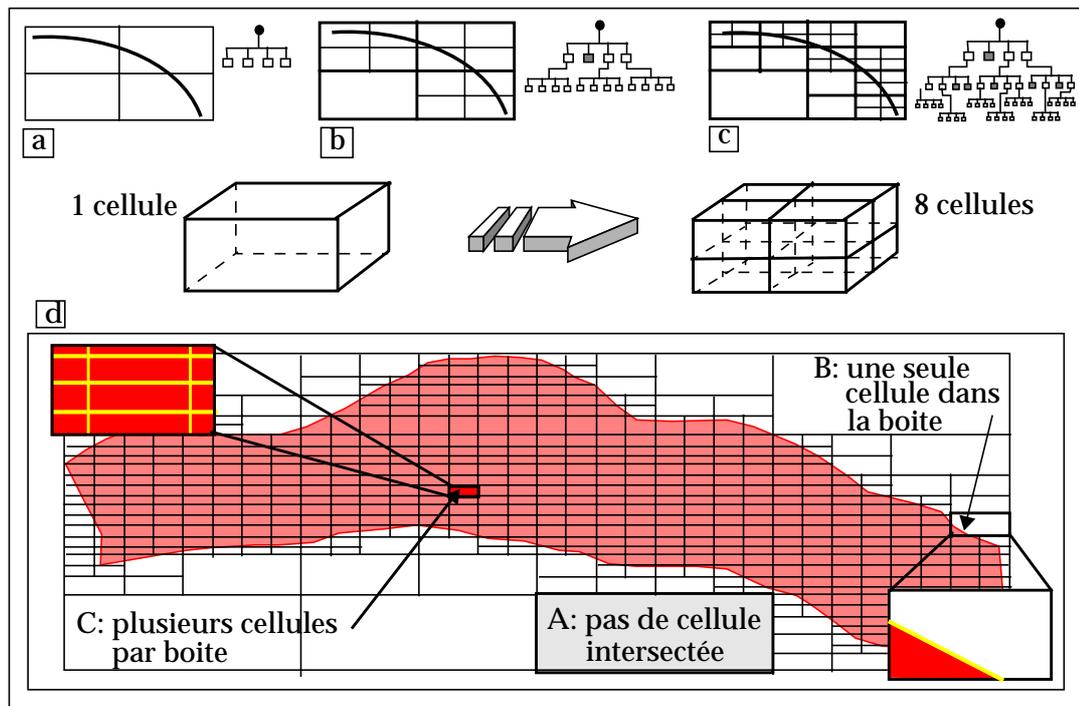


Figure 4-25 : Construction de l'arbre octant par découpages successifs des cellules: un découpage intelligent de l'espace à trois dimensions (vue de dessus).

Une fois construit, il permet de retrouver très rapidement et en un nombre limité de calculs la ou les GFAULTCELLS intersectées par la ligne tracée par le GFAULTSHOOTER. Dans un premier temps, la boîte intersectée par la ligne est sélectionnée. Il ne reste plus ensuite qu'à tester les GFAULTCELLS qu'elle ou ses voisines contiennent.

En fin de recherche, on stocke la cellule la plus proche de l'origine du tir, les coordonnées du point d'impact ainsi que ses coordonnées barycentriques (u,v) dans la cellule [figure 4-26]. On fait en effet intervenir les coordonnées barycentriques du point d'impact dans les calculs des coefficients utilisés par l'interpolateur. Si le point d'impact est très proche du point p0 de la cellule, il aura une plus grande influence sur le point p0 que sur les trois autres points de la GFAULTCELL.

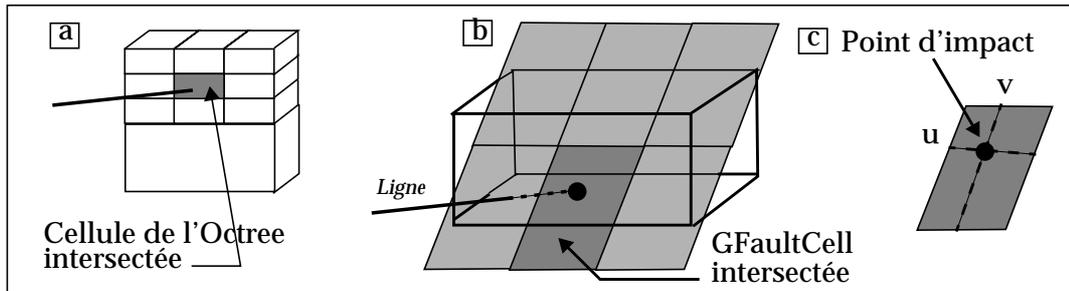


Figure 4-26 : L'octree permet de préselectionner la boîte (a) contenant les GFaultCells (b) parmi lesquelles le point d'impact doit être recherché (c).

#### d) Résultats obtenus

Nous avons testé l'adaptation de la contrainte «attraction par des points» (*fuzzy control points*) sur l'objet faille. Cette contrainte permet d'attirer la faille (contrôlée) par des points de donnée (contrôleur).

Malheureusement, l'interpolateur utilise la connectivité entre les atomes de l'objet contrôlé afin de respecter la contrainte imposée. Du fait de la grande asymétrie de la paramétrisation de l'objet faille (les atomes n'existent que le long de la courbe principale), les résultats ont été relativement décevants: l'interpolation se passe très bien selon la direction de la courbe principale, mais se passe très mal dans la direction perpendiculaire, du fait de l'absence de connectivité de type "atomique" dans cette direction [figure 4-27].

C'est pourquoi nous avons décidé d'implanter la seconde méthode: **la surface cachée**.

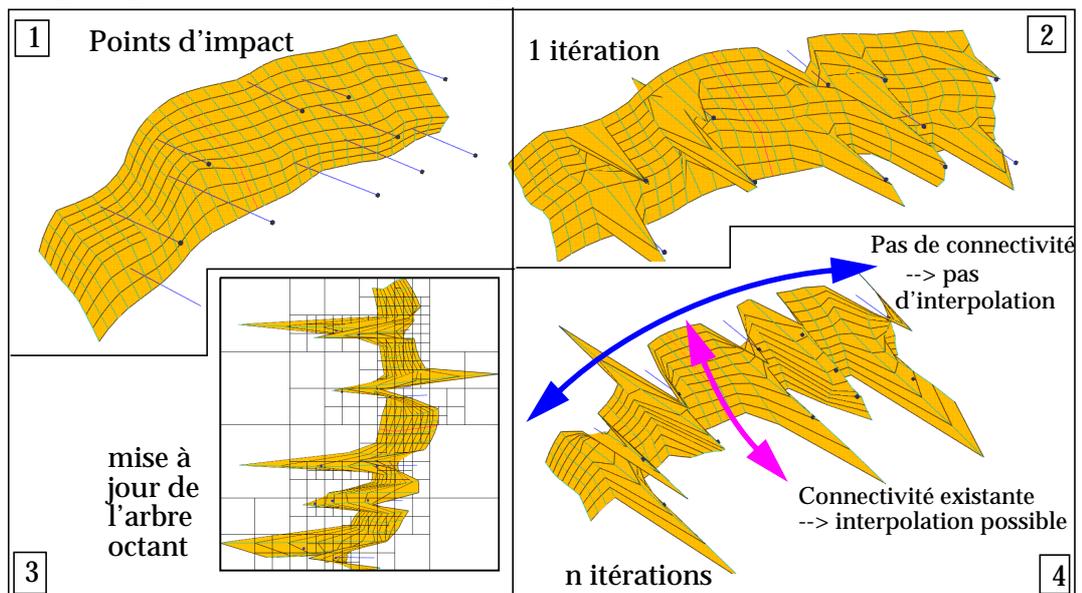


Figure 4-27 : La définition des contraintes directement sur l'objet faille est impossible du fait de l'asymétrie de la paramétrisation.

### 3.2.3 L'association GFaultBundle - TSurf

#### a) Principe

Le principe, même s'il peut paraître plus complexe au départ, est assez simple et permet de réutiliser un maximum de code existant. L'idée a été de se dire que puisque les contraintes sont déjà implantées pour les surfaces triangulées, il suffisait de *stocker une telle surface à l'intérieur du GFaultBundle*. Ainsi, dès qu'une contrainte est installée sur l'objet faille, elle est transmise en interne à la surface "cachée". L'interpolation est réalisée sur la surface triangulée, qui retransmet ensuite les nouvelles positions aux points de l'objet faille. Tout ce mécanisme est **caché à l'utilisateur**, qui ne connaît lui, que l'objet faille avec lequel il travaille.

L'objet faille peut très bien exister sans la surface interne tant que l'on n'applique pas de contraintes sur lui. C'est le cas typique où la création d'une **association** est utile. Au premier ajout de contrainte sur l'objet faille, la surface interne est créée à partir des points des génératrices. En même temps, une propriété est ajoutée sur les atomes de la surface triangulée. Cette propriété permet de stocker l'indice du point de l'objet faille à partir duquel il a été construit.

Ensuite, pour chacune des contraintes que l'on veut pouvoir appliquer sur l'objet faille, il suffit de rajouter une surcouche qui installe automatiquement la contrainte équivalente sur la surface interne. Cette surcouche gère en outre le passage des modifications de la surface triangulée à l'objet faille après une interpolation.

#### b) Exemple: interaction avec des points attracteurs (fuzzy control points)

L'installation de fuzzy control points sur l'objet faille se déroule de la manière suivante [figure 4-28]:

- 1: à l'appel de la commande, un lien de type GFAULTBUNDLEFCPCNSTRLINK est créé entre l'ensemble de points et l'objet faille
- 2: lors de la construction de ce lien, si elle n'existe pas déjà, la surface interne est créée.
- 3: dès que la surface est construite, un lien de type FCPCNSTRLINK classique est créé entre cette surface triangulée et l'ensemble de points contrôleur.

Ensuite, lorsque l'interpolation géométrique de l'objet faille est demandée par l'utilisateur, les étapes suivantes sont réalisées :

- l'interpolation s'effectue sur la surface triangulée
- une fonction est ensuite appelée pour mettre à jour les coordonnées des points de la faille à partir des nouvelles positions des atomes de la surface triangulée:
  - l'indice de chaque point de la faille est calculé:  $index = u * nv + v$
 avec  $u$ : numéro de l'atome,  $nv$ : nombre de points par génératrice et  $v$ : numéro du point sur la génératrice
  - les positions (x,y,z) des atomes de la surface triangulée sont dupliqués dans un tableau, dans le même ordre que les points de l'objet faille
  - pour chaque génératrice, la nouvelle position de l'atome qui la porte est calculée, en fonction des nouvelles coordonnées des points correspondants de la surface triangulée. La base locale est recalculée durant cette étape.
  - les coordonnées des atomes correspondant à la génératrice courante sont transformés dans la nouvelle base locale.

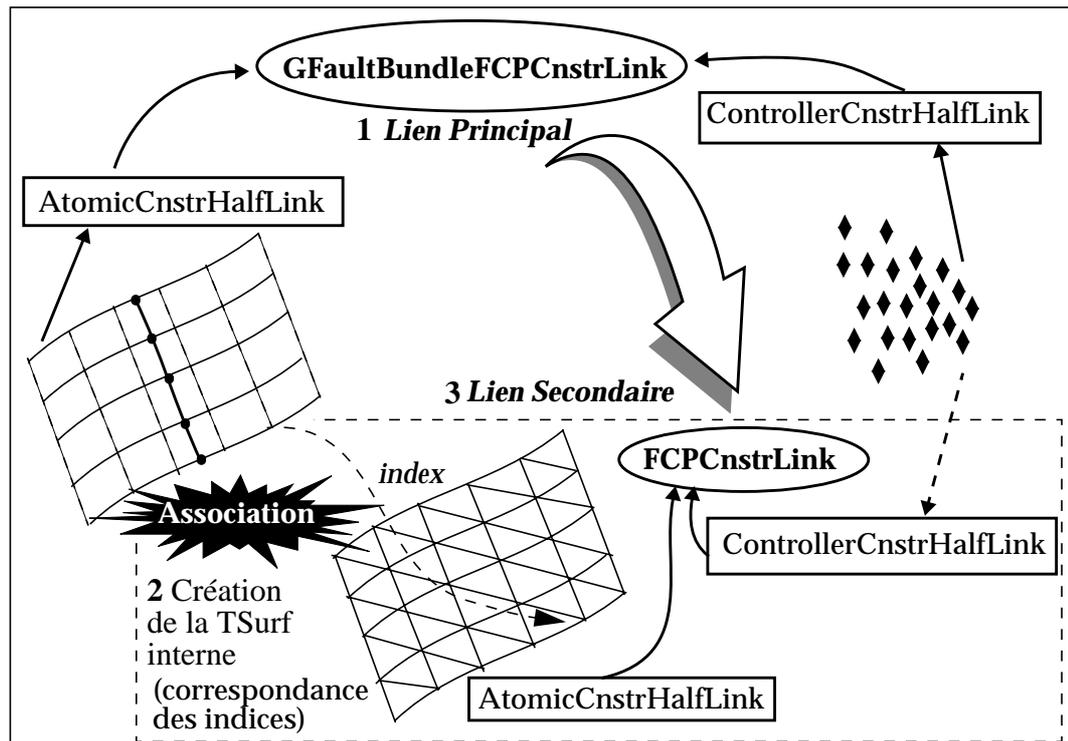


Figure 4-28 : L'installation de contrainte sur l'objet faille provoque l'installation automatique de la contrainte correspondante sur sa surface interne.

### c) Contraintes disponibles

Les contraintes disponibles actuellement sont assez peu nombreuses, mais suffisantes jusqu'à présent. Nous avons déjà vu la contrainte "fuzzy control points" (FCP). De la même manière la contrainte "**Horizon on Fault**" (HOF) a été implantée. Elle force le bord d'un horizon à rester le long de l'objet faille lors de l'interpolation.

Selon le même principe, l'équivalent de la contrainte "bord de surface sur surface" (OTS = *On TSurf*) a été implantée entre deux objets faille. Cette contrainte comporte un degré de complexité supplémentaire, puisque cette contrainte "**Fault On Fault**" (FOF) fait intervenir la contrainte classique *OTS* entre les surfaces triangulées, et deux associations supplémentaires (entre chacun des objets failles et leurs surfaces internes respectives) [figure 4-29]. Cela donne un peu plus de choses à gérer, surtout lorsque l'on souhaite pouvoir modifier la géométrie et/ou la topologie des deux objets faille après avoir installé les contraintes. La seule restriction actuelle, est que l'atome portant la génératrice contrôlée ne doit pas être détruit une fois la contrainte installée.

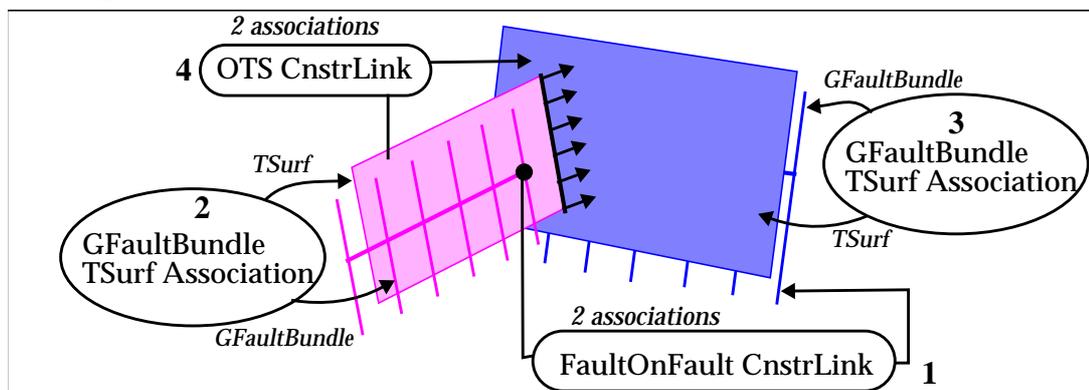


Figure 4-29 : La contrainte "FaultOnFault" met en relation six associations

## 4. Conclusion sur la paramétrisation

La paramétrisation de l'objet faille ayant été définie de manière à correspondre aux données disponibles pour modéliser les failles, elle permet de prendre en compte très rapidement les données fournies au départ et les nouvelles interprétations, comme nous le verrons sur quelques exemples dans le chapitre suivant.

L'héritage des fonctions des lignes polygonales offre de nombreuses fonctionnalités d'édition, notamment l'extension au niveau des extrémités, l'ajout d'un point, la densification. A ces fonctionnalités de base, des fonctions spécifiques à l'objet faille ont été ajoutées, comme la modification du nombre de points par génératrices par exemple. Cet objet peut donc servir pour construire rapidement des modèles de faille.

Une fois ce prototype construit, les transformations de cet objet en un ensemble de lignes polygonales ou en surface triangulée permettent de bénéficier de fonctions d'édition plus précises de ces objets pour parfaire la modélisation.

Cette première utilisation de l'objet faille, n'est aucunement liée aux études d'incertitudes. Le chapitre suivant montre à l'aide de quelques exemples comment ces fonctions permettent de créer rapidement un squelette de faille.

---

---

## Chapitre 5 Exemples d'utilisation de cet objet sur des données réelles

---

---

Après avoir détaillé les principales fonctionnalités de l'objet faille, ce chapitre donne quelques exemples d'utilisation de cet objet sur différents types de données réelles.

La première section détaille la construction d'une faille dans un cas classique simple, où les données sont constituées d'horizons migrés profondeur et de polygones de failles uniquement. La deuxième section donne quelques exemples où des données supplémentaires peuvent être utilisées afin de raffiner la forme de la faille initiale.

### 1. Méthode de construction sur un cas simple

Ce modèle est très simple puisqu'il n'est composé que de trois horizons, d'une faille majeure (*F1*) traversant les horizons de part en part et de trois failles mineures (*f2*, *f3*, *f4*), dont l'une, (*f2*), se branche sur la faille majeure. Cette faille majeure s'amortit au niveau de l'horizon supérieur [figure 5-1].

Sur ce jeu de données, les polygones et les bords des ensembles de points sont concordants, mise à part une zone très restreinte. Nous verrons un peu plus loin que cela n'est pas toujours le cas, loin de là.

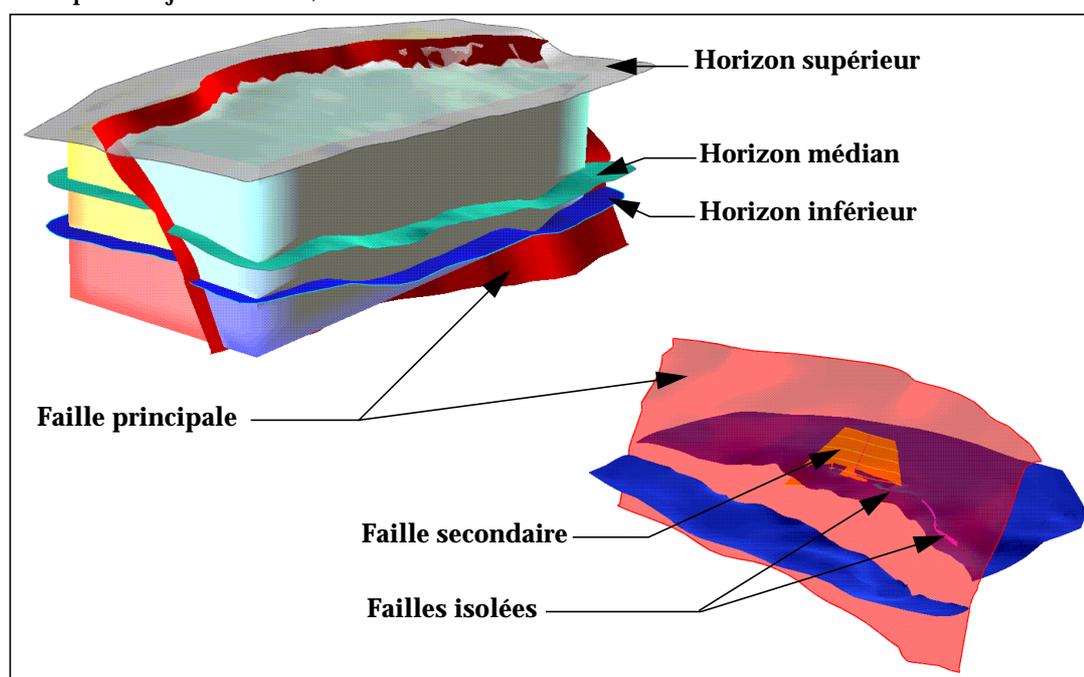


Figure 5-1 : Modèle simple composé de trois horizons, d'une faille principale, d'une faille synthétique se branchant sur cette faille principale et de deux failles isolées (*f3* et *f4*)

## 1.1 Principe général de la construction

### 1.1.1 Description des données

Les données disponibles pour construire ce modèle étaient relativement restreintes:

- **un ensemble de points migrés profondeur par horizon**

L'interprétation des horizons se fait d'abord en domaine temps (voir Chapitre 2). Il faut donc les transformer en domaine profondeur afin de réaliser la construction du modèle structural. Nous avons eu directement les points migrés et nettoyés à notre disposition (aucun point situé dans la zone de faille).

- **des polygones de failles pour les deux horizons inférieurs.**

Ces polygones de failles correspondent à des courbes dessinées par le géologue structuraliste sur une carte. En général, les polygones permettent de réduire l'incertitude sur la position exacte de la faille dans l'intervalle libre entre les deux bords de l'horizon.

Ces polygones de failles étant dessinés en carte, seules les coordonnées (x,y) sont disponibles, la coordonnée z étant par défaut mise à z=0.

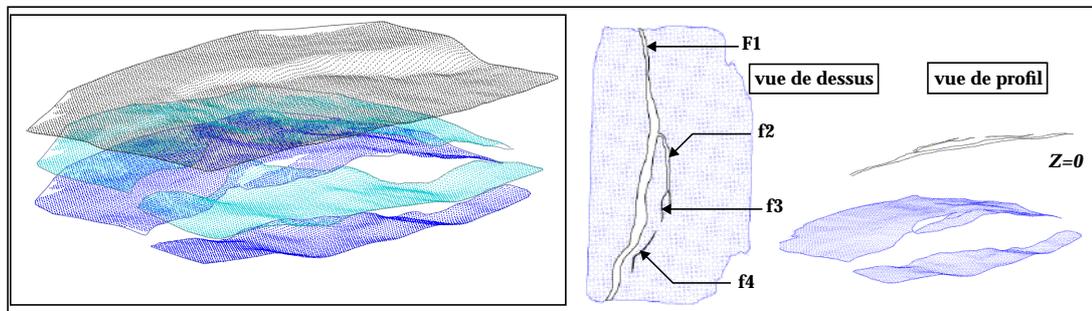


Figure 5-2 : Données disponibles pour le modèle simple: 3 ensembles de points migrés profondeur

### 1.1.2 Méthode de construction

Afin de construire les failles du modèle, la méthode consiste, pour chaque faille, à calculer la *ligne médiane* entre les deux bords de l'horizon qui lui correspondent. Le fait d'utiliser des lignes médianes au lieu d'utiliser directement les bords de l'horizon, évite de construire des failles "en forme d'escalier" [figure 5-3], [76 - Dubrule 1].

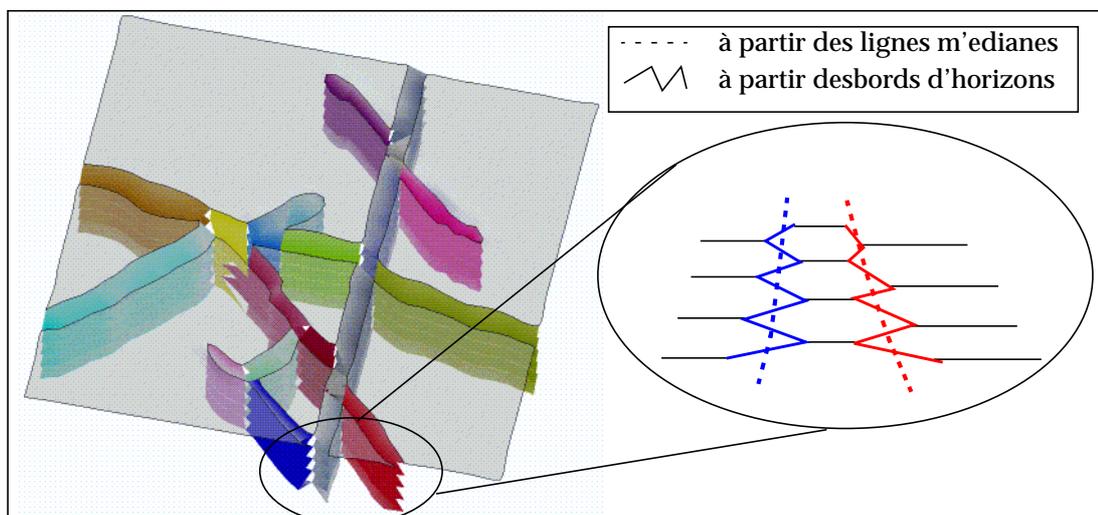


Figure 5-3 : Construire les lignes médianes à partir des bords des horizons évite les effets d'escalier (données fournies par EXXON).

La construction s'effectue en quatre étapes principales:

1. **projection des polygones de failles** sur les horizons afin d'obtenir des coordonnées Z correctes.  
Pour cette première étape, trois approches sont proposées dans le paragraphe suivant.
2. **corrélation des bords** d'horizons qui correspondent à une même faille, non seulement sur le même horizon mais aussi d'un horizon à l'autre  
Pour cette *étape capitale*, deux approches sont proposées dans ce chapitre. Une troisième méthode, beaucoup plus performante pour des jeux de données complexes, sera proposée plus loin (voir Chapitre 6).
3. pour chaque horizon, **calcul des lignes médianes** pour chacune des failles du modèle, à partir des lignes corrélées  
La technique est assez simple et sera décrite plus loin. Pour le cas particulier de ce jeu de données, où la faille s'amortit au niveau de l'horizon supérieur, l'extraction de la ligne médiane à partir du calcul de la courbure sera présenté.
4. **rassemblement des lignes médianes** correspondant à une même faille pour créer un objet faille.

## 1.2 Projection des polygones de failles à la bonne altitude

Afin de transformer les polygones de failles pour qu'ils soient localisés à la bonne profondeur, différentes approches ont été utilisées. La première d'entre elles a été mise au point durant mon stage de D.E.A. [81 - Lecour 1]. Elle comporte de *nombreux inconvénients*, dont un temps d'exécution beaucoup trop lent pour être utilisée de manière efficace.

La deuxième méthode est une *méthode de projection* qui me semble beaucoup plus efficace que la précédente. Cette méthode est largement utilisée dans les compagnies pétrolières utilisant gOcad.

Enfin, une dernière approche correspond aux résultats obtenus par Isabelle Duvinage lors de sa thèse [77 - Duvinage 1], [103 - Duvinage 3], et permet de calculer les bords des horizons de manière semi-automatique, même lorsque les polygones de failles ne sont pas fournis.

### 1.2.1 Méthode de "l'emporte-pièce"

Cette méthode consiste à construire une surface continue correspondant à l'ensemble de points dans un premier temps. Le polygone de faille est ensuite dupliqué de telle manière que les deux courbes obtenues se trouvent de part et d'autre de l'horizon. A partir de ces deux courbes, une surface triangulée verticale est construite. Cette surface est ensuite utilisée comme emporte-pièce pour découper la surface continue [figure 5-4].

Il ne reste plus ensuite qu'à déterminer quelles sont les parties à conserver et quelles sont les parties à détruire (ces dernières sont situées à l'intérieur de l'emporte-pièce). Une technique classique de géométrie algorithmique a été utilisée: pour chacune des parties créées, une ligne horizontale est tracée entre un point situé sur cette partie en direction de la surface verticale. Si le nombre d'intersections avec l'emporte-pièce est impair, la partie est gardée, dans le cas contraire, elle est détruite.

Cette technique permet d'obtenir de bons résultats, mais la dernière étape est longue (les opérations de tirs sur la surface verticale sont assez lents). De plus, la méthode qui permet de découper la surface continue par l'emporte pièce donne un maillage qui nécessite des opérations de nettoyage pour que la surface puisse être utilisée par la suite. C'est pourquoi nous avons utilisé d'autres techniques.

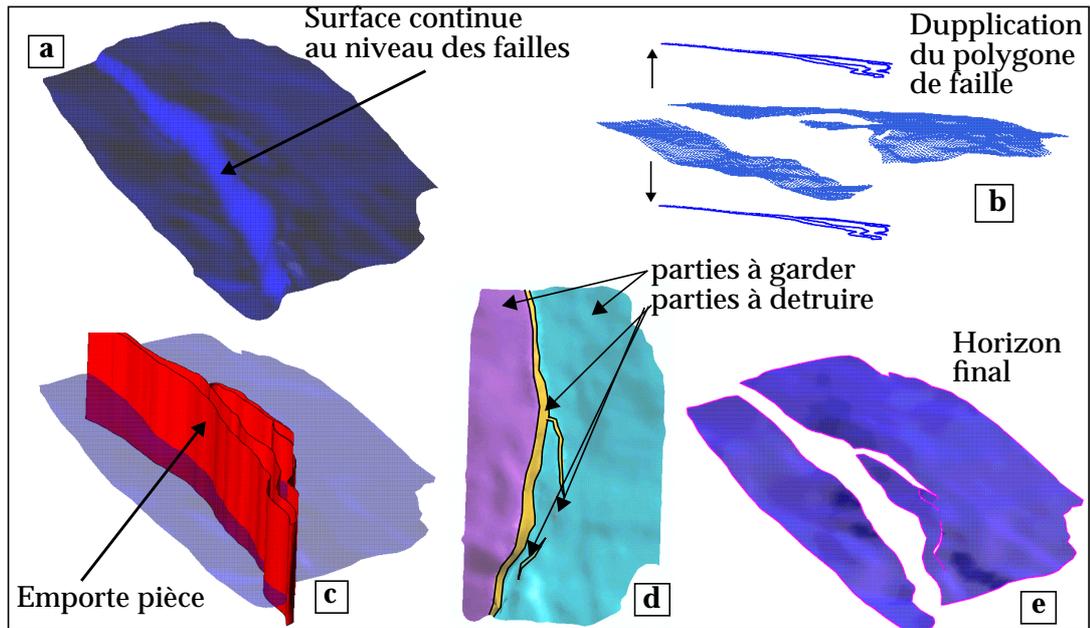


Figure 5-4 : Méthode de "l'emporte-pièce". [a]: surface continue; [b]: translation du polygone de faille de part et d'autre de l'horizon; [c]: construction de l'emporte pièce; [d]: découpage de la surface continue; [e]: surface finale.

### 1.2.2 Méthode de projection

Cette méthode de projection a été proposée par Philippe Samson (Elf Exploration Production) et est beaucoup plus rapide que la méthode précédente.

Elle consiste à rechercher pour chaque point du polygone ( $p$ ), le point de l'horizon ( $h$ ) le plus proche en distance horizontale. La valeur  $Z$  du point  $h$  ainsi trouvé est ensuite affectée au point  $p$  du polygone. Ceci revient à rechercher le point  $h$  le plus proche du point  $p$  après avoir projeté l'horizon sur un plan horizontal [figure 5-5].

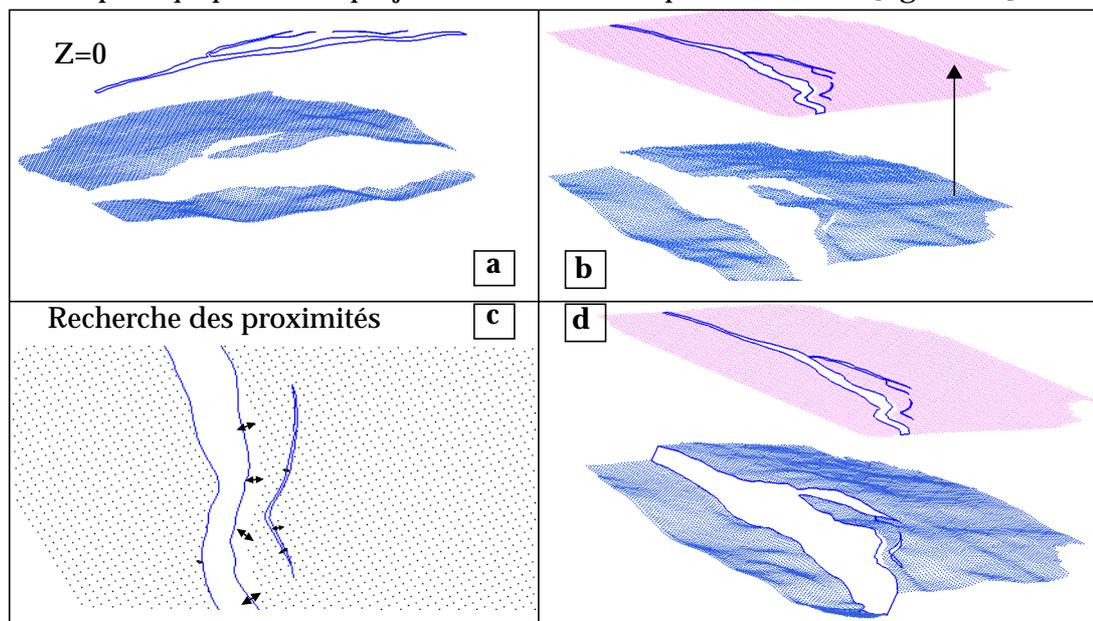


Figure 5-5 : Projection du polygone sur l'ensemble de points. [a]: données initiales; [b]: plan de recherche des proximités; [c]: recherche des correspondances entre les points du polygone et ceux de l'horizon sur le plan horizontal; [d]: affectation des bonnes coordonnées  $Z$  et polygone de faille projeté à la bonne profondeur.

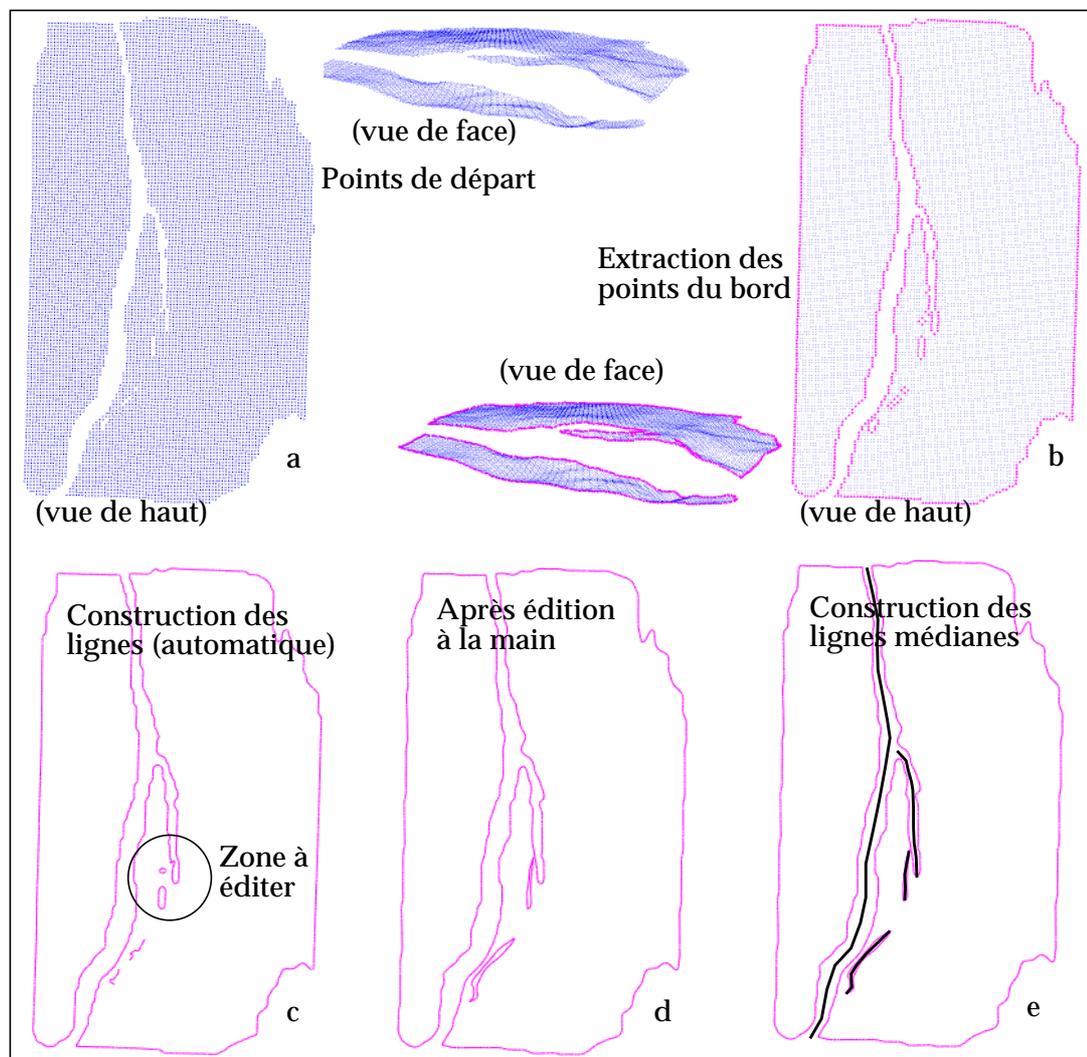
### 1.2.3 Construction des polygones à partir des horizons

Une troisième méthode a été mise au point durant la thèse d'Isabelle Duvinage [103 - Duvinage 3].

Cette méthode utilise des techniques de morphologie mathématique, qui permettent de nettoyer les données sismiques (retrait des points situés dans les zones correspondant à des failles, fermeture des trous ne correspondant pas à des failles).

Une autre utilisation de ces techniques est l'*extraction automatique des points situés sur les bords* de l'horizon. Une fonction permet ensuite de construire une première ligne interpolée à partir des points extraits. Dans les zones où le rejet des failles est assez faible, ou dans les zones complexes du point de vue structural, une édition de ces courbes est nécessaire [figure 5-6].

Cette méthode permet d'obtenir un résultat beaucoup plus rapidement qu'une fastidieuse digitalisation sur l'ensemble de points, lorsque les polygones de failles ne sont pas disponibles.



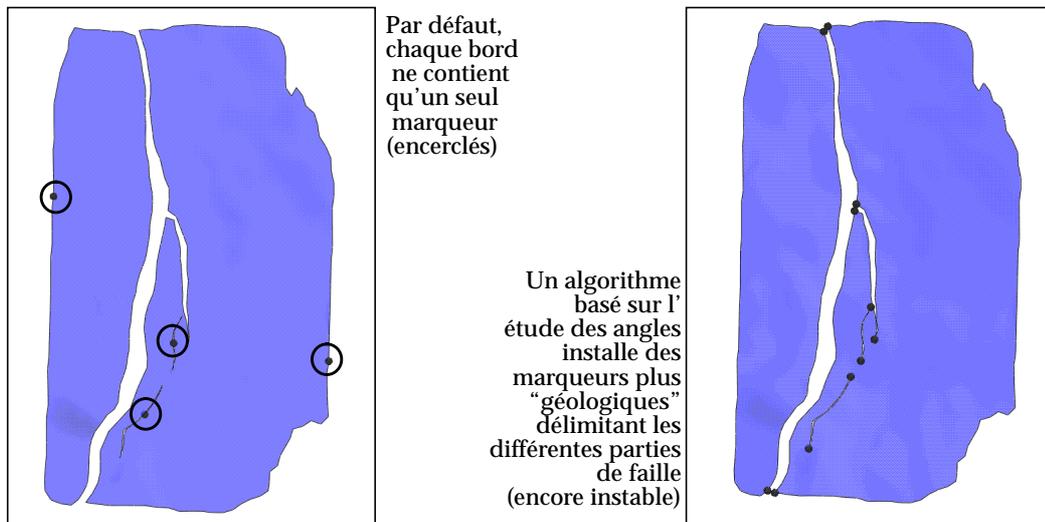
**Figure 5-6 :** Construction des lignes médianes à partir des horizons uniquement, en utilisant les méthodes de morphologie mathématique. [a]: ensemble de points initial; [b]: extraction des points du bord; [c]: construction automatique de lignes polygonales; [d]: Edition des lignes dans les zones délicates; [e]: construction automatique des lignes médianes.

### 1.3 Construction des lignes médianes

Une fois le polygone de faille projeté à la bonne profondeur, il reste à corrélérer les parties de lignes qui correspondent à une même faille afin de construire la ligne médiane entre les deux bords d'horizons.

#### 1.3.1 Corrélation des lignes grâce à l'emporte pièce

La première méthode utilisée a elle aussi été mise au point durant mon stage de D.E.A. . Pour cette méthode, il faut commencer par délimiter la courbe du bord en différentes sous-parties: il faut séparer les bords externes des bords correspondant aux failles, et dans ces derniers, découper la ligne au niveau des zones de branchement. Il existe des algorithmes pour effectuer ce découpage de manière semi-automatique (en utilisant des critères d'angle par exemple), mais ces algorithmes sont instables lorsque le bord de l'horizon n'est pas assez lisse.



**Figure 5-7 :** Installation de marqueurs afin de délimiter les bords externes des failles, ainsi que les zones de connexion entre failles.

Une fois ce pré-découpage effectué, un processus permet de déterminer pour chaque point de la courbe du bord de l'horizon, à quelle partie de la surface "emporte-pièce" il correspond. Une ligne est tracée à partir de ce point, en direction de la surface verticale. Le point d'impact le plus proche de l'origine de la ligne horizontale est ensuite sélectionné.

La dernière étape consiste à affecter le bord auquel le point origine appartient à une faille donnée. Comme la surface verticale est constituée de différents morceaux, il est facile de séparer les bords correspondant aux failles f3 et f4. Par contre, la partie correspondant aux failles F1 et f2 est en un seul morceau et un post-traitement doit être employé afin de séparer les deux ensembles de bords.

Durant mon D.E.A., un premier algorithme qui donne de très bon résultats sur ce jeu de données simple, où la faille principale n'est reliée qu'à une seule faille secondaire a été proposé. Malheureusement, cet algorithme serait inutilisable sur des jeux de données plus complexes.

Il consiste, en quelques mots, à sélectionner les points à partir desquels la ligne horizontale ne rencontre que deux fois la surface verticale. Le plan moyen de cet ensemble de points est ensuite calculé et utilisé pour séparer les points qui restent (ceux qui donnent plus de deux intersections avec la surface verticale), en utilisant un critère de distance.

Cet algorithme suppose donc qu'il existe une zone suffisamment importante, au niveau de laquelle la faille principale ne contient pas de branchement. Cette hypothèse est beaucoup trop forte pour pouvoir être utilisée sur de nombreux jeux de données.

De plus, le processus permettant de tirer à partir des points du bord vers la surface verticale est assez lent et rend l'utilisation de l'algorithme difficile sur des jeux de données plus complexes. C'est pourquoi le travail sur cet algorithme n'a pas été poussé plus loin et que nous avons utilisé d'autres techniques par la suite.

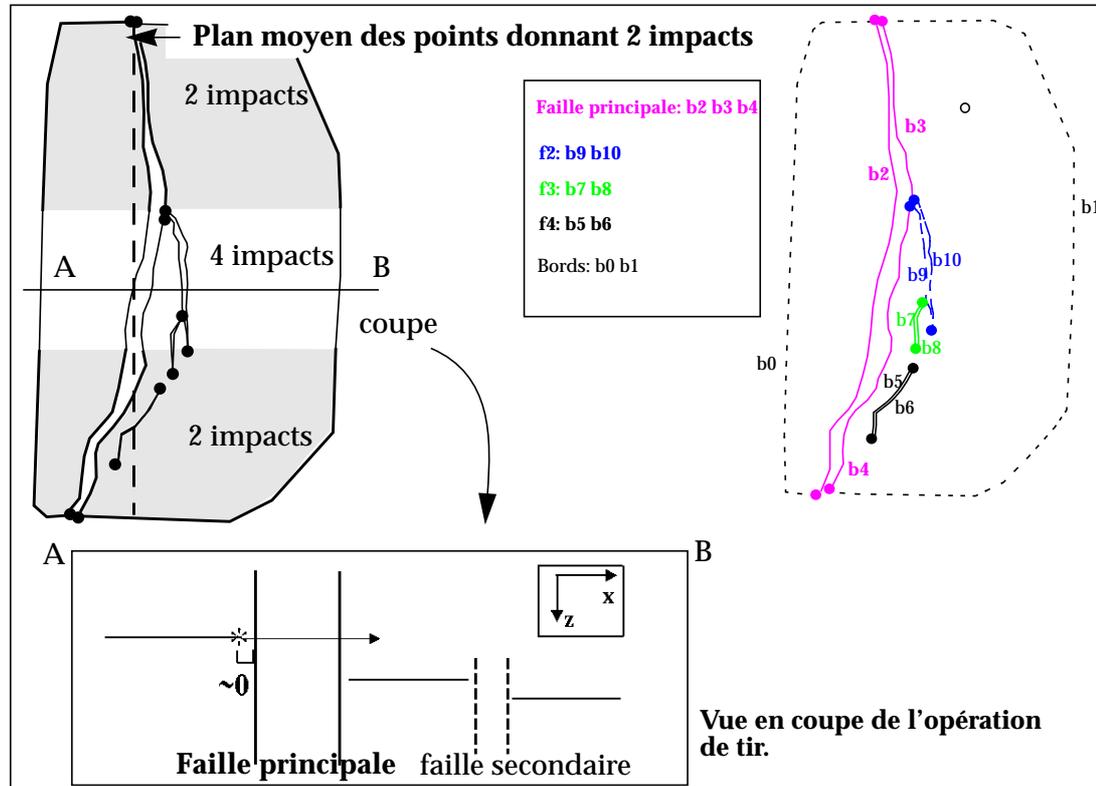


Figure 5-8 : Corrélation des bords par tirs successifs. [a]: découpage des courbes au niveau des fourches; [b]: opération de tir; [c]: classification des bords en fonction de la faille à laquelle ils appartiennent.

### 1.3.2 Corrélation des lignes à la main

Une autre méthode, *beaucoup plus fastidieuse* consiste à découper le polygone de faille projeté au niveau des failles secondaires, puis à regrouper à la main les morceaux qui correspondent à une même faille. Une fonction permet en effet de construire une courbe unique (PLINE) à partir de plusieurs morceaux de courbes.

Cette technique a l'avantage de marcher dans tous les cas, mais ne peut être appliquée que sur des jeux de données où le nombre de failles à construire est relativement restreint.

Au début de ce travail de thèse, nous avons utilisé beaucoup cette technique, en attendant les résultats obtenus par Isabelle Duvinage [102 - Duvinage 2]. Comme cela sera montré plus loin (Chapitre 7), elle a mis au point une technique quasi-automatique qui permet de relier tous les morceaux de bords qui correspondent à une même faille, non seulement sur un horizon, mais aussi d'un horizon à l'autre.

Bien que cette technique ne soit pas encore parfaite, vu la complexité de certains jeux de données, elle permet d'obtenir une solution initiale souvent proche du résultat souhaité et permet d'aller beaucoup plus vite dans la construction des lignes médianes.

### 1.3.3 Construction des lignes médianes

Une fonction simple permet de construire la ligne médiane entre deux I LINES d'une même PLINE. Mais il faut que la PLINE ne contienne que deux I LINES. La première étape de la construction consiste donc à ajouter un segment sur la faille principale au niveau de chacune des fourches, afin de relier les deux morceaux du bord de l'horizon situés de part et d'autre de la fourche [figure 5-9].

La fonction rééchantillonne ensuite les deux I LINES avec le même nombre de points et calcule pour chaque couple, le point milieu correspondant. La ligne médiane est obtenue en reliant tous les points milieu par des segments.

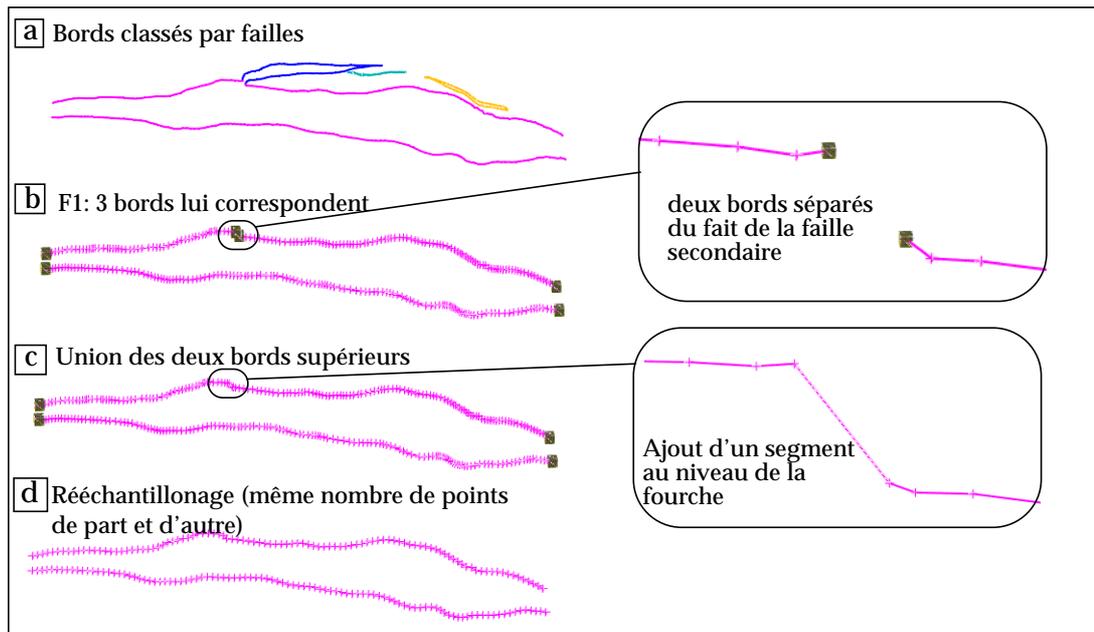


Figure 5-9 : préparation des courbes avant la construction de la ligne médiane.

Lorsque les incertitudes sont grandes, notamment au niveau des failles listriques, elles sont dissymétriques: la faille se trouve plus proche du côté toit que du côté mur. C'est pourquoi, la fonction précédente autorise l'utilisateur à spécifier une distance normalisée à laquelle la ligne "médiane" doit être construite (0 correspond à l'une des courbes, 1 à la deuxième courbe et 0.5 à la vraie médiane) [figure 5-10].

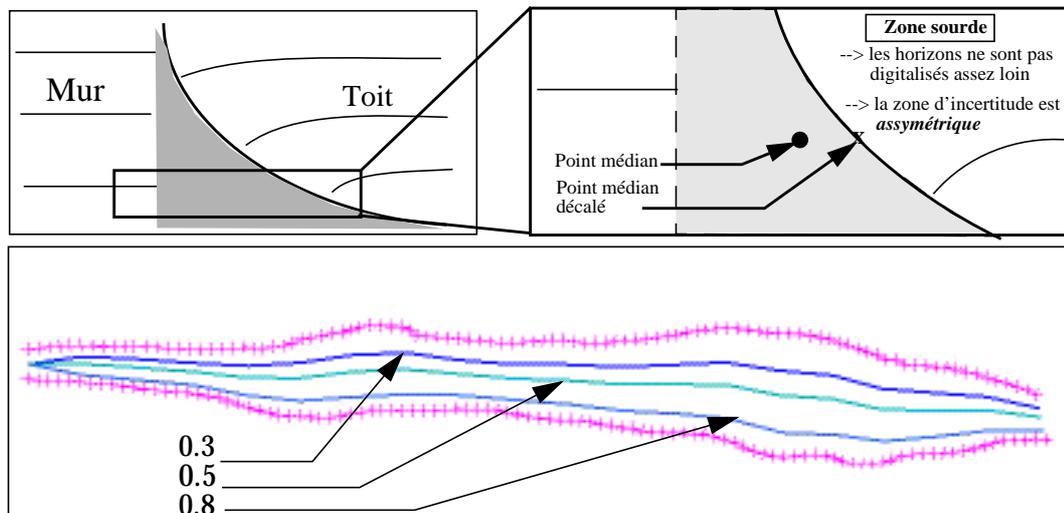


Figure 5-10 : Il est possible de construire la ligne "médiane" plus proche de l'une des deux lignes, lorsque l'utilisateur le demande.

### 1.3.4 Cas particulier de l'horizon supérieur

Comme cela a été souligné plus haut, la faille principale s'amortit au niveau de l'horizon supérieur et les techniques précédentes ne peuvent être utilisées pour construire la ligne médiane.

Pour cet horizon, nous avons calculé les courbures puis extrait les courbes d'isovaleurs correspondant à une courbure nulle. Parmi toutes les lignes construites, nous n'avons conservé que celle correspondant à la faille principale [figure 5-11].

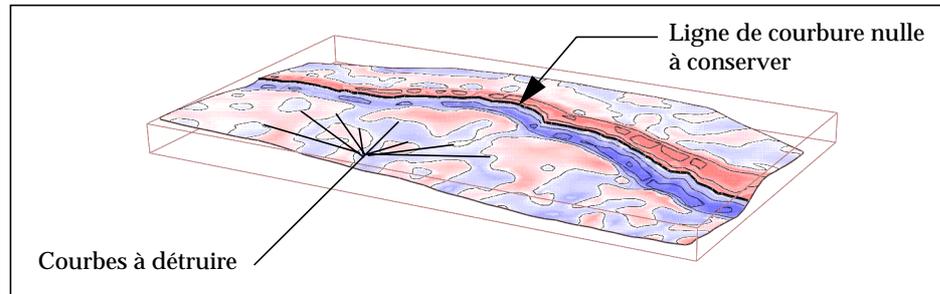


Figure 5-11 : Calcul de la courbure pour l'horizon supérieur et sélection de la courbe correspondant à la trace de la faille.

## 1.4 Construction de la faille

### 1.4.1 Faille initiale

Une fois que les courbes médianes entre les bords des différents horizons sont construites, une commande permet de les relier afin de créer un objet faille. L'utilisateur doit donner le nom de la faille à créer, ainsi que le nombre de points sur les génératrices. Dans un deuxième temps, il lui est demandé de sélectionner successivement les trois courbes. La commande calcule pour chacune des trois courbes l'atome situé à l'abscisse curviligne 0.5. Le backbone est construit à partir de ces trois atomes, puis chaque courbe médiane est attachée à l'atome qui lui correspond [figure 5-12].

Comme on peut le voir sur la figure, les génératrices correspondent parfaitement aux courbes médianes initiales, à condition que l'utilisateur ait demandé un assez grand nombre de points pour décrire les génératrices.

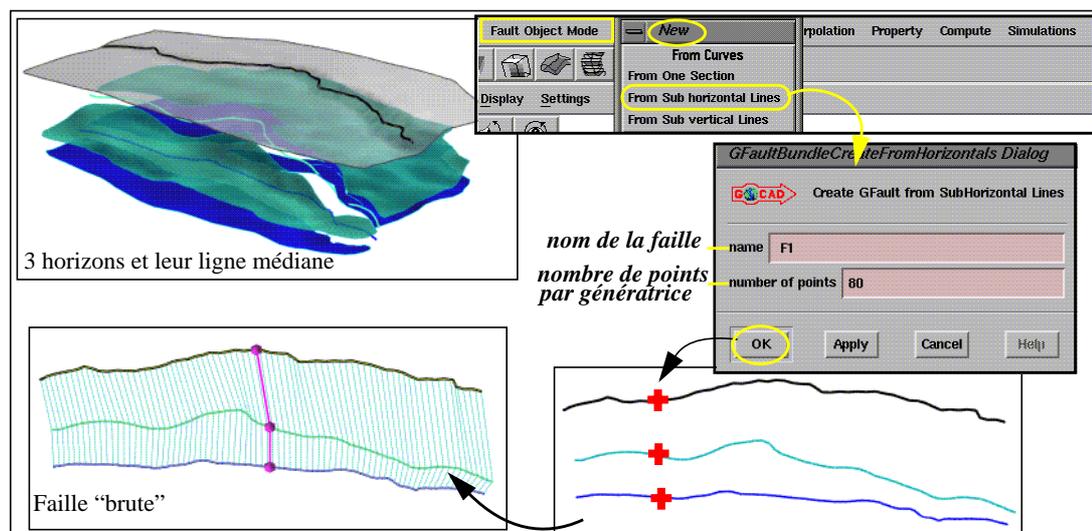


Figure 5-12 : Construction de l'objet faille à partir de lignes horizontales: il suffit de cliquer successivement sur les lignes médianes après avoir donné le nom de la faille et le nombre de points par génératrice.

### 1.4.2 Edition rapide de la faille brute: rééchantillonnage et extension

Comme nous l'avons expliqué auparavant, du fait de l'héritage des fonctions des lignes polygonales, il est possible de **densifier** la courbe principale afin qu'elle contienne plus de nœuds et soit plus facilement éditable. Dans ce cas, une génératrice est automatiquement interpolée à partir des génératrices initiales pour chaque nouveau nœud créé.

De la même manière, il est possible d'utiliser la fonction d'**extension** d'un segment situé à une extrémité, et la génératrice correspondante est automatiquement créée [figure 5-13].

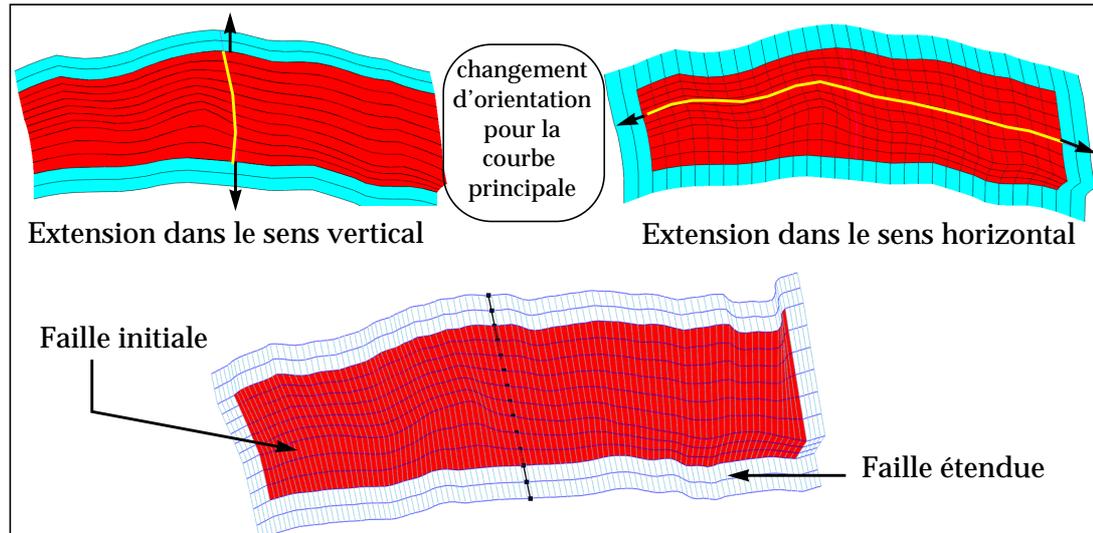


Figure 5-13 : Extension de la faille dans ses deux directions principales

Enfin, un algorithme permet de **modifier le nombre de points** par génératrices. Si le nombre de points est augmenté, de nouvelles propriétés sont d'abords ajoutées à l'objet faille.

La première étape de cet algorithme consiste à stocker tous les points de la faille dans un tableau à double entrée, génératrice par génératrice. Pour chaque génératrice, un nouvel échantillonnage est calculé en fonction du nombre de points demandé par l'utilisateur. La position de l'atome du backbone est recalculée en fonction des nouveaux points. Enfin, pour chaque génératrice, la base locale est recalculée en fonction des nouvelles coordonnées de l'atome du backbone, puis les coordonnées des points correspondants transformées dans la nouvelle base locale [figure 5-14].

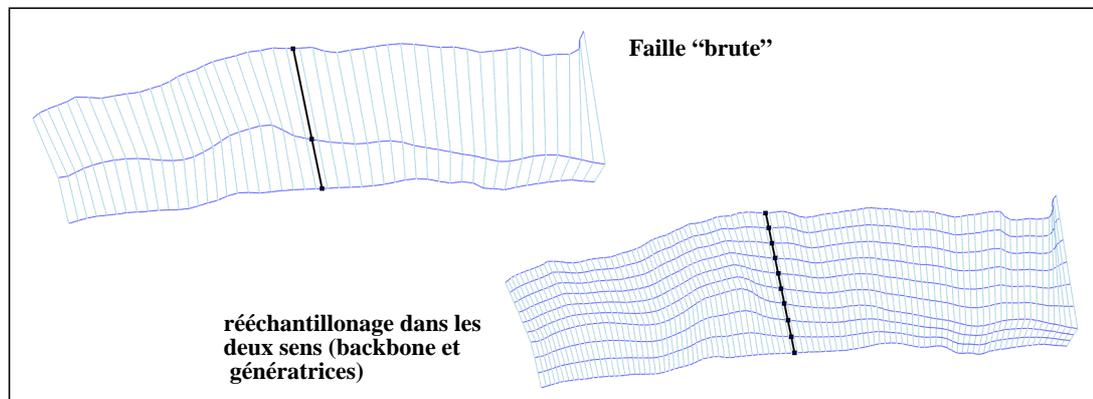


Figure 5-14 : Fonctions d'édition simple sur la faille initiale [a]; [b]: rééchantillonnage dans les deux directions principales de la faille.

### 1.4.3 Ajustement aux données

La dernière étape de la modélisation de la faille est de prendre en compte d'autres lignes d'interprétation. Comme nous l'avons expliqué plus haut, l'objet faille comporte un système d'associations qui permettent de lui appliquer un certain nombre de contraintes.

L'une de ces contraintes consiste à ajuster sa géométrie à un ensemble de courbes ou de points de données et est appelée "fuzzy control points" ou FCP.

Lorsque la commande est appelée par l'utilisateur, la surface associée à la faille est créée si elle n'existait pas déjà, et la contrainte FCP lui est appliquée. Dans le même temps, une autre contrainte est installée sur les bords de la surface. Cette deuxième contrainte n'autorise les points du bord qu'à se déplacer le long d'un plan. Ceci évite que la surface ne rétrécisse durant l'interpolation.

Lorsque l'interpolation de la géométrie de la faille est demandée, elle est en fait appliquée à la surface triangulée qui transmet ensuite les nouvelles positions des points l'objet faille qui lui est associé. Ceci s'effectue en utilisant le système d'indices donnant la correspondance entre les points de la faille et les points de la surface correspondante [figure 5-15].

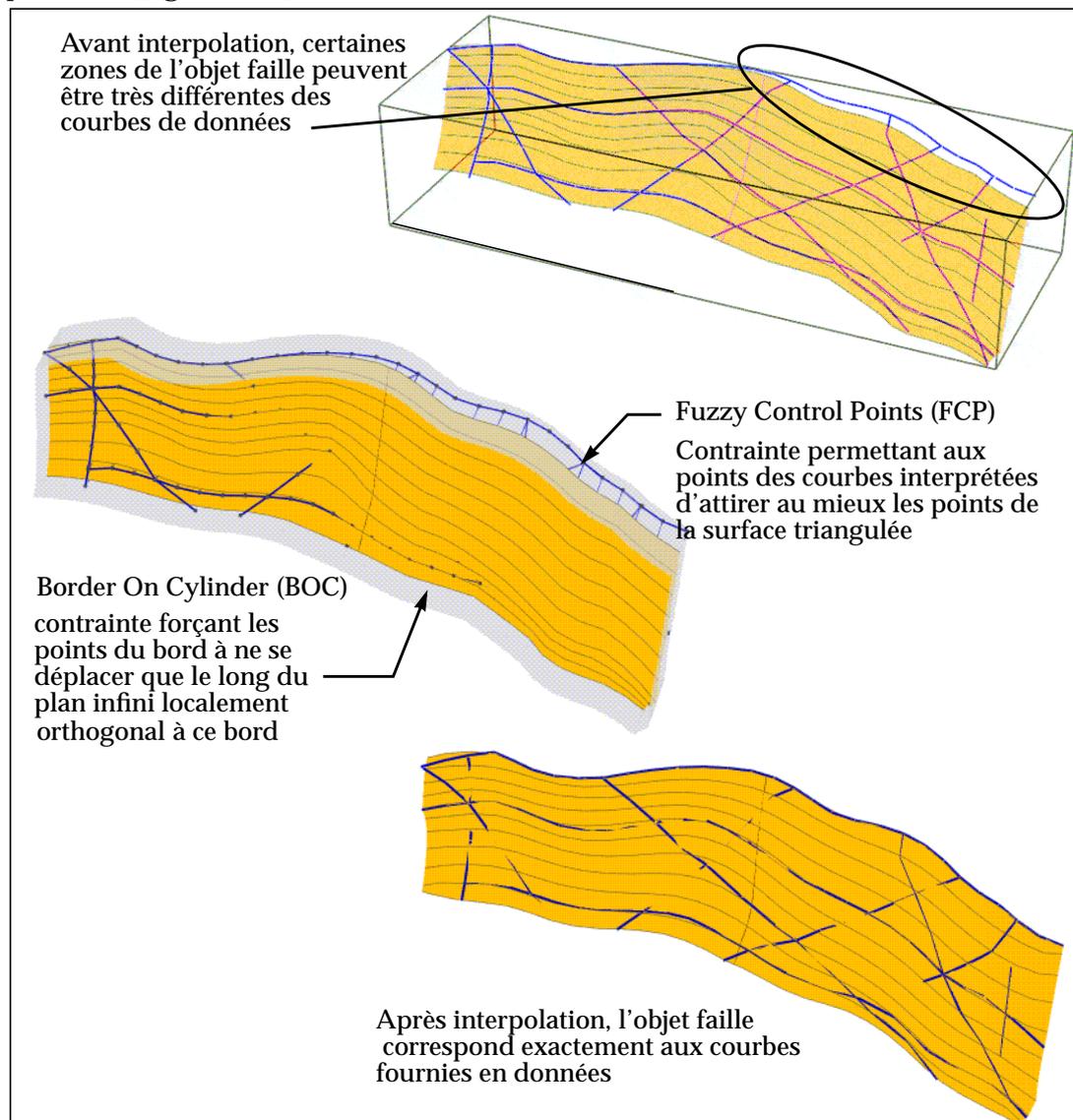


Figure 5-15 : Ajustement de la géométrie de la faille à de nouvelles données en utilisant la contrainte "FCP".

## 2. Méthode de construction sur des jeux de données plus complexes

L'exemple que nous avons étudié précédemment était assez simple, afin de pouvoir détailler les premières étapes de la modélisation des failles à l'aide de ce nouvel objet. Cependant, il arrive parfois que du travail supplémentaire doive être effectué dans des cas où les données sont imprécises ou incohérentes.

Ce paragraphe montre comment d'autres fonctionnalités de cet objet peuvent aider le géologue à éditer rapidement la faille initiale. Dans les deux cas étudiés ci-dessous le début de la construction est identique à celle présentée pour le jeu de données précédent.

### 2.1 Edition à partir de courbes interprétées

Lorsque l'on construit la faille uniquement à partir des horizons correspondant à la *zone réservoir*, on n'obtient qu'une toute petite partie de la faille. Sur l'un des jeux de données fournis par le centre de recherche Elf-GRC (Londres), un certain nombre d'interprétations de failles donnant beaucoup plus d'informations sur l'extension verticale de la faille étaient disponibles.

L'autre avantage de ce jeu de données est que nous avons à notre disposition non seulement les données, mais aussi le modèle final réalisé par Christophe Basire au cours de son VSNS [64 - Basire] en collaboration avec des structuralistes. Ceci nous donne la possibilité de comparer nos résultats avec le modèle final.

Un algorithme qui permet de construire un objet faille de manière automatique à partir de toutes ces données (horizons migrés profondeur et interprétations de failles) commence par la sélection de toutes les extrémités des courbes fournies en données. Ensuite, l'enveloppe convexe correspondant à cet ensemble de points est calculée. Cette enveloppe peut être ajustée au niveau des concavités si l'utilisateur le demande. La deuxième étape consiste à couper ces lignes par des plans horizontaux, et à construire des courbes "génératrices" à partir des points d'intersection entre les interprétations de failles et les plans horizontaux. Ces courbes sub-horizontales servent ensuite à construire les génératrices de l'objet faille.

**Remarque;** Afin que la faille construite corresponde exactement à la courbe enveloppe, un cas particulier est fait pour les deux génératrices extrêmes. Les plans horizontaux sont calculés de telle manière qu'ils ne couvrent pas toute l'étendue verticale de la faille. Les deux génératrices extrêmes sont prises directement comme des sous-parties de la courbe enveloppe en fin de processus [figure 5-16].

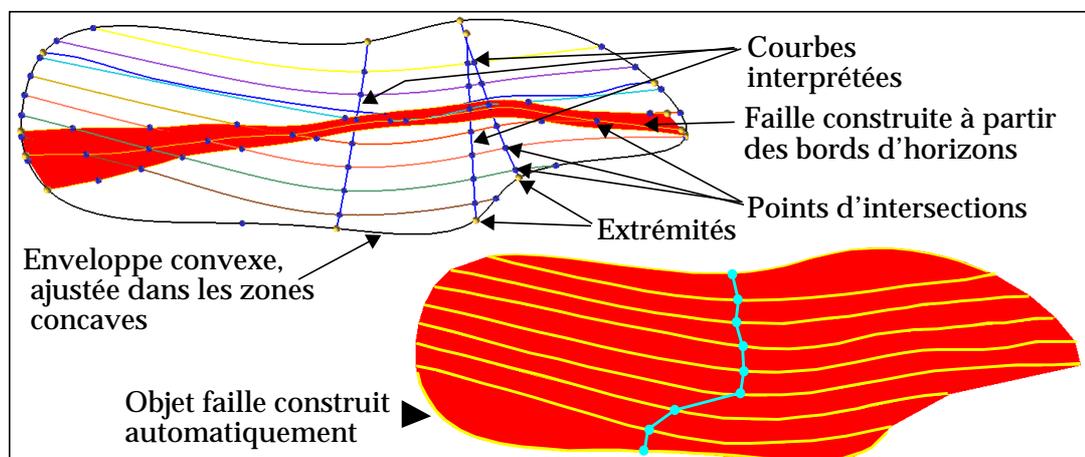
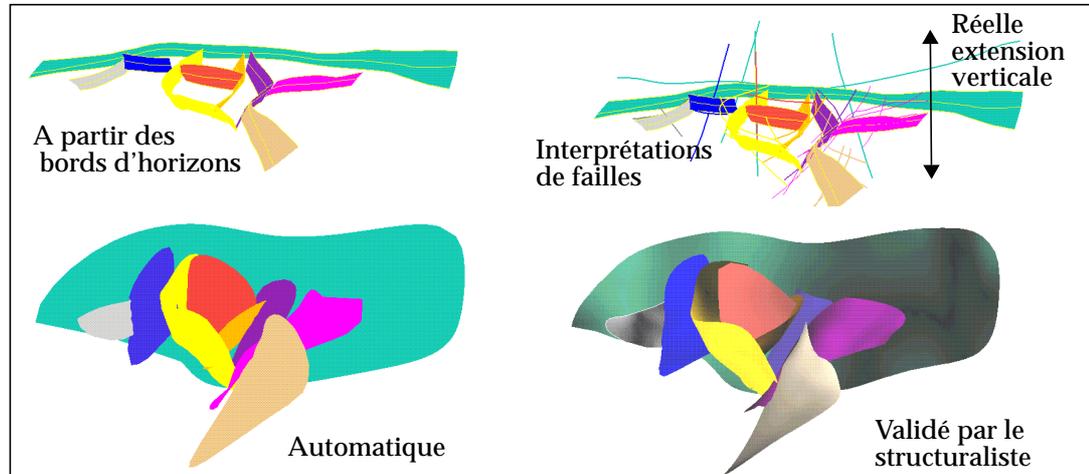


Figure 5-16 : Différents éléments permettant de passer automatiquement d'une faille partielle construite uniquement à partir des bords d'horizons, à une faille complète prenant en compte toutes les données disponibles.

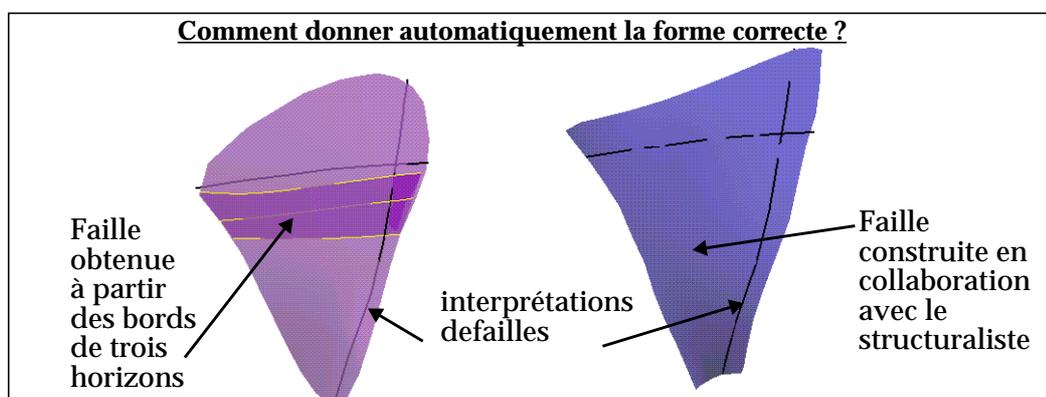
Le modèle final construit par Christophe Basire m'a permis de comparer les résultats obtenus par cette méthode automatique et ceux obtenus par l'édition de surfaces triangulées classiques, en collaboration avec un géologue structuraliste. Comme le montre la [figure 5-17]. Le résultat automatique est très proche du modèle approuvé par le structuraliste. Il reste à modifier les failles secondaires au niveau de la zone de branchement afin que le modèle soit bien fermé.



**Figure 5-17 :** [a]: failles construites à partir des bords d'horizons; [b]: les interprétations de failles donnent une information sur l'extension verticale des failles; [c]: réseau de faille obtenu par le processus automatique; [d]: modèle validé par le géologue structuraliste.

**Remarque:** un résultat similaire aurait pu être obtenu en construisant une surface de faille directement à partir de l'ensemble de points extraits à partir des courbes interprétées. La méthode présentée ici permet en plus de prendre en compte l'extension horizontale de la faille donnée par les bords d'horizons, lorsque l'interprétation de la faille n'a pas été poussée assez loin latéralement. Elle permet en outre de construire directement un objet faille qui à mon avis est plus facilement éditable qu'une surface triangulée.

Les connaissances du structuraliste sont indispensables pour valider le modèle obtenu. Construire un modèle de manière purement automatique me paraît impossible, à moins d'avoir beaucoup plus de données disponibles. Cependant, ce résultat automatique peut facilement être utilisé comme base par le structuraliste afin de vérifier la cohérence du modèle, d'autant plus que la dernière fonctionnalité de l'objet faille présentée dans ce chapitre permet de l'étendre rapidement sur n'importe quelle surface (horizon ou faille).

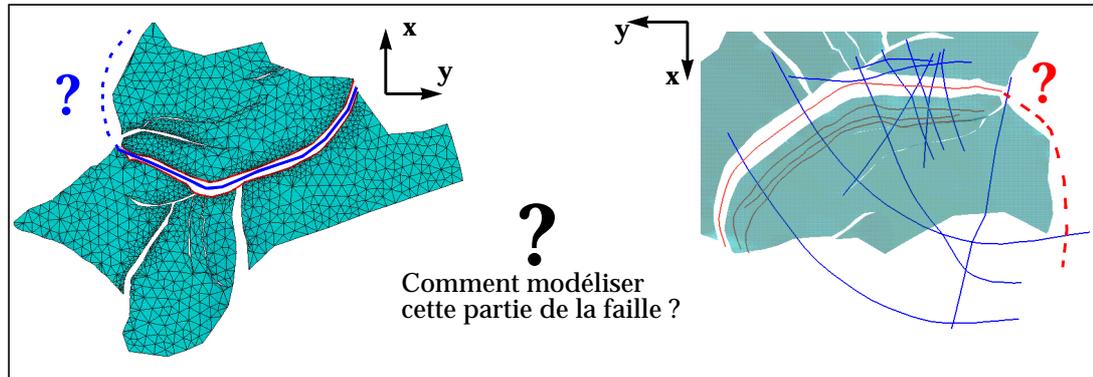


**Figure 5-18 :** Bien que la faille obtenue de manière automatique corresponde aux données, les connaissances du géologues sont toujours nécessaires afin de construire un réseau cohérent.

## 2.2 Edition “à la main”

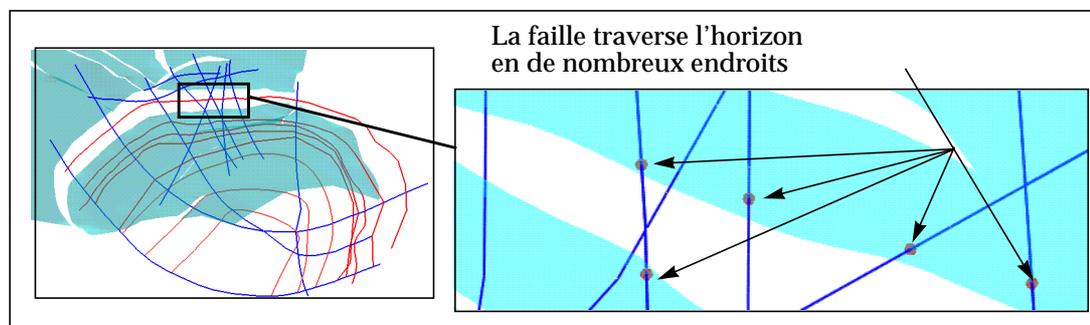
Une fois la faille de base construite à partir des lignes médianes, il est possible de l'éditer assez facilement afin de diminuer les incohérences entre les différents types de données. Ces incohérences peuvent être dues soit au pointé des horizons imprécis, soit aux interprétations de failles non valides du point de vue géologique.

L'un des jeux de données qui m'ont été fournis contient une faille listrique au niveau de laquelle les interprétations des horizons n'ont pas été poussées assez loin, et qui rendent difficile la modélisation de la faille.



**Figure 5-19 :** Lorsque l'interprétation de l'horizon n'a pas été poussé assez loin, il faut une information supplémentaire pour modéliser la faille.

Des interprétations de failles sont aussi disponibles sur ce jeu de données, mais il existe des incohérences entre l'interprétation des horizons et les interprétations de failles. En effet, ces dernières traversent les horizons à plusieurs endroits. De plus, dans la zone où l'horizon ne permet pas de modéliser la faille, il n'y a pas de courbe interprétée pour guider la modélisation [figure 5-20].



**Figure 5-20 :** Les données présentent des incohérences: les interprétations de failles traversent les horizons [b]. A quelles données doit-on faire confiance?

Ce cas est relativement courant: la modélisation en trois dimensions arrive en fin de chaîne, et les données interprétées par différentes équipes (horizons d'une part et interprétations de failles d'autre part) ne sont pas forcément confrontées avant de construire le modèle profondeur en trois dimensions.

Dans un tel cas, une fois encore, seul le géologue peut valider ou non le modèle obtenu, faire des choix entre les différents types de données.

Nous espérons seulement que l'objet faille et ses fonctions d'édition rapide peuvent aider à simplifier cette tâche, en permettant de construire une première faille uniquement à partir des données d'horizons, puis en éditant localement les zones posant problème. Mais il ne faut pas oublier l'apport d'un type de données pour l'instant irremplaçable: les connaissances géologiques.

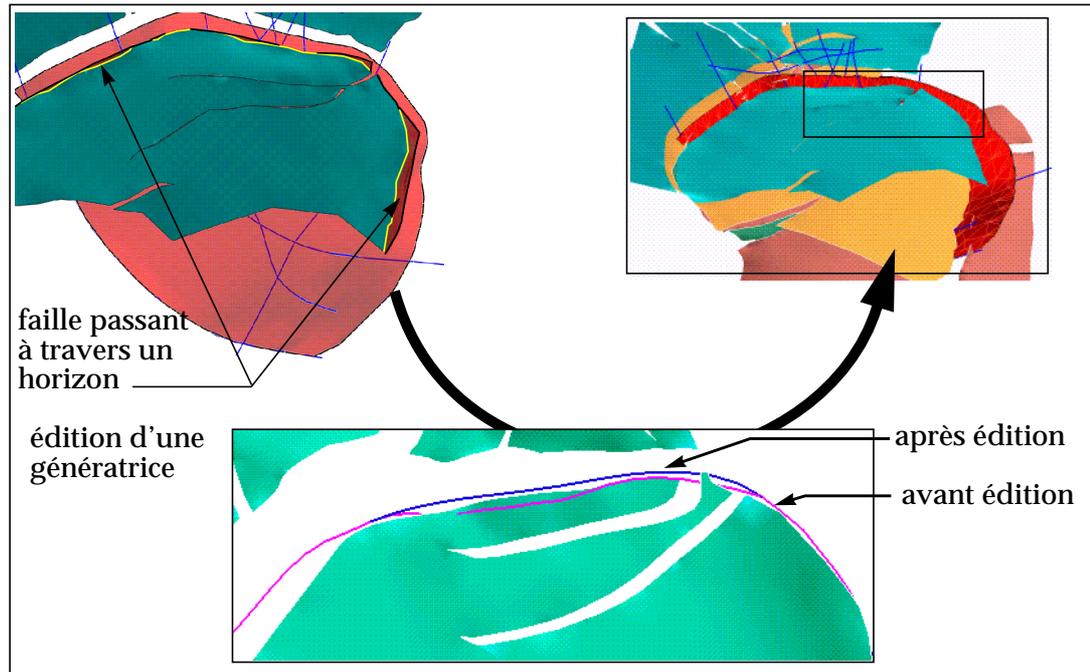


Figure 5-21 : L'édition des courbes permet de diminuer les incohérences ou de pallier le manque d'informations.

### 2.3 Connexions avec d'autres surface

La dernière fonction importante lors de la modélisation de la faille est de pouvoir connecter rapidement une faille secondaire sur sa faille principale, ou de connecter une faille aux bornes du modèle. La méthode est très simple et rapide.

La première étape consiste à détecter quel bord de la faille secondaire doit être étendu. Le point de la faille principale le plus proche de l'extrémité du backbone portant ce bord est calculé. Un atome correspondant à ce point d'impact est construit et ajouté au backbone. La dernière étape consiste à projeter la génératrice correspondant au bord sur la surface de la faille principale et de l'affecter comme génératrice pour l'atome qui vient d'être créé.

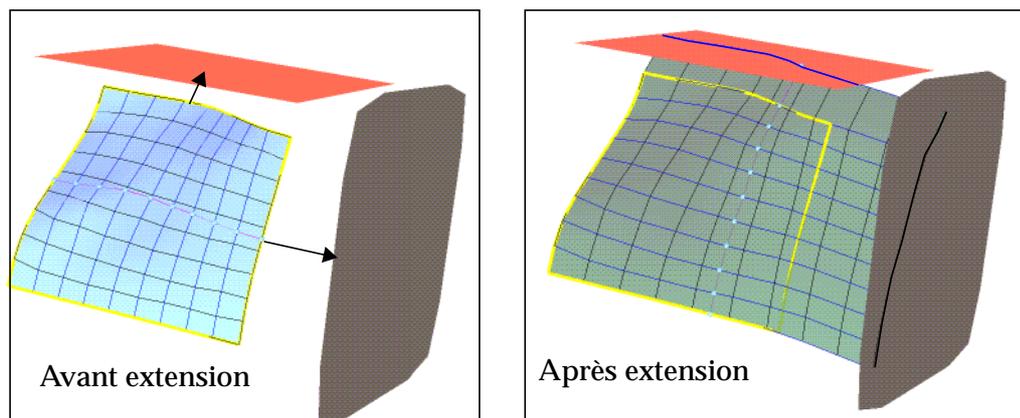


Figure 5-22 : Extension de l'objet faille vers une faille ou un horizon.

**Remarque:** la surface sur laquelle la génératrice extrême est projetée n'a pas besoin d'être plane, comme pourraient le laisser supposer ces images.

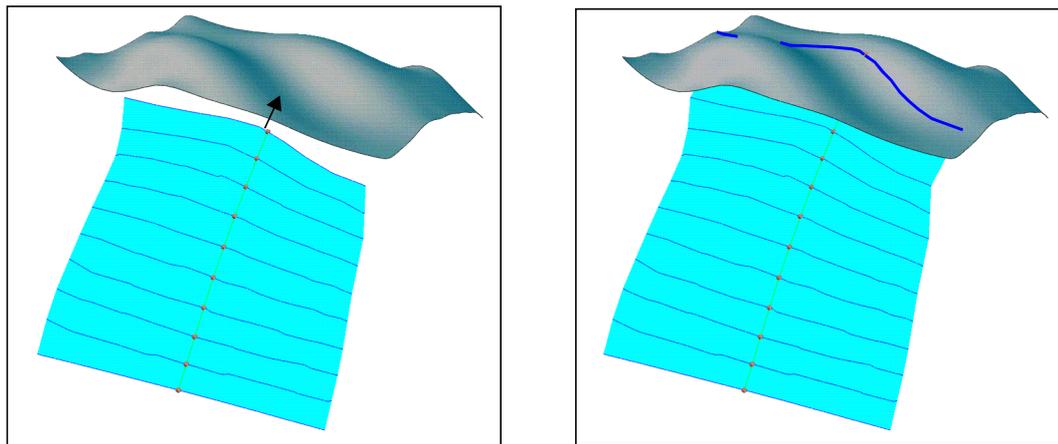


Figure 5-23 : Exemple d'extension lorsque l'horizon n'est pas plan.

### 3. Conclusion sur l'objet Faille

Cet objet peut donc servir à construire rapidement des modèles de faille, *sélectionner les données valides* et éliminer les données incohérentes, lors de la modélisation des failles en domaine profondeur.

On peut même imaginer que cet objet soit utilisé dans le domaine temps afin de permettre à l'interpréteur de *vérifier son interprétation des données sismiques* directement en trois dimensions. Pour cela, il faut une meilleure communication entre les stations d'interprétation et gOcad. Cet aspect a été abordé dans le troisième chapitre [cf [Partie1 - chapitre 3](#)].

Comme nous l'avons dit, cette première utilisation de l'objet faille, n'est aucunement liée aux études d'incertitudes.

Ceci dit, notre propos est de pouvoir introduire une notion d'incertitude à cet objet et de pouvoir simuler différentes positions et/ou formes acceptables du point de vue géologique autour du modèle de référence.

Le chapitre suivant décrit plus précisément cet aspect de l'objet faille, ainsi que les méthodes de simulations. Dans la suite, nous considérons que le modèle de référence validé par le structuraliste m'est fourni, soit sous forme de surface triangulée, soit directement sous forme d'objet faille.

---

---

## Chapitre 6 Estimation des incertitudes et Méthodes de simulation

---

---

Comme cela a été montré dans le chapitre 3, la modélisation des failles en domaine profondeur arrive *en bout de chaîne*. Les données ont été traitées par des personnes de différents services et à chaque étape, des incertitudes de natures et d'importances diverses se sont accumulées, sans que les différentes équipes n'aient l'historique complet des choix effectués [64 - Basire].

Une telle incertitude sur la forme et la position des failles peut avoir des conséquences non négligeables sur l'estimation des volumes du réservoir. Un petit changement de pendage sur une faille bordière peut avoir un impact considérable sur l'estimation de la quantité d'hydrocarbures que l'on peut extraire du sous-sol, et donc sur la prévision des installations nécessaires ([figure 6-1] a).

Il peut même arriver qu'une erreur d'interprétation sur la position de la faille ait des conséquences catastrophique en phase de production. En effet, la plupart du temps, les implantations de puits se font le *plus loin possible des failles*, afin d'éviter ce genre de déconvenues. Mais les réservoirs auxquels les compagnies pétrolières s'intéressent désormais se trouvent dans des zones de plus en plus complexes et sont de plus en plus petits. Il arrive donc, qu'afin de récupérer le maximum d'hydrocarbures, on se place plus à proximité des failles. Si la faille a été mal positionnée, suite à une erreur de migration, d'une mauvaise sismique ou d'une erreur d'interprétation, le puits peut, dans certains cas extrêmes, passer complètement en dehors du réservoir recherché ([figure 6-1] b).

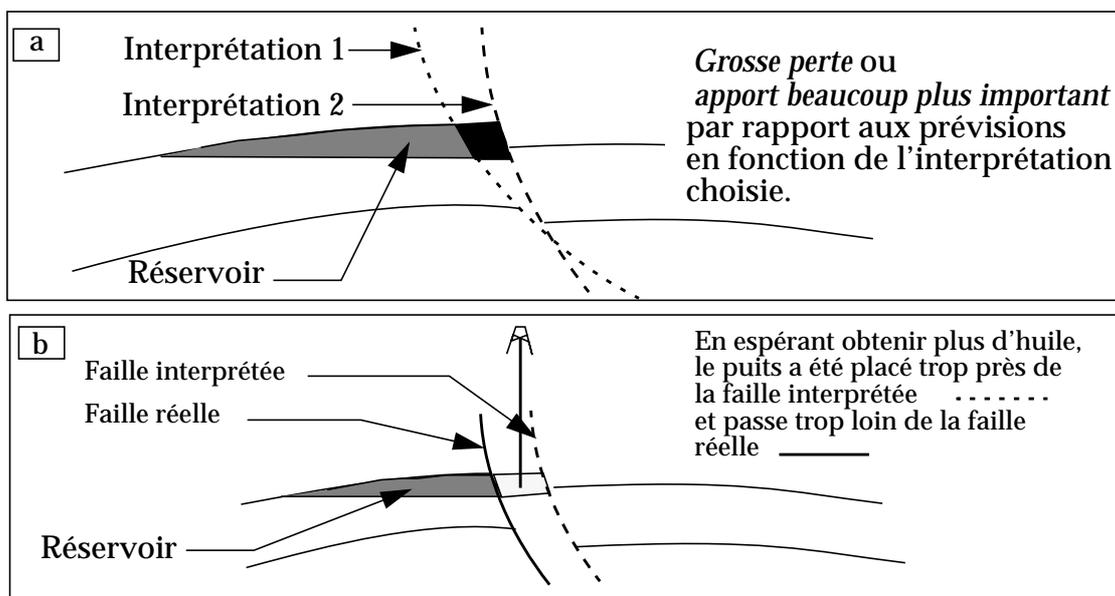


Figure 6-1 : Deux cas où une mauvaise interprétation peut avoir de graves conséquences lors de la production: [a]: mauvaise interprétation du pendage; [b]: mauvaise interprétation de la position réelle de la faille.

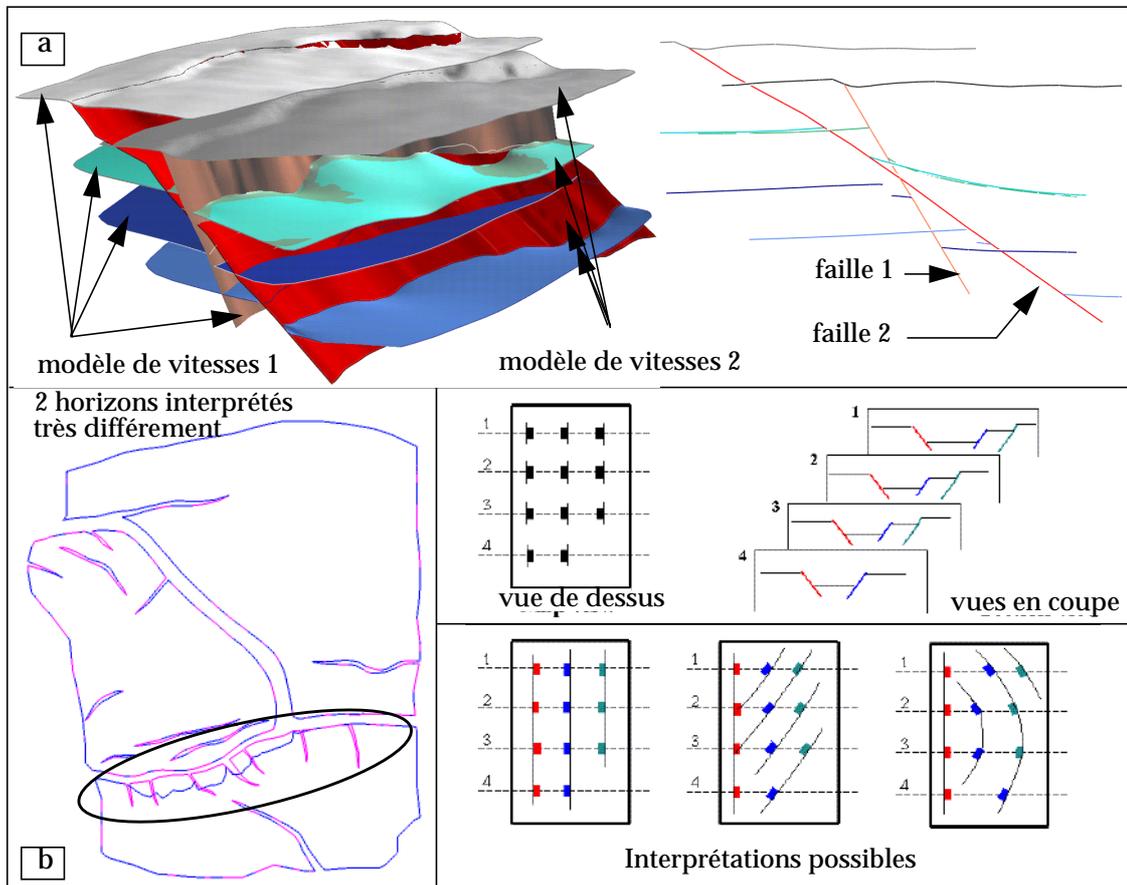
Parmi toutes les incertitudes répertoriées, nous n'avons pris en compte qu'une partie d'entre elles, les autres étant déjà étudiées par d'autres personnes ([67 - Juliard], [70 - Thore 1]), ou devant être étudiées par la suite. Nous avons ainsi distingué:

- les incertitudes dues à une mauvaise estimation du modèle des **vitesse de migration** ([figure 6-2] a). Nous laissons de côté ces incertitudes, c'est à dire que nous travaillons à *modèle de vitesses constant*

- les incertitudes dues aux **problèmes d'interprétation** le long du contact horizon-faille, surtout dans les zones de branchement

**Ce sont ces incertitudes que nous avons prises en compte.** Afin de simplifier le travail, nous avons toutefois limité ces incertitudes en supposant que l'interprétation d'un horizon à l'autre était cohérente (ce qui n'est pas toujours le cas) ([figure 6-2] b).

- les incertitudes dues aux problèmes de **corrélation des segments** piqués en deux dimensions dans l'espace à 3 dimensions ([figure 6-2] c). Ce type d'incertitudes est aussi laissé de côté dans notre travail: nous avons travaillé à *scénario de réseau de failles constant*. Ce dernier point sera repris plus largement dans le chapitre 9 de ce mémoire.



**Figure 6-2 :** Parmi toutes les incertitudes affectant les failles, nous n'avons pas pris en compte: [a]: les incertitudes dues aux vitesses de migration; [b]: les incertitudes dues à une interprétation incohérente d'un horizon à l'autre; [c]: les incertitudes dues au choix du scénario pour le réseau de failles [121 - Freeman].

Le but de ce chapitre est de montrer comment l'objet faille décrit dans le chapitre précédent peut non seulement prendre en compte ces incertitudes, mais aussi permettre de simuler différentes formes plausibles pour la faille à l'intérieur de cette zone incertaine. En effet, lorsqu'un modèle unique ne peut représenter l'incertitude sur la géométrie de la faille, une méthode consiste à construire un grand nombre de modèles possibles à l'aide de techniques géostatistiques [87 - Bagtzoglou].

## 1. Estimation des incertitudes

Jusqu'à présent, les estimations de l'incertitude correspondant à l'interprétation des données sismiques ne sont *pas fournies*. Bien que ne disposant ni des données sismiques, ni des différents choix effectués durant l'interprétation, nous proposons ici les méthodes *très simples* utilisées jusqu'à présent afin de poursuivre le travail.

### 1.1 Le problème

Les incertitudes dues à la phase d'interprétation sont essentiellement de deux types selon lesquels l'estimation de l'incertitude se fera différemment:

1. une absence de données ([figure 6-3]):

C'est le cas que j'ai rencontré le plus souvent sur les jeux de données fournis par Elf Exploration Production, et qui correspondent pour la plupart à des données "nettoyées", et où le pointé s'arrête dès que la sismique devient trop floue. Les polygones de failles sont dessinés par le structuraliste dans l'intervalle correspondant à la faille.

L'exemple ci-dessous montre qu'il existe parfois une différence énorme entre le dessin du structuraliste et l'arrêt du pointé réalisé par l'interpréteur, notamment au niveau des fourches.

A priori, la confiance que l'on peut accorder au polygone de faille est plus grande, mais cela reste une interprétation. Lorsque le pointé sur la sismique et le polygone de failles diffèrent beaucoup, on peut supposer que l'incertitude est encore grande à ce niveau, sauf si la zone où les deux interprétations divergent est relativement restreinte, et qu'ailleurs, on obtient une bonne corrélation.

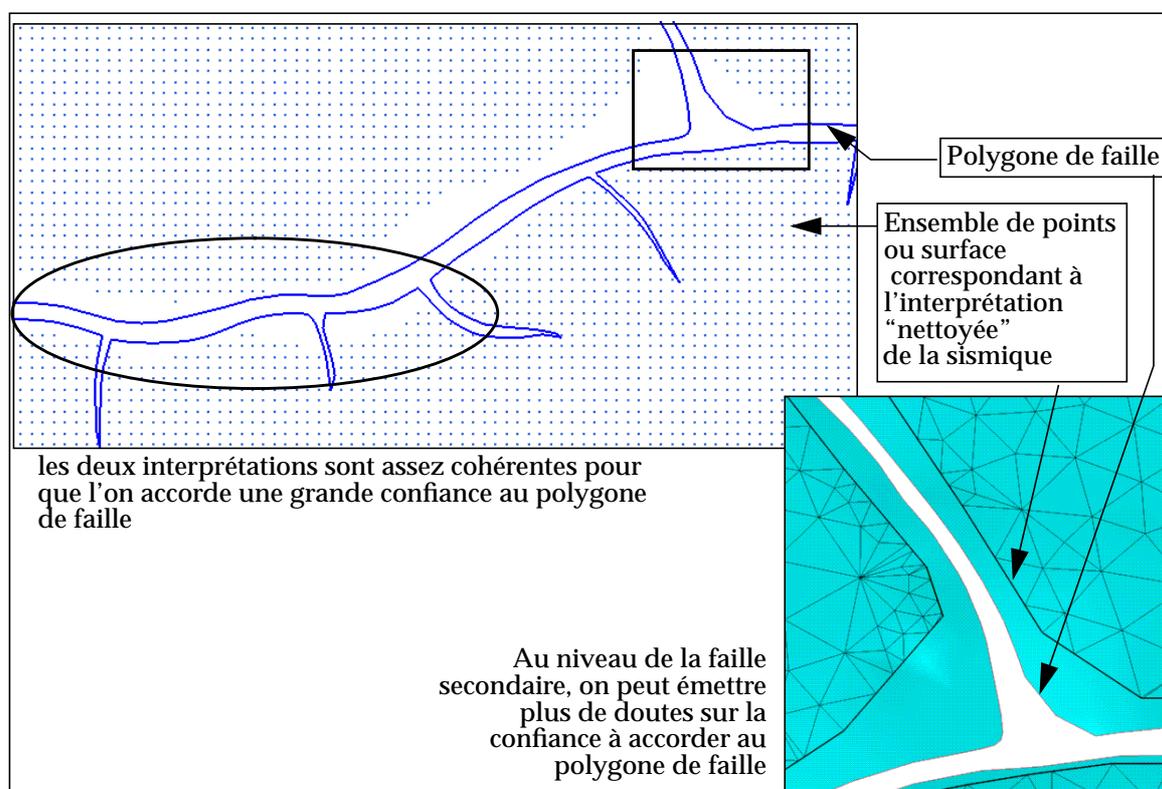
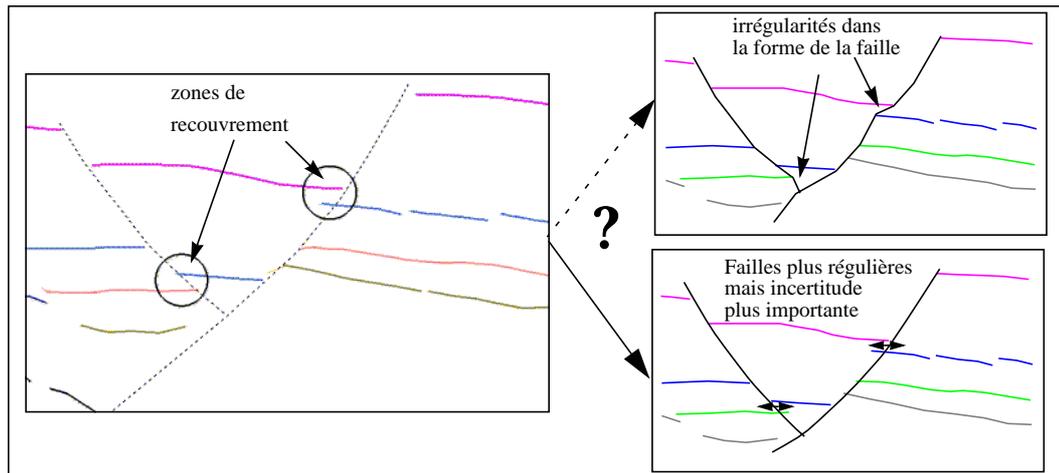


Figure 6-3 : La confiance à accorder au polygone de faille doit être modulée en fonction de la corrélation avec les données sismiques. Lorsque les deux interprétations sont concordantes, l'incertitudes est faible, mais s'il manque des données sismiques, il est beaucoup plus difficile de prendre une décision.

## 2. des recouvrements

Au contraire, lorsque le pointé a été poussé trop loin, ou qu'un décalage existe au niveau de l'un des horizons situé en profondeur, on obtient des zones de recouvrement. Si l'on construit une faille passant exactement par les bords des horizons, on obtient une forme qui présente de fortes irrégularités localement. Ceci peut éventuellement être acceptable si la lithologie correspond à une couche plus dure à ce niveau. Mais n'ayant aucune information sur la lithologie, j'ai considéré ces irrégularités comme étant le résultat d'une interprétation poussée trop loin.

Dans ce cas, il faut prendre une décision et adapter la forme de la faille afin d'obtenir une forme acceptable du point de vue géologique, mais ceci augmente encore l'incertitude (devant l'infinité de choix possibles, a-t-on fait le bon?).



**Figure 6-4 :** Lorsqu'il existe des zones de recouvrement entre deux horizons successifs, le choix de la forme de la faille est laissé à celui qui la construit. Bien que ce choix s'effectue en fonction de règles structurales, une forte incertitude peut persister.

### 1.2 Estimation de l'incertitude à l'aide de formules simples

Les incertitudes sont représentées par un couple de vecteurs (un de chaque côté de la faille) pour chacun des points de la faille. Ce paragraphe décrit une méthode simple qui permet de calculer ces vecteurs d'incertitude en comparant les bords de l'horizon pointé et le polygone de faille.

Pour cela, un certain nombre de propriétés sont calculées et stockées sur la ligne médiane. Pour chacune d'elles, on distinguera la propriété *G* correspondant à la partie "gauche" de la faille, et la propriété *D* correspondant à la partie "droite" de la faille. Le côté gauche est arbitrairement orienté dans le même sens que le troisième vecteur de la base locale. Jusqu'à présent, cela correspond au côté concave de la faille.

1. *G1* et *D1*: propriétés vectorielles qui correspondent aux vecteurs associant un point du polygone de faille et le point correspondant sur le bord de l'horizon. Les correspondances sont recherchées perpendiculairement à la faille.
2. *G2* et *D2*: propriétés vectorielles qui correspondent aux vecteurs entre un point de la ligne médiane entre les deux bords du polygone de faille, et le point correspondant sur le bord de l'horizon. Les correspondances sont aussi recherchées perpendiculairement à la faille.
3. *G3* et *D3*: propriétés scalaires représentant la longueur des vecteurs *G1* et *D1* et sur lesquelles des statistiques peuvent être appliquées. Plus l'histogramme calculé à partir de l'une de ces propriétés est resserré autour de la valeur zéro, et plus on peut accorder une grande confiance au polygone de faille.

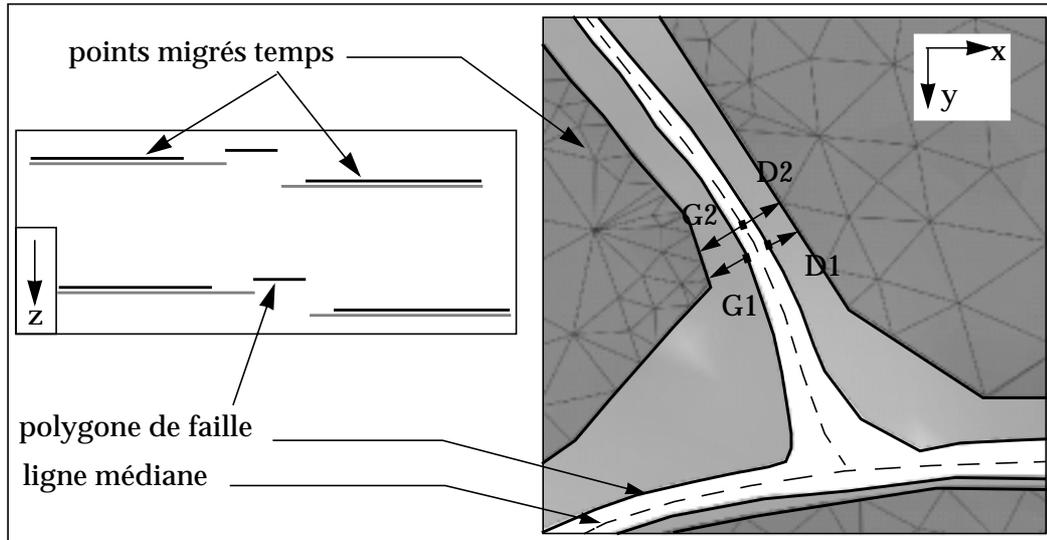


Figure 6-5 : Calcul des différentes propriétés utilisés pour construire les vecteurs d'incertitude

Afin d'obtenir des vecteurs assez réguliers, nous proposons deux méthodes qui combinent les propriétés précédentes, après avoir calculé une valeur globale pour chaque côté de la faille. Cette valeur peut correspondre à la valeur moyenne de la propriété, majorée par un nombre d'écart types choisi par l'utilisateur afin de tenir compte des plus grandes incertitudes, ou de manière plus simple être égale à cette incertitude maximale.

$$g1 = \overline{G1} + n \cdot \sigma_{G1} \quad \text{ou} \quad g1 = G1_{\max}$$

$$g2 = \overline{G2} + n \cdot \sigma_{G2} \quad \text{ou} \quad g2 = G2_{\max}$$

Ensuite, deux vecteurs d'incertitude sont calculés pour chacun des points (un pour le côté "gauche" et un pour le côté "droit"). Là encore, deux méthodes sont proposées:

- le vecteur d'incertitude correspond à la *valeur globale* calculée plus haut.
- le vecteur d'incertitude est calculé comme une combinaison linéaire de la valeur précédente, et de la propriété G2 (resp. D2). Ceci permet d'associer une plus *grosse incertitude* dans les zones où la ligne médiane est loin des bords des points interprétés sur la sismique (typiquement: les *zones de branchement*), mais a pour gros inconvénient le choix des coefficients  $\alpha$  et  $\beta$  [figure 6-6].

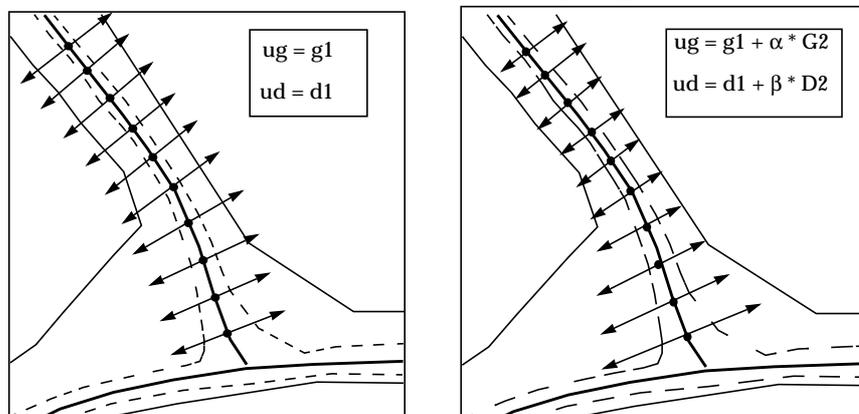


Figure 6-6 : Les deux méthodes empiriques proposées pour calculer les vecteurs d'incertitude.

Le problème est encore plus difficile à résoudre lorsque l'incertitude correspond à une zone de recouvrement [figure 6-7].

Si la zone de recouvrement se produit entre les deux bords d'un *même horizon*, on peut proposer comme incertitude la distance entre la ligne médiane et entre les bords correspondants sur l'horizon.

Par contre, si le recouvrement apparaît entre *deux horizons différents*, il est beaucoup plus difficile de calculer une estimation de l'incertitude à ce niveau. Pour le vecteur qui se trouve du côté de la zone de recouvrement, une méthode est proposée, mais de l'autre côté il est plus difficile de proposer un calcul de l'incertitude.

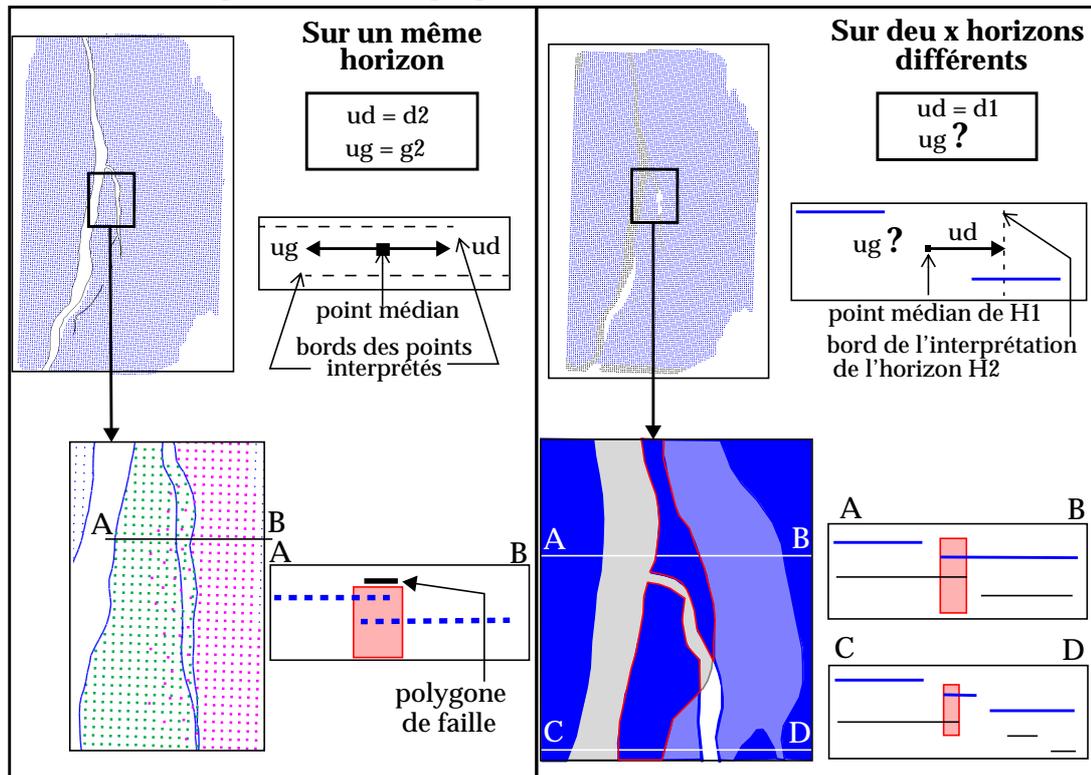


Figure 6-7 : Lorsque l'incertitude est due à des zones de recouvrement, son estimation est plus difficile, surtout si le recouvrement se produit entre deux horizons différents.

Comme on le voit, les formules présentées précédemment n'ont aucun fondement géologique et ne sont proposées que pour pallier l'absence d'estimation de l'incertitude. Dans la pratique, elles n'ont pas encore été utilisées.

### 1.3 Solution idéale: estimations d'incertitudes fournies par le géophysicien

Considérant que l'incertitude due au pointé est fortement liée aux données sismique, nous avons décidé que **cette partie n'était pas de notre ressort**, mais devait être *fournie par les interpréteurs* [70 - Thore 1], [90 - Froidevaux]. Nous espérons en effet que l'estimation de l'incertitude fasse partie de l'étude à l'avenir, et nous avons préféré concentrer nos efforts sur l'utilisation de ces estimations, plutôt que de travailler sur des formulations qui ne correspondront pas à la plupart des données.

Des cartes qualitatives représentant la confiance qui peut être accordée aux données sismiques sont disponibles sur certains jeux de données. En l'absence de précisions supplémentaires, elles sont utilisées actuellement pour estimer une valeur d'*incertitude globale* pour chacune des failles [figure 6-8].

Cette solution simple permet de tester notre travail, mais n'utilise qu'une petite partie des potentialités de la gestion des vecteurs d'incertitude avec l'objet faille.

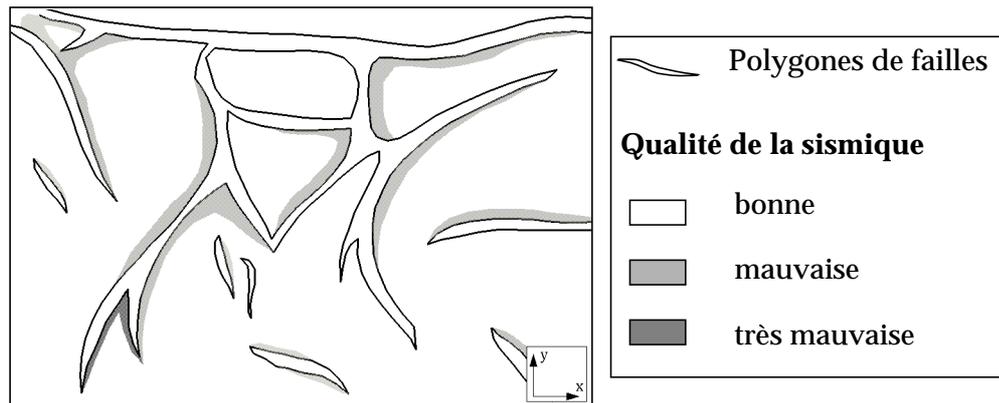


Figure 6-8 : Exemple de carte donnant une estimation de la qualité des données sismiques (fournie de temps en temps par l'interpréteur).

## 2. Comment tenir compte des estimations de l'incertitude

La paramétrisation simple de l'objet faille permet de stocker en chacun des points une estimation vectorielle de l'incertitude [94 - Lecour 3]. Comme l'estimation de l'incertitude est un travail supplémentaire demandé à l'interpréteur, on ne peut demander une estimation pour chacun des points de la faille.

Cette section montre comment les structures mises en place pour gérer ces vecteurs d'incertitude permettent d'affecter à chaque point un couple de vecteurs (ug, ud), à partir de quelques valeurs fournies en entrée. Le deuxième paragraphe est un peu plus technique et peut éventuellement être laissé de côté

### 2.1 Principe

Comme cela a été montré lors du chapitre précédent, l'objet faille peut très bien être utilisé comme constructeur rapide de faille, sans que l'aspect "incertitudes" ne soit abordé. Le stockage des vecteurs d'incertitude n'est donc pas un élément nécessaire à la construction et au bon fonctionnement de l'objet faille, mais plutôt une extension des possibilités de cet objet.

De plus, dès que la topologie de l'objet faille est modifiée (ajout d'un atome sur la courbe principale, ou modification du nombre de points par génératrices), il faut être capable de remettre à jour automatiquement les vecteurs d'incertitude.

Cette fois encore, cela correspond très bien à la création d'une **association** entre la faille et un gestionnaire de vecteurs d'incertitudes.

Le principe du fonctionnement des associations a été présenté dans le chapitre 4. Seuls quelques rappels sont fait ici:

- les variables stockées dans l'association ne sont créées que lorsque le besoin s'en fait sentir
- l'association se contente d'observer l'objet auquel elle est liée. Dès que l'objet est modifié, elle note qu'il y a eu un changement (elle est marquée comme étant "dirty"). Dans le même temps, les variables erronées sont détruites sans lancer le calcul correspondant au nouvel état de l'objet
- ce n'est que lorsque une classe a besoin d'utiliser les variables stockées dans l'association (par exemple lorsque l'on veut dessiner les vecteurs d'incertitude à l'écran), que le calcul est lancé pour tenir compte de la modification.

Ce système est un peu plus lourd à gérer, mais permet de stocker un minimum de choses entre deux opérations faisant appel à cette association.

## 2.2 L'association FaultUncertainty

Afin de mieux comprendre certains des schémas présentés dans la suite de ce chapitre, voici quelques conventions qui seront utiles:

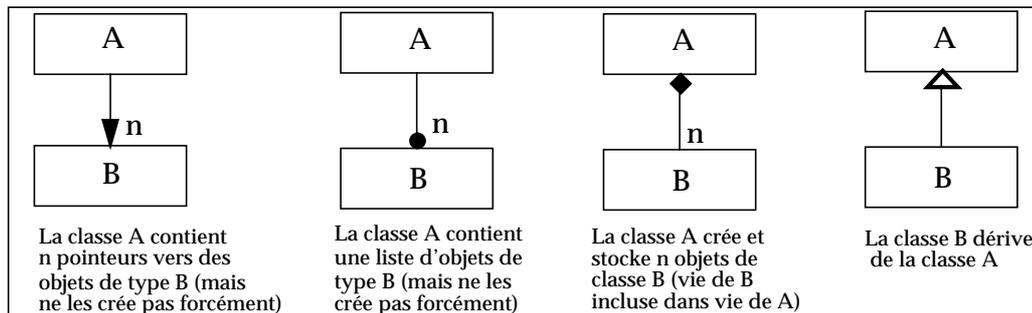


Figure 6-9 : Conventions pour décrire les diagrammes de classes

### 2.2.1 Stockage des estimations

Une notion importante dans la gestion des vecteurs d'incertitude est la séparation entre les vecteurs estimés par l'interpréteur (données) et les vecteurs interpolés à partir des données.

Le diagramme des classes intervenant dans la gestion des vecteurs d'incertitude est assez simple [figure 6-12].

On distingue les classes suivantes:

- **CONSTRAINEDVALUE**

Cette classe très simple stocke uniquement un entier (*value\_*). Cet entier permet de savoir si la donnée correspond à l'un des vecteurs uniquement (gauche ou droite), ou si les deux vecteurs sont fournis en entrée.

- **UNCERTAINTYVALUE**

Par définition cette classe contient un couple de vecteurs (*left*, *right*) et une variable de type **CONSTRAINEDVALUE** qui permet de savoir si la donnée concerne le vecteur gauche, le vecteur droite ou les deux vecteurs. Cette classe est très générale et pourrait éventuellement être utilisée pour un autre objet que l'objet faille (adaptée, elle pourrait stocker l'incertitude sur les horizons). Les principales fonctionnalités de cette classe sont de retourner le vecteur d'incertitude droite (fonction *right()*) ou gauche (fonction *left()*). Cette classe permet en outre de calculer les vecteurs d'un côté à partir des vecteurs de l'autre côté par symétrie, si une seule des deux informations n'est fournie.

- **GFAULTUNCERTAINTYVALUE**

Cette classe dérive de la précédente et concerne la gestion d'un couple de vecteurs d'incertitude associé à un objet faille. En plus de ces deux vecteurs, elle stocke la position origine des deux vecteurs dans l'espace réel (xyz) ainsi que son abscisse curviligne *s\_* le long de la génératrice.

L'origine doit obligatoirement se trouver sur une des génératrices de la faille, mais pas obligatoirement sur l'un des points de cette génératrice.

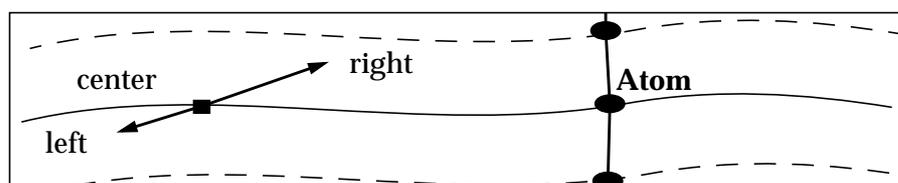


Figure 6-10 : Éléments stockés dans une GFaultUncertaintyValue

Enfin, avant de stocker les deux vecteurs d'incertitude, ceux-ci sont transformés dans la *base locale* correspondant à la génératrice sur laquelle ils se rattachent (d'où l'obligation d'avoir l'origine sur une génératrice en particulier).

L'intérêt d'une telle transformation se voit beaucoup mieux sur une faille listrique. Supposons que deux vecteurs soient fournis uniquement, à chacune des extrémités, dirigés selon l'axe *Ox* mais avec des directions opposées. L'interpolation linéaire donnera un vecteur de longueur nulle au milieu de la génératrice. Au contraire, s'ils sont stockés dans une base locale, l'interpolation donnera le vecteur "naturel" localement perpendiculaire à la génératrice.

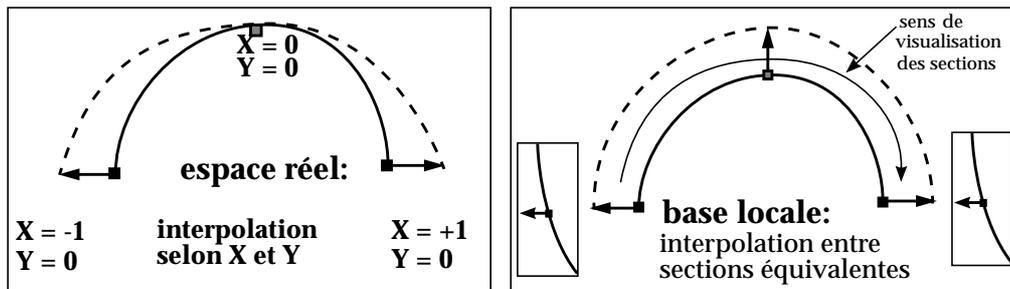


Figure 6-11 : Pourquoi les incertitudes sont stockées dans la base locale.

• **GFAULTUNCERTAINTIES**

Cette classe associe à chaque atome une liste de **GFAULTUNCERTAINTYVALUES**. L'ajout ou la destruction d'une estimation d'incertitude est gérée par cette classe, lorsqu'on lui fournit l'atome et l'origine correspondants. Elle permet en outre de retrouver une incertitude donnée en spécifiant l'atome contenant la génératrice sur laquelle elle se rattache et l'origine des vecteurs qui la composent.

CLASSE	VARIABLES	FONCTIONS
	<p>value_</p> <ul style="list-style-type: none"> <li>0: aucune</li> <li>1: left</li> <li>2: right</li> <li>3: les 2</li> </ul>	
	<p>Vector3d left_ Vector3d right_ ConstrainedValue cv_</p>	<p>left() right() set_left() set_right() unset_left() unset_right() symeterize()</p>
	<p>Point3d position_ float s_</p>	
	<p>Table&lt;Atom, GFUV&gt; atom2uncertainties_</p>	<p>add_u() remove_u() find_u()</p>

Figure 6-12 : Classes intervenant dans la gestion des vecteurs d'incertitude.

## 2.2.2 Mécanisme de mise à jour

### a) Description sommaire de la méthode choisie

Dès que l'on ajoute un vecteur d'incertitudes, une fonction de mise à jour est appelée afin d'incorporer la nouvelle valeur dans le processus d'interpolation des vecteurs d'incertitudes.

Après avoir nettoyé le stockage de tous les vecteurs qui ne correspondent pas à des données, cet algorithme se déroule en trois grandes étapes [figure 6-13]:

#### 1. Interpolation pour les génératrices qui portent des données

Cette étape consiste à affecter des valeurs aux points des génératrices portant au moins un vecteur de type "donnée", soit par propagation vers les extrémités, soit par interpolation linéaire pour les points du centre.

Si la génératrice ne stocke qu'une seule estimation d'incertitude, celle-ci sera affectée à tous les points de la génératrice. Par contre, si la génératrice stocke plus d'une estimation de l'incertitude, cette première étape se découpe elle-même en trois phases:

- a. recherche du premier point ( $p1$ ) portant une estimation d'incertitude ( $u1$ ) à partir de l'une des extrémités ( $e1$ ). Affectation de l'incertitude  $u1$  à tous les points situés entre  $e1$  et  $p1$ .
- b. recherche du premier point ( $p2$ ) portant une estimation d'incertitude ( $u2$ ) à partir de l'autre extrémité ( $e2$ ). Affectation de l'incertitude  $u2$  à tous les points situés entre  $e2$  et  $p2$ .
- c. interpolation linéaire entre  $u1$  et  $u2$  pour tous les points situés entre les points  $p1$  et  $p2$ .

#### 2. Affectation des valeurs sur les génératrices extrêmes

Cette deuxième étape utilise le même principe que les points a. et b. précédents, mais appliqués plus globalement aux génératrices extrêmes, et non plus aux points extrêmes d'une génératrice particulière.

Elle commence par la recherche de la première génératrice ( $G1$ ) portant au moins une incertitude de type donnée à partir de l'extrémité du backbone  $E1$ , puis copie des incertitudes stockées sur  $G1$  sur toutes les génératrices situées entre  $E1$  et  $G1$ .

La même opération est ensuite effectuée en partant de l'autre extrémité ( $E2$ ) du backbone. On recherche la première génératrice ( $G2$ ) portant une estimation de l'incertitude à partir de l'extrémité  $E2$  et on copie ses vecteurs d'incertitude sur les génératrices situées entre  $E2$  et  $G2$ .

#### 3. Interpolation pour les points qui n'ont pas encore été pris en compte

Pour tous les autres points, s'il en reste, une interpolation linéaire est calculée à partir des vecteurs de données, et des vecteurs calculés lors des étapes précédentes.

Cette méthode très simple permet ainsi de ne stocker que les estimations d'incertitudes fournies par les géophysiciens, tout en étant capable de calculer instantanément un vecteur pour chacun des points de la faille.

Elle permet aussi de mettre à jour ces calculs, dès que les vecteurs de types données sont modifiés (ajout d'un vecteur, modification d'un vecteur ou retrait d'un vecteur considéré comme aberrant pas exemple).

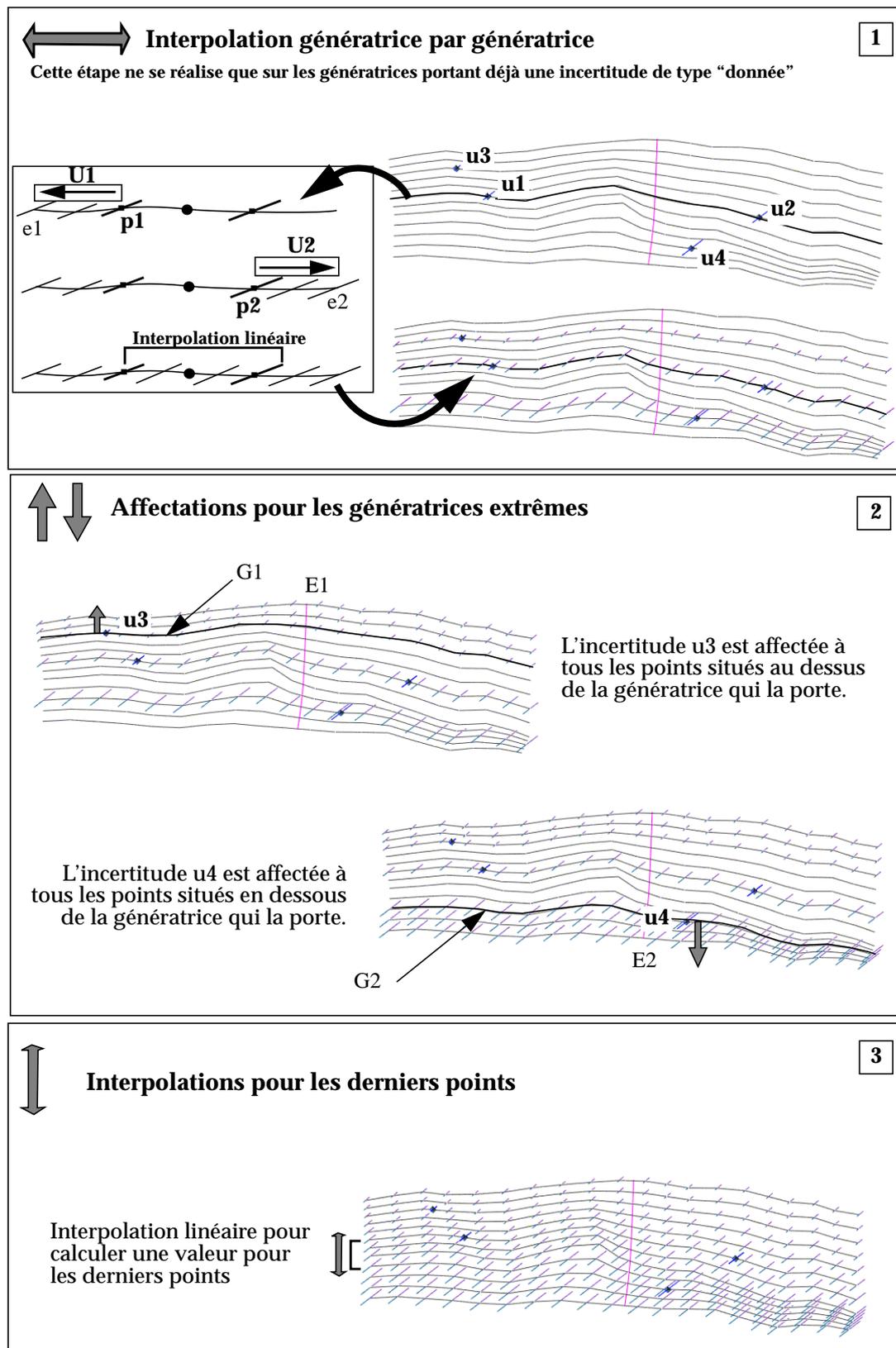


Figure 6-13 : Les trois grandes étapes intervenant dans la mise à jour des vecteurs d'incertitude: [a]: interpolation pour les génératrices portant des données; [b]: affectation des valeurs ainsi interpolées aux génératrices extrêmes; [c]: interpolation linéaire pour affecter l'incertitude au niveau des derniers points.

### b) Pourquoi ne pas faire une “vraie” extrapolation?

On peut reprocher à la méthode précédente le fait qu'on ne réalise pas une vraie extrapolation des incertitudes aux extrémités, mais que l'on affecte la dernière valeur de type donnée à tous les points extrêmes. Ceci déforme un peu le résultat, par rapport à celui que l'on aurait obtenu en suivant la tendance amorcée loin des extrémités.

On aurait éventuellement pu appliquer une méthode d'extrapolation, mais cela aurait entraîné un trop grand risque d'obtenir des inversions dans les vecteurs ainsi extrapolés, lorsqu'une interpolation entre deux vecteurs trop différents doit être effectuée. C'est pourquoi nous nous en sommes tenus à cette approximation, qui jusqu'à présent est amplement suffisante.

En effet, s'il l'on revient à la signification de ces vecteurs, on se rend compte qu'ils ne représentent qu'une estimation de l'incertitude sur une donnée mal connue: la géométrie de la faille. Cette incertitude est tellement mal définie actuellement qu'une seule valeur d'incertitude globale est utilisée pour faire les premiers tests. Il ne m'a pas paru utile, dans l'état actuel des choses, de mettre au point une méthodologie plus complexe pour réaliser les interpolations, alors que la méthode actuelle est amplement suffisante (puisque'elle prend déjà en compte des cas plus complexes que les estimations qui peuvent être fournies actuellement).

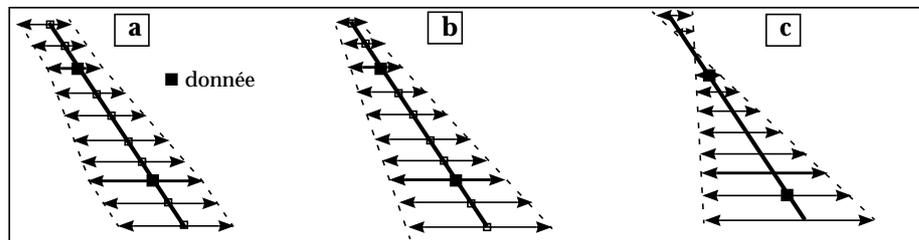


Figure 6-14 : Pourquoi la méthode de mise à jour n'utilise pas une vraie approximation. [a]: méthode utilisée; [b]: extrapolation dans un cas où tout se passe bien; [c]: extrapolation avec inversion des directions.

### c) Mise au point d'un mécanisme de contrôle de la cohérence des estimations

Une fois que les vecteurs d'incertitude seront estimés de manière plus fine, il faudra imaginer un mécanisme de vérification de la cohérence des données initiale.

Actuellement, nous supposons que les incertitudes étant difficiles à estimer, elles varient peu le long de la faille. On peut imaginer que dans quelques années, cette estimation des incertitudes ne sera plus un problème, et que des différences plus grandes pourront apparaître (par exemple une incertitude plus faible près d'un puits où la faille a été repérée, et une incertitude plus forte loin de ce puits).

Il est évident que si la forme de la faille est estimée de façon sûre en un point donné, il n'y a pas une énorme incertitude au niveau du point adjacent. Actuellement, si la cas se produit, les méthodes de simulation balayent toute la zone d'incertitude fournie, sans se poser de question. Lorsque ces estimations seront plus précises, il faudra absolument un mécanisme de contrôle de la cohérence des vecteurs d'incertitude.

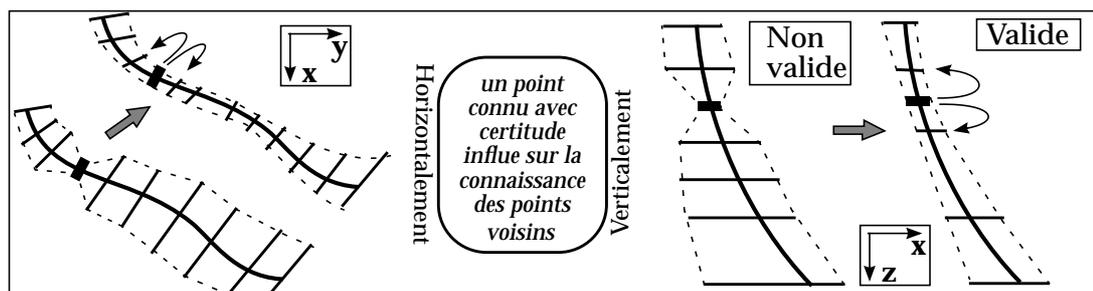


Figure 6-15 : Vérification de la cohérence des estimations d'incertitude

## 2.3 Exemple d'utilisation de ces mécanismes sur des données réelles

Cette section montre de manière plus pratique comment les incertitudes estimées sur la sismique ou à partir de formules simples peuvent être stockées sur l'objet faille.

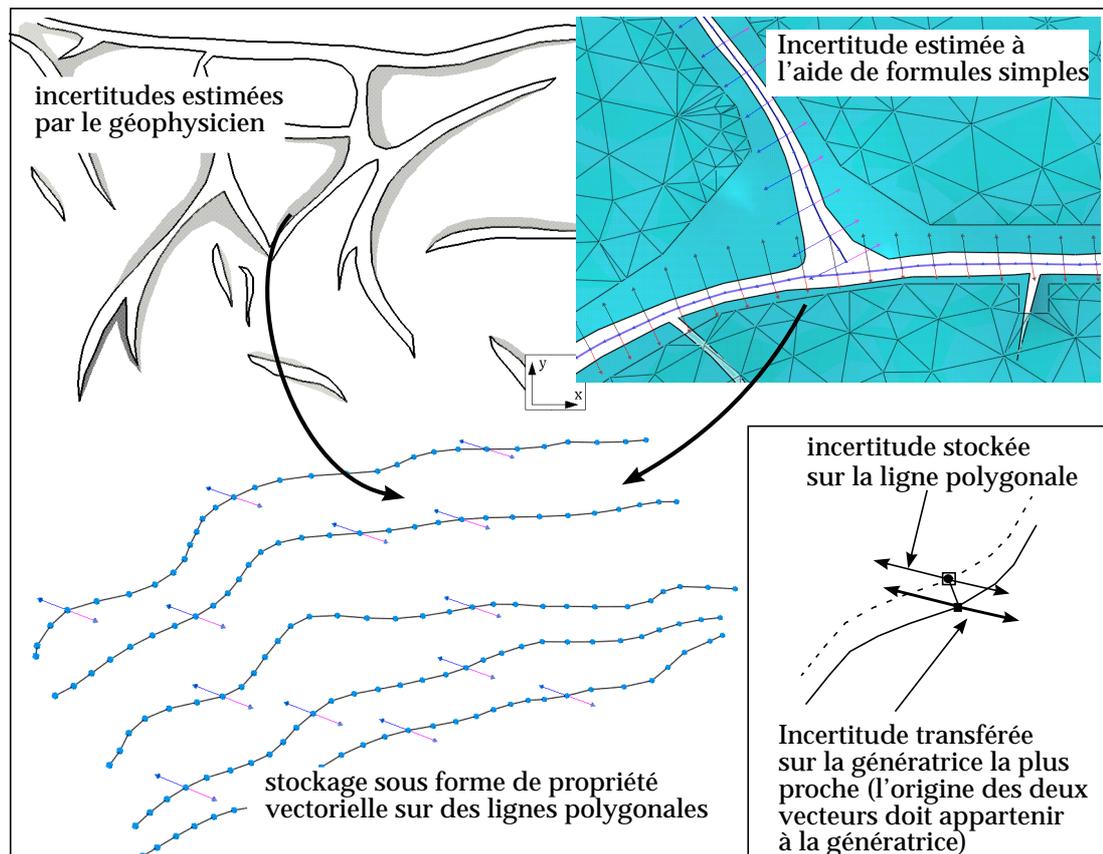
### 2.3.1 Différentes méthodes pour ajouter des vecteurs d'incertitude

Actuellement, il existe deux familles de méthodes pour ajouter des estimations d'incertitudes sur une faille:

- Si les incertitudes sont stockées sous forme de **deux propriétés vectorielles** sur une ligne polygonale (une propriété par côté), il est très facile de les transférer sur l'objet faille. L'atome portant les propriétés vectorielles est utilisé comme origine, et les deux vecteurs sont transformés dans la base locale avant d'être stockés dans une **GFAULTUNCERTAINTYVALUE**.

Ces incertitudes peuvent provenir soit d'une estimation faite par le géophysicien à partir de la qualité des données sismiques, soit résulter des formules simples détaillées dans le paragraphe 1.2 (page 118).

Si le point de la ligne polygonale n'appartient pas à la génératrice la plus proche, une correction est faite afin que l'origine des vecteurs d'incertitude se trouve effectivement sur la génératrice qui les porte.



**Figure 6-16 :** Avant d'être transférées sur l'objet faille, les incertitudes peuvent être stockées sur une ligne polygonale.

**Remarque:** on peut envisager d'adapter cette méthode afin que l'objet stockant les estimations d'incertitudes soit une surface triangulée au lieu d'une ligne polygonale. Le fait de stocker ces incertitudes sur des objets plus "classiques" laisse la possibilité de les estimer directement sur la station d'interprétation, et de les transférer par la suite sur l'objet faille.

- Les incertitudes peuvent être ajoutées directement sur l'objet faille en précisant l'*amplitude* de l'incertitude et en sélectionnant le point *origine* à l'aide de la souris. Pour cette famille de méthodes, deux possibilités sont fournies: les estimations de l'incertitude peuvent être fournies sous forme de **vecteurs** (un pour la partie gauche et un pour la partie droite), ou sous forme **scalaire**.

Dans ce dernier cas, les vecteurs correspondants sont automatiquement calculés en fonction de la normale locale, de la normale à la génératrice concernée ou de la normale à la faille. Selon les configurations, il vaudra mieux utiliser l'une ou l'autre des possibilités offertes [figure 6-17]:

- lorsque la courbure de la faille varie peu, la meilleure solution est d'utiliser la normale locale
- si la forme de la génératrice varie beaucoup en plan, il est préférable d'utiliser la normale à la génératrice ou la normale globale à la faille, afin d'éviter les risque de croisements entre les vecteurs d'incertitude au niveau du maximum de courbure.

Le choix entre ces deux options se fait en fonction de la similarité entre les génératrices: si les génératrices se ressemblent beaucoup, la normale globale à la faille suffit, si elles sont trop différentes, on pourra préférer l'utilisation d'une normale adaptée à chaque génératrice.

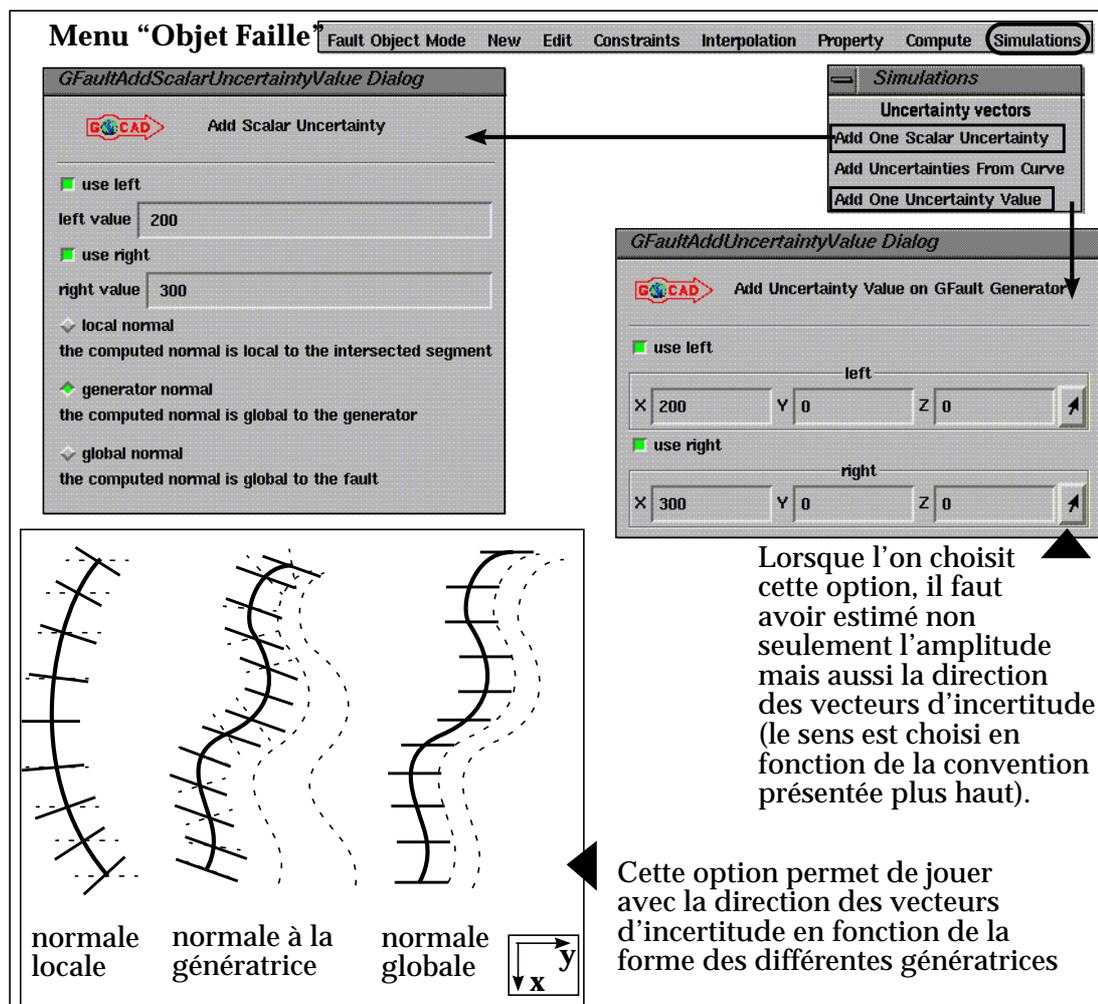


Figure 6-17: Les deux méthodes permettant d'ajouter des estimations de l'incertitude sur l'objet faille. La donnée de vecteurs sera utilisé pour donner une incertitude globale à la faille, la méthode scalaire sera plutôt utilisée pour faire varier l'incertitude le long de la faille.

### 2.3.2 Résultats obtenus après interpolation

Une fois que quelques vecteurs estimés ont été ajoutés à la faille, la méthode d'interpolation présentée précédemment est appliquée afin de calculer l'incertitude au niveau de chacun des points de la faille. Ceci permet de définir le volume à l'intérieur duquel la géométrie de la faille peut être modifiée.

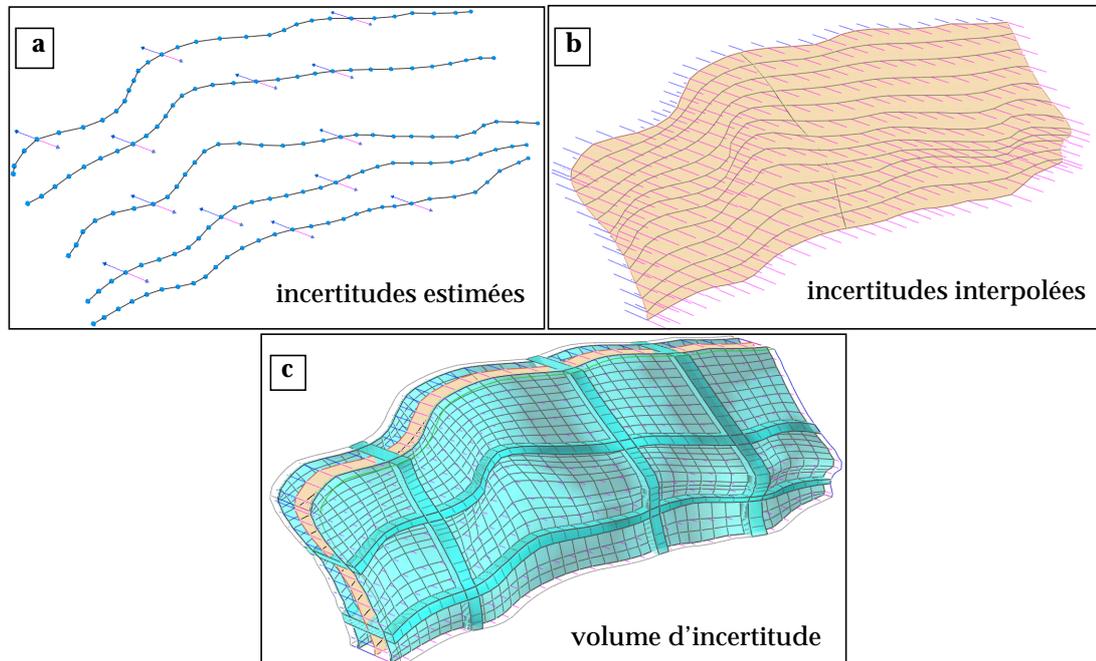


Figure 6-18 : Les quelques vecteurs d'incertitude estimés [a] sont interpolés le long de la faille [b] afin de définir le volume dans lequel la géométrie de la faille peut être simulée.

## 3. Méthodes de simulations

Une fois que ce volume d'incertitude a été défini, les simulations de nouvelles géométries pour la faille peuvent commencer.

Cette section décrit dans un premier temps les hypothèses et le principe général de ces simulations, puis décrit les méthodes qui ont été dérivées à partir de ce principe général.

### 3.1 Principe général

Nous avons pris comme hypothèse de base que le modèle de faille fourni par le structuraliste correspond à la **géométrie la plus probable** de la faille.

Cette géométrie initiale est basée non seulement sur les *données* (interprétation de la sismique, utilisation des données de puits) mais surtout sur les *connaissances* du géologue sur le style structural de la région.

Ne pouvant retrouver automatiquement ces règles structurales à partir des seules données géophysiques actuellement, nous avons décidé de nous appuyer sur ce modèle et de lui accorder une plus forte probabilité d'apparition lors des simulations. Nous avons ensuite proposé des méthodes donnant des simulations équiprobables de la géométrie de la faille autour de cette position de référence.

Dans le même temps, nous avons considéré que ces positions correspondant aux extrémités des vecteurs d'incertitude devaient apparaître durant les simulations, mais avec une probabilité très faible, car très éloignées de la solution de référence.

### 3.1.1 Quelques notions de géostatistique

#### a) Bref historique

Afin de mettre au point les méthodes de simulation pour les failles, nous avons utilisé des notions de base de la géostatistique. Cette branche de la statistique a été mise au point dans les années 1970 afin d'introduire une notion de *continuité spatiale* dans l'analyse des données et la modélisation des phénomènes naturels (corrélations selon des directions privilégiées par exemple). Un des buts principaux de cette science est d'étudier les **incertitudes spatiales**: comment estimer la valeur d'un paramètre en connaissant des valeurs voisines et les corrélations spatiales dans la zone étudiée? Quelle est l'erreur commise lors d'une telle estimation?

Les techniques de simulation issues des géostatistiques ont d'abord été utilisées dans le domaine minier [3 - Damsleth], [92 - Journal 2]. Depuis quelques années, elles sont abondamment utilisées dans le domaine pétrolier [91 - Journal 1].

Les géostatistiques ont surtout contribué à la **modélisation des valeurs de perméabilité dans le réservoir**. Ces études concernent la simulation des hétérogénéités dans les réservoirs et la distribution des propriétés dans les différents faciès ainsi déterminés. Dans ce domaine, deux groupes de méthodes sont proposées. Les *méthodes pixel* affectent directement une valeur de propriété aux cellules de la grille réservoir, compte tenu de certaines hypothèses sur le style de géologie, et la corrélation spatiale de cette propriété [88 - Deutsch 1], [96 - Srivastava], [97 - Suro]. Les *méthodes objets*, elles, s'attachent à délimiter d'abord les zones correspondant à des objets géologiques types (chenaux, lobes, levées...), puis simulent les valeurs de propriété à l'intérieur de chacun des objets ensuite [99 - Tran], [100 - Viseur]. Comme souvent lorsque deux écoles s'affrontent, il est souhaitable de trouver un compromis entre les deux techniques, et d'utiliser un mixage des deux méthodes [92 - Journal 2].

Depuis quelques années, les techniques géostatistiques sont aussi utilisées pour **intégrer l'information provenant des données sismiques** (réparties sur l'ensemble du réservoir, mais peu précises), **et les données de puits** (plus précises mais moins nombreuses et irrégulièrement réparties). Une des techniques correspondant à ces études, le cokrigage colocalisé, commence par découper les puits en différents faciès géologiques. Ensuite, les données sismiques sont analysées au niveau des puits afin de déterminer les meilleures corrélations entre certaines caractéristiques sismiques et les faciès déterminés auparavant. Ceci permet de classer les données sismiques en "faciès sismiques" présentant des caractéristiques communes. Il ne reste plus ensuite, qu'à utiliser la tendance donnée par la sismique loin des puits pour simuler les propriétés correspondantes [figure 6-19].

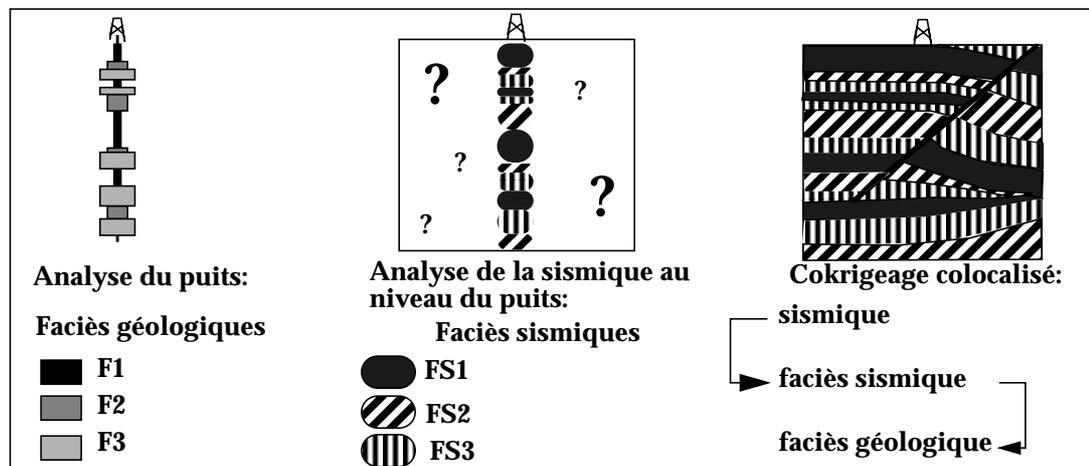


Figure 6-19 : Principe très simplifié de l'utilisation du cokrigage colocalisé pour estimer les faciès à partir de données de puits et de données sismiques.

Malgré l'utilisation croissante des géostatistiques dans le monde pétrolier, encore peu d'études concernent **la simulation de la géométrie des réservoirs**. En effet, la détermination de la géométrie du réservoir est fortement liée à un autre type d'information qui, bien que critique dans beaucoup de cas, n'est *pas facilement quantifiable*, et donc difficile à relier aux méthodes géostatistiques classiques [92 - Journal 2].

Lorsque de telles études structurales sont réalisées, elles ne concernent souvent que la simulation de la géométrie des horizons [68 - Samson], [69 - Shuguang]. En effet, le problème des failles est plus complexe à gérer, puisqu'il existe moins de données sûres d'une part, et qu'il faut maintenir les contacts entre les failles et les horizons après chaque simulation d'autre part.

Cependant, nous pensons que certaines des méthodes géostatistiques peuvent être adaptées à l'étude des incertitudes entourant la géométrie des failles, puisque l'on veut calculer des réalisations équiprobables de leur forme et de leur position, tout en gardant une corrélation forte dans les deux directions principales de ces dernières. Ce type d'étude se rapproche des **méthodes de simulations objet** utilisées pour modéliser les hétérogénéités du réservoir.

## b) Vocabulaire

### 1. Variable aléatoire (V.A.)

Une variable aléatoire  $Z(x, y, z)$  est une variable qui peut prendre un certain nombre de valeurs (*réalisations*) ( $z_i, i=1, \dots, N$ ) ayant chacune une probabilité donnée ( $p_i, i=1, \dots, N$ ). Dans le cadre des géostatistiques, les données disponibles sont considérées comme l'une des réalisations de cette variable aléatoire.

Cette variable aléatoire peut être scalaire (l'une des coordonnées d'un point dans l'espace) ou vectorielle (vecteur normal en ce point par exemple). Lorsque le nombre de réalisations possibles est limité, on parle de variable aléatoire discrète.

Les probabilités d'une telle variable aléatoire doivent vérifier les conditions suivantes:

$$p_i \geq 0, \text{ pour } i=0, \dots, N \quad \text{et} \quad \sum_{i=1}^N p_i = 1$$

### 2. Fonction de répartition de probabilité (Cumulative distribution function = *cdf*)

Lorsque la variable aléatoire peut prendre un nombre de valeurs infini, on parle de variable aléatoire continue. Sa distribution de probabilité est alors caractérisée par une fonction: la *fonction de répartition* (ou *cdf* dans la littérature). Cette fonction est caractérisée de la manière suivante:

$$F(z) = \text{Prob}\{Z \leq z\} \quad \text{“probabilité que la V.A. } Z \text{ soit inférieure à } z\text{”}$$

Dans le cas discret, cette courbe cumulative correspond à la notion d'histogramme cumulé:

$$F(z_i) = \sum_{(j)} p_j \text{ avec } (j): \text{ ensemble des réalisations telles que } z_j \leq z_i$$

### 3. Fonction de densité de probabilité (Probability Density Function: *pdf*)

Une autre manière de caractériser la variable aléatoire est de représenter la dérivée de la fonction de répartition (lorsqu'elle existe):  $f(z) = F'(z)$

Cette notion est beaucoup plus facile à imaginer dans notre cas, et c'est sur cette fonction que nous avons surtout basé la méthode que nous proposons pour simuler la géométrie de la faille.

#### 4. Modèles de variables aléatoires

De nombreuses fonctions de densité existent pour modéliser des choses simples. Parmi elles, on peut citer [figure 6-20]:

- la **loi uniforme**: dans le cas d'une variable discrète pouvant prendre  $N$  valeurs, chaque valeur a la probabilité  $1/N$  d'apparaître
- la **loi gaussienne**: elle est très utilisée dans les algorithmes classiques. Ceci est dû en partie aux symétries la caractérisant, qui lui donnent de bonnes propriétés mathématiques.
- la **loi triangulaire**: elle permet d'associer une probabilité maximum à l'une des valeurs, une probabilité nulle aux deux valeurs extrêmes de l'intervalle des valeurs possible et une variation linéaire de la probabilité entre ces valeurs caractéristiques. Cette loi présente aussi la possibilité d'être dissymétrique par rapport à la valeur de probabilité maximale.

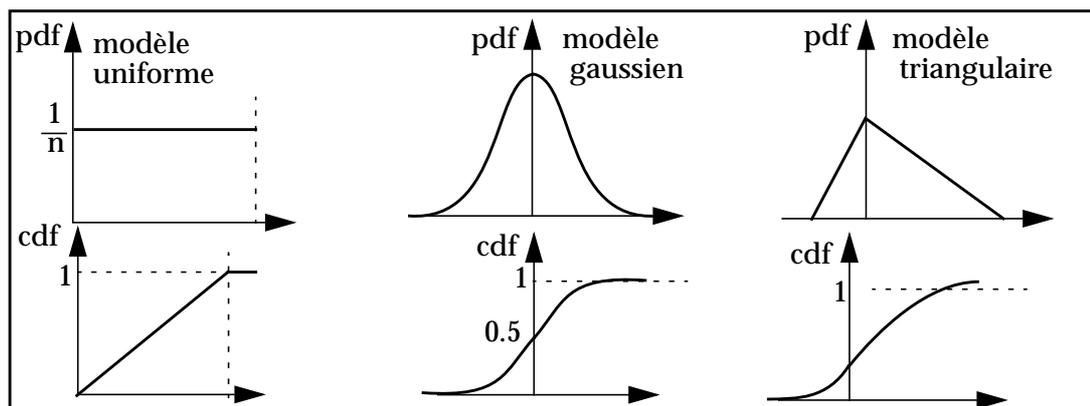


Figure 6-20 : Quelques fonctions de densité de probabilité couramment utilisées dans les modèles, et courbes cumulatives correspondantes.

Afin de répondre aux deux critères que nous nous sommes fixés et dans un souci de simplicité, nous avons choisi d'associer à chaque point de la faille une fonction de densité de probabilité **triangulaire**, avec la probabilité maximum au niveau de la position de référence et la probabilité minimum aux extrémités des vecteurs d'incertitude. Ce choix a aussi été guidé par la possibilité de gérer des *vecteurs d'incertitude asymétriques* de part et d'autre de la faille. Comme les vecteurs d'incertitudes peuvent varier le long de la faille, les *cdf locaux* peuvent être différents d'un point à un autre.

**Remarque:** lorsque la faille n'est pas fournie au départ, et que nous ne disposons que des données géophysiques pour construire la faille, ou lorsque l'on ne veut pas privilégier l'interprétation du structuraliste, il est possible de choisir d'associer une loi uniforme à chacun des points, plutôt qu'une loi triangulaire. De plus, rien n'empêche de choisir une autre loi (gaussienne par exemple). Le processus général de simulation restera le même, quelle que soit la loi choisie.

#### 5. Fonction aléatoire

Toute la géostatistique est basée sur l'observation que toute variable aléatoire représentant un phénomène naturel de l'espace 3D (ou *variable régionalisée*) possède une structure de variabilité particulière: deux valeurs  $Z(u)$  et  $Z(u+h)$  éloignées d'une distance  $h$  sont *corrélées*. L'ensemble des variables aléatoires  $Z(u_i)$  définies en chacun des points  $u_i$  du domaine d'étude s'appelle **fonction aléatoire**  $Z(u)$ . Du fait de la complexité de la variabilité de la fonction  $Z(u)$  dans l'espace (anisotropies, discontinuités), son étude directe est exclue. De la même manière, l'interprétation des variations de  $Z(u)$  et  $Z(u+h)$  de manière complètement indépendante n'est pas acceptable, puisqu'elle ne rend pas compte de la corrélation spatiale [91 - Journal 1].

### c) Notion de variogramme

Les variables aléatoires  $Z(u_i)$  sont corrélées entre elles, la corrélation dépendant à la fois de la distance  $h$  séparant deux points ( $u_i$ ) et ( $u_i+h$ ), et de la nature de la variable considérée. Le degré de variabilité spatiale entre deux variables aléatoires  $Z(u_1)$  et  $Z(u_2)$  est mesuré par une fonction appelée *variogramme* qui mesure l'espérance du carré de la différence entre les deux variables aléatoires:

$$\gamma(u_1, u_2) = E\{[Z(u_1) - Z(u_2)]^2\}$$

Sous l'hypothèse de stationnarité, cette variabilité entre  $Z(u)$  et  $Z(u+h)$  ne dépend que de la distance entre  $h$  entre les deux points, et est indépendante de la position  $u$ . Le variogramme permet donc de mesurer la *corrélation* ( $C(h)$ ) entre les différentes variables aléatoires:  $C(h) = 1 - \gamma(h)$ . Si  $u_1$  et  $u_2$  sont confondus,  $\gamma(u_1, u_1)$  est nul, et plus on les deux positions sont éloignées, plus cette valeur augmente. La distance à partir de laquelle le variogramme ne varie plus correspond à la distance à laquelle les variables varient de manière indépendante.

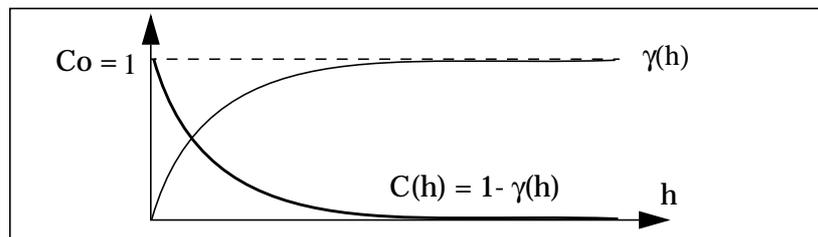


Figure 6-21 : Relation entre le variogramme et la fonction de corrélation.

Ce variogramme est estimé de la manière suivante [figure 6-22]:

1. un pas d'échantillonnage  $h$  est choisi
2. dans une direction donnée, on calcule la variance du paramètre pour tous les couples de points distants de  $h$ . Si les deux valeurs sont égales, la variance sera nulle. Au contraire, lorsque elles sont éloignées, la variance est élevée.
3. on recommence la même opération pour des couples de points séparés de la distance  $2h$ , puis  $3h$ ,.... Ceci permet de tracer un graphe qui représente la variabilité du paramètre selon la direction étudiée en fonction de la distance  $h$ . La distance à partir de laquelle les valeurs ne sont plus corrélées s'appelle la portée.
4. on recommence les opérations 2 et 3 dans différentes directions de l'espace. L'ensemble des portées obtenues dessine un ellipsoïde, dont les trois axes principaux donnent la direction de variation maximum, la direction de variation minimum et la direction de variation moyenne.

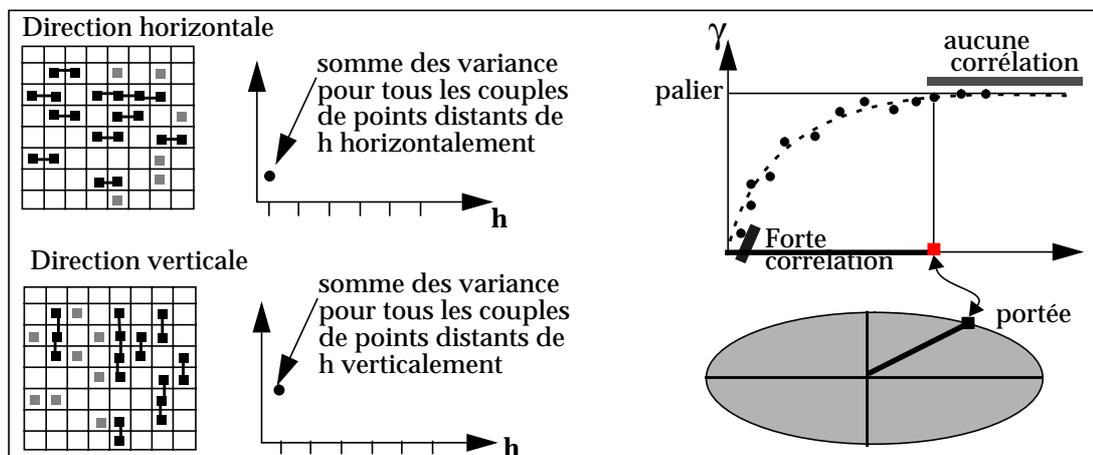


Figure 6-22 : Comment construire un variogramme (par soucis de simplicité la grille est représentée en deux dimensions uniquement, mais la méthode est transposable en trois dimensions).

Le variogramme mesurant indirectement la corrélation entre deux variables aléatoires, il est souvent utilisé pour calculer par des méthodes de simulation des valeurs  $Z(u+h)$  inconnues à partir de valeurs  $Z(u)$  connues. De nombreuses méthodes existent mais elles supposent en général qu'un nombre minimum de valeurs sont connues pour pouvoir contraindre les simulations sur tout le domaine d'étude. Ceci est rarement le cas pour la géométrie des failles, et nous verrons un peu plus loin que toutes ces méthodes ne sont pas forcément adaptées à notre problème.

Le paragraphe suivant montre comment une technique particulière (le calcul de *champs de probabilités*) a permis de concilier les méthodes géostatistiques et le peu de données disponibles pour caractériser les failles.

### 3.1.2 Utilisation de ces notions pour les simulations

#### a) Problèmes à éviter pour respecter la géométrie des failles

Comme cela a été souligné dans le chapitre 3, les contraintes géométriques concernant les failles sont beaucoup plus nombreuses que pour les horizons. Les surfaces de failles correspondent à des surfaces de glissement, le long desquelles les couches ont pu coulisser [98 - Thibaut].

Si l'on ne fait pas attention à **corrélérer les nombres aléatoires  $Z(u)$  dans les deux directions principales de la faille**, on obtient des résultats qui ne respectent plus du tout la forme initiale de la faille [figure 6-23].

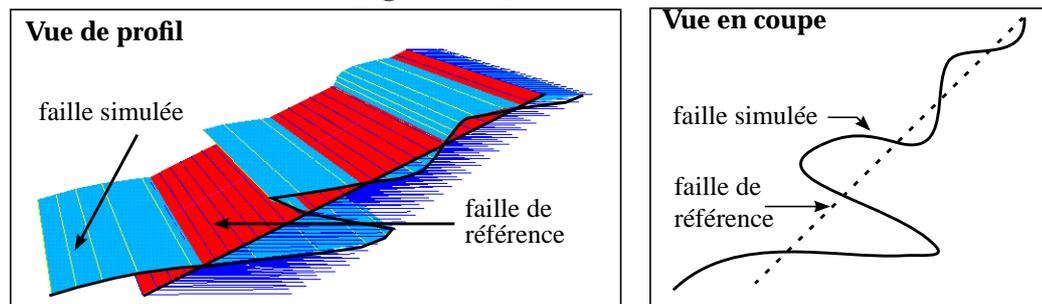


Figure 6-23 : Simulation qui semble inadmissible pour représenter une faille.

On pourrait éventuellement accepter certaines simulations “non lisses” et attribuer les variations plus fortes de la courbure à des différences de lithologie. En effet, lorsque la faille traverse des couches plus dures (calcaire, grès), sa forme est plus rectiligne que lorsqu'elle traverse des couches argileuses [figure 6-24].

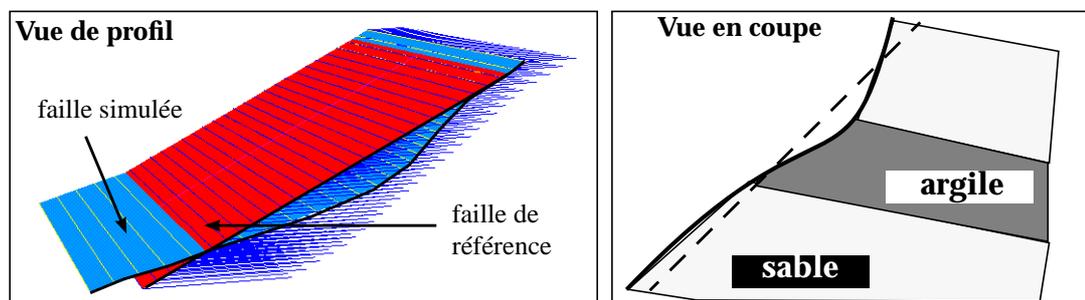


Figure 6-24 : Un type de résultat acceptable si la lithologie correspond, mais rejeté par défaut dans les méthodes de simulations proposées.

Ne disposant d'aucune information sur la lithologie des jeux de données fournis, nous avons préféré restreindre un peu le domaine des réalisations acceptables plutôt que de risquer de simuler des formes non géologiques.

Considérant que si la forme de la faille est fortement liée à la lithologie, cette caractéristique se retrouve dans la géométrie de la faille de référence, nous avons sélectionné des fonctions aléatoires simples qui respectent la forme initiale de la faille (pendage, sauf avis contraire de l'utilisateur; directions principales...).

Ceci est réalisé en imposant une **corrélation forte** entre les variables aléatoires calculés le long de la faille.

### b) Solution choisie: l'utilisation des champs de probabilités

Une technique, qui permet à séparer le calcul de la distribution **locale** de probabilité, de la génération de modèles équiprobables **globalement** sur le domaine d'étude, est connue sous le nom de "*simulation par champ de probabilités*" [90 - Froidevaux].

Elle consiste donc à calculer une **fonction aléatoire** corrélée spatialement de manière *non conditionnelle*. En effet, la plupart du temps, nous ne disposons d'aucune information pour conditionner les simulations sur la géométrie des failles.

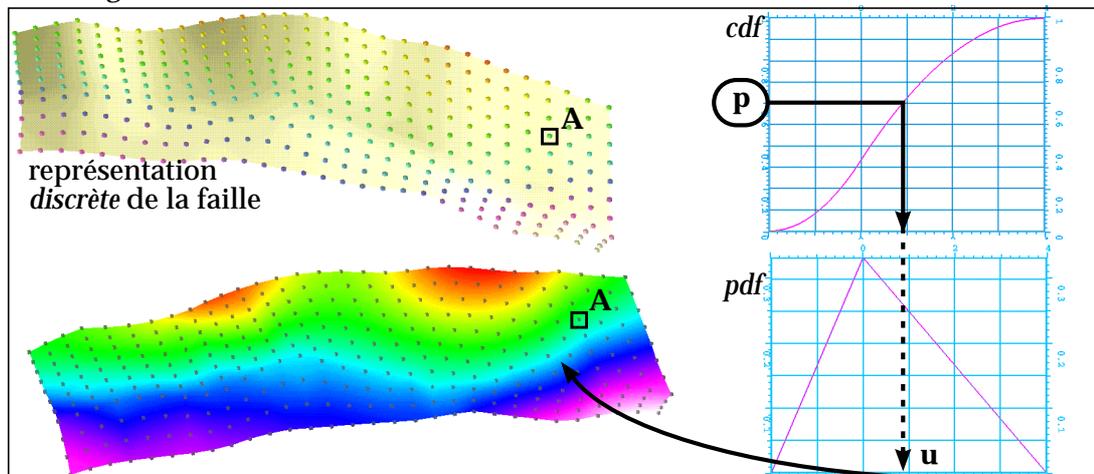
Le seul conditionnement est fourni par la longueur des vecteurs d'incertitude et par la corrélation demandée entre les valeurs simulées. Le fait de pouvoir choisir le type de fonction aléatoire indépendamment de la distribution locale de probabilité permet de *mieux contrôler* la forme des simulations et de sélectionner les fonctions aléatoires qui donnent des formes acceptables pour les failles.

Afin de garder une forme acceptable du point de vue géologique, nous avons décidé que cette fonction aléatoire devait en outre respecter les deux règles suivantes:

- c1: être *monotone* selon la direction verticale
- c2: avoir une *grande longueur de corrélation* selon la direction horizontale

Suivant cette méthode, la simulation d'une nouvelle géométrie pour la faille peut être décomposée en trois étapes simples:

- estimation des lois de *distribution locales* (lois triangulaires dans notre cas)
- calcul d'une simulation non conditionnée du *champ de probabilités* à l'aide d'une fonction aléatoire  $P(u)$  (calcul d'une réalisation  $p$  de la variable aléatoire  $P(u_j)$  pour chacun des points  $u_j$  appartenant au graphe des nœuds de la faille). Cette fonction aléatoire doit respecter les deux conditions c1 et c2.
- calcul de la nouvelle position  $u_j'$  correspondant à cette réalisation  $p$  en utilisant la fonction de répartition (*cdf*) attachée au point  $u_j$ . [figure 6-25]. Cette dernière étape est largement utilisée dans le domaine de géostatistique sous le nom de *tirage de Monte Carlo*.



**Figure 6-25 :** Principe général des simulations: à chaque point de la faille une fonction de densité de probabilité triangulaire (ou uniforme) est associée. Une réalisation de la variable aléatoire est ensuite utilisée pour calculer une nouvelle position à partir de la cdf associée à ce point.

### c) Présentation des fonctions aléatoires utilisées

Partant de cette méthode générale, on peut imaginer n'importe quelle forme pour la fonction aléatoire  $P(u)$ . Le seul critère à respecter, est que  $P(u)$  doit avoir une distribution uniforme dans l'intervalle  $[0, 1]$  et respecter les deux conditions c1 et c2.

Différentes fonctions aléatoires sont proposées. De la méthode la plus simple à la méthode la plus complexe, on peut trouver [figure 6-26]:

1. fonction aléatoire **constante** sur tout le graphe
2. fonction aléatoire **constante** selon la direction horizontale, et variant **linéairement** selon la direction verticale
3. fonction aléatoire variant **linéairement** selon la direction horizontale et **constante** selon la direction verticale
4. fonction aléatoire variant **linéairement** suivant les deux directions principales de la faille
5. fonction aléatoire **constante** suivant la direction verticale et variant selon une **fonction non linéaire** suivant la direction horizontale
6. fonction aléatoire variant selon une **fonction non linéaire** suivant la direction verticale et **constante** suivant la direction horizontale
7. fonction aléatoire variant **linéairement** selon la direction verticale, et selon une **fonction non linéaire** selon l'axe horizontal
8. fonction aléatoire variant selon une **fonction non linéaire** à la fois verticalement et horizontalement.

Toutes les fonctions n'ont pas été testées, seules les fonctions de type 1, 2, 5, 7 et 8 ont été testées, car elles ont pu être reliées à des paramètres de la faille (trace horizontale, pendage, position). Comme cela sera montré plus loin, les fonctions appartenant aux types 6 et 8 sont plus difficiles à définir, puisqu'elles ne respectent pas forcément la condition c1 (monotonie suivant la direction verticale).

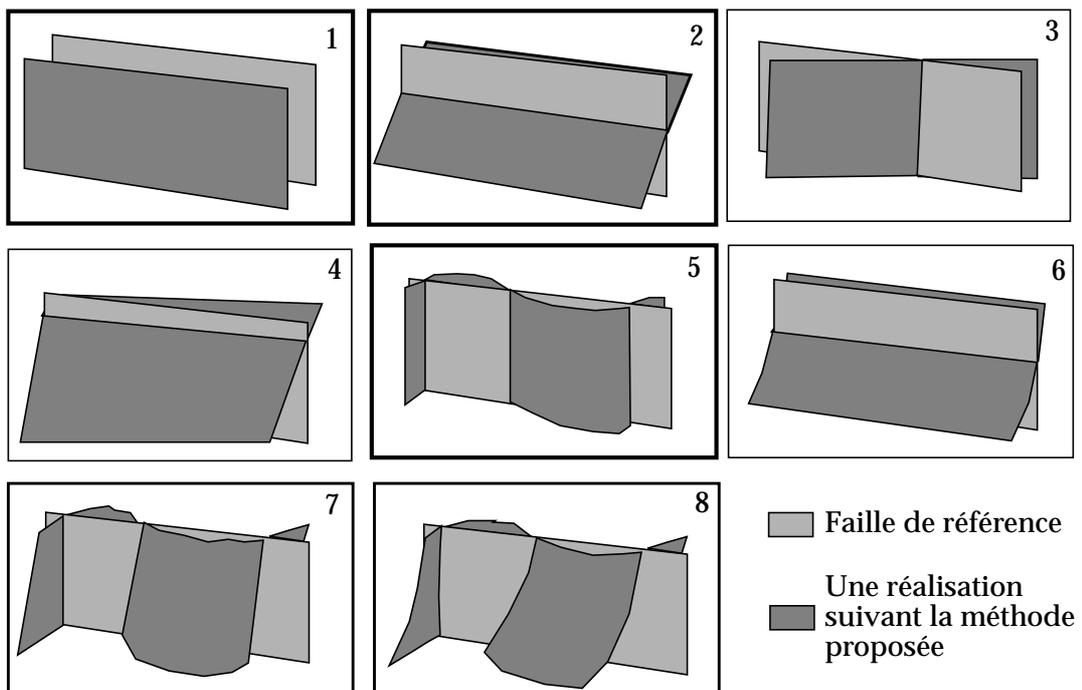


Figure 6-26 : Différentes fonctions aléatoires qui peuvent être proposées (seules les cadres en gras ont été testés pour le moment).

### 3.2 Méthodes simples

Lorsque les modifications à apporter sont assez modestes (décalage de la position de la faille sans changer sa forme, ou modification du pendage), deux fonctions aléatoires simples et très rapides sont proposées [93 - Lecour 2].

Dans un premier temps, le champ de vecteurs d'incertitudes est considéré relativement homogène. En fin de section, les problèmes posés lorsque cette hypothèse n'est pas respectée seront discutés.

#### 3.2.1 Fonction aléatoire constante = *décalage latéral*

Lorsque l'on considère que la forme de la faille est correcte, mais que des doutes existent uniquement sur sa position, une première méthode permet de déplacer la faille latéralement [figure 6-27].

Pour ce faire, une réalisation unique de la variable aléatoire  $P(u)$  est calculée de manière aléatoire, suivant une loi uniforme dans l'espace  $[0,1]$ . On a donc:

si  $D$  représente le graphe des points de la faille de référence,

$$P_1(u) = p = \text{Constante, avec } p \in \text{Uniforme}[0, 1] \quad \forall (u \in D)$$

Outre le calcul à partir d'une loi uniforme dans l'intervalle  $[0, 1]$ , un autre mode permet à l'utilisateur de sélectionner un certain nombre de positions remarquables (cas extrême, certains quantiles).

Lorsque les vecteurs d'incertitude sont égaux tout au long de la faille (cas le plus courant actuellement), cette méthode correspond à une simple translation.

Dans le cas contraire, puisque les *cdf* locales sont liées aux longueurs des vecteurs d'incertitude, l'utilisation du même nombre  $p$  tout au long de la faille crée une légère déformation par rapport à la faille de référence. Ce point sera rediscuté plus loin.

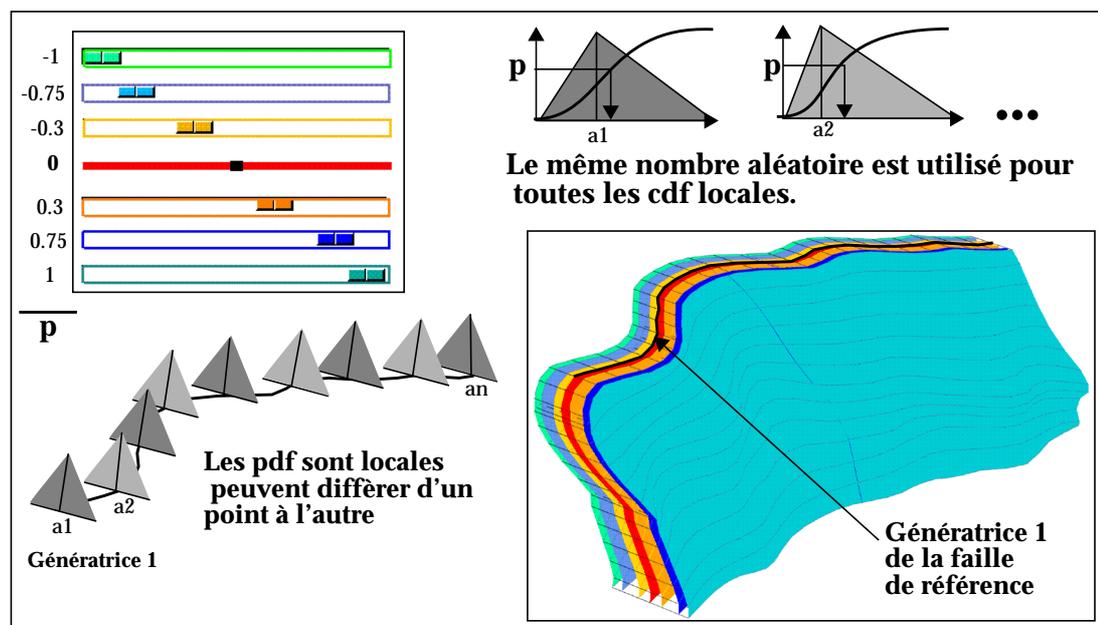


Figure 6-27 : Exemple de la méthode de simulation la plus simple: la fonction aléatoire est constante sur toute la faille. Le même nombre aléatoire  $p$  est utilisé pour inverser les différentes courbes cumulatives locales.

### 3.2.2 Fonction linéaire verticalement et constante horizontalement = modification du pendage

La méthode présentée dans le paragraphe précédent n'est un cas particulier de cette méthode. Elle peut être utilisée lorsque la forme de la faille est considérée correcte, mais lorsqu'il existe une incertitude sur le pendage.

Elle consiste à calculer deux nombres aléatoires  $p_1$  et  $p_2$  à partir d'une loi uniforme dans l'intervalle  $[0, 1]$  et à les utiliser sur les *cdf* locales des deux génératrices extrêmes de la faille. Pour tous les autres points, une fonction permet de calculer une interpolation linéaire entre  $p_1$  et  $p_2$ . On obtient donc une fonction aléatoire représentée par une droite verticalement et par la forme de référence horizontalement.

Ainsi, si  $D_1, D_2, D^*$  représentent respectivement les points de la génératrice supérieure, de la génératrice inférieure et de tous les autres points,  $n$  le nombre de génératrices et  $k$  l'indice de la génératrice sur laquelle le point  $u$  se trouve, on obtient pour cette fonction aléatoire:

$$p_2(u) = (1 - \alpha) \cdot p_1 + \alpha \cdot p_2 \quad \forall (u \in D^*) \quad \text{avec } \alpha = \frac{k}{n}$$

$$p_2(u) = p_1 \quad \forall (u \in D_1), \text{ et } p_2(u) = p_2 \quad \forall (u \in D_2)$$

Si l'utilisateur n'autorise pas d'inversion de pendage par rapport à la faille principale, on choisit de plus  $p_1$  inférieur à  $p_2$ . Dans le cas contraire, les deux réalisations sont affectées dans leur ordre d'apparition. Pour cette méthode il existe aussi deux modes: les nombres peuvent être calculés aléatoirement ou choisis par l'utilisateur pour vérifier quelques simulations particulières [figure 6-28].

Nous verrons plus loin les problèmes posés par cette méthode, a priori simple.

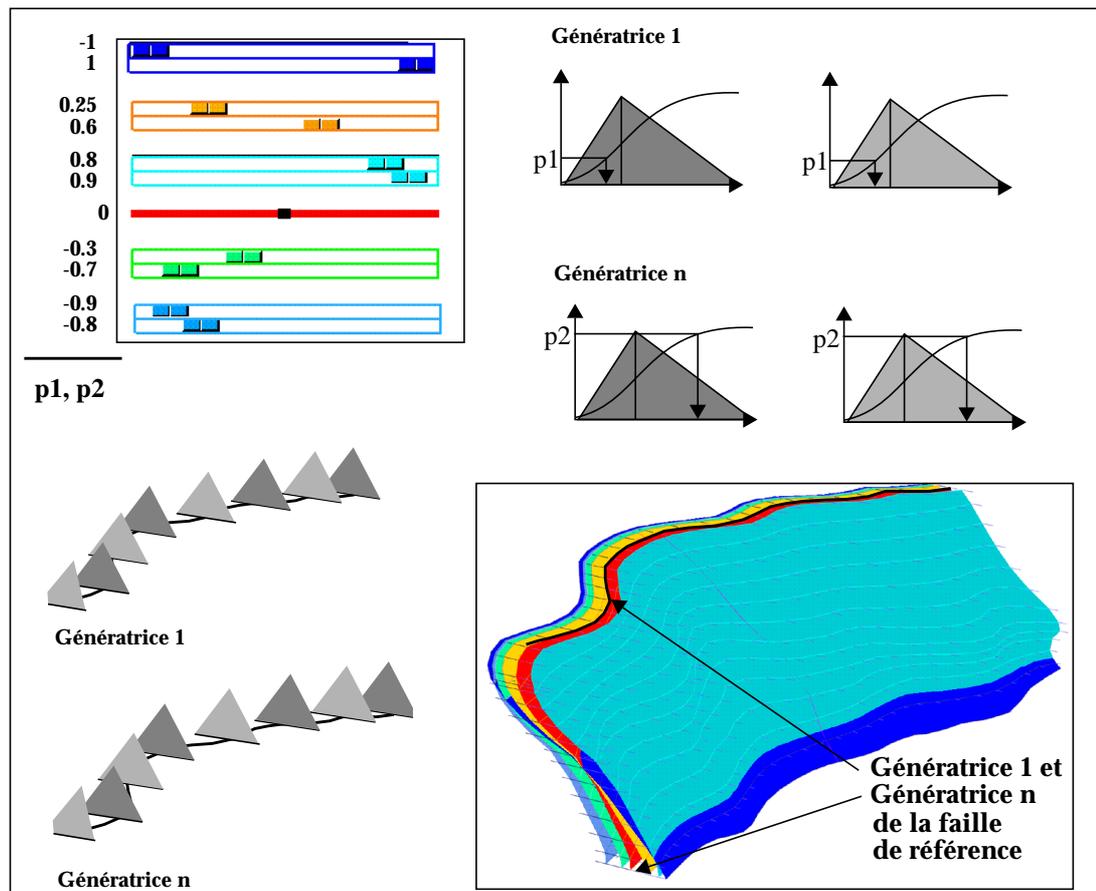


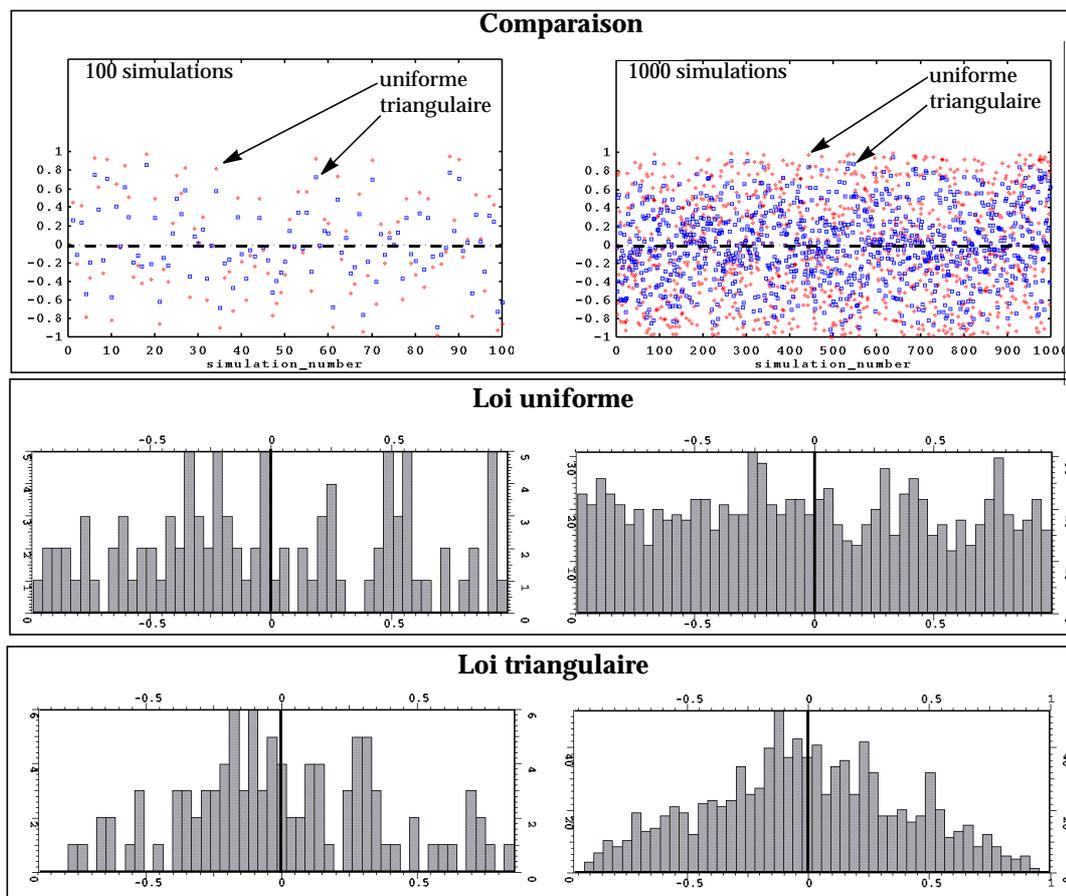
Figure 6-28 : Exemple de simulation permettant une variation de pendage. Dans ce cas, la fonction aléatoire correspond à une série de droites.

### 3.2.3 Influence du type de loi

Les méthodes précédentes sont suffisamment simples pour que l'on puisse comparer les résultats obtenus en utilisant une loi uniforme et ceux obtenus en utilisant une loi triangulaire.

A condition de faire un assez grand nombre de simulations (500 au minimum pour voir une différence entre les deux distributions), on vérifie bien que lorsqu'une loi triangulaire est utilisée, les positions simulées sont plus concentrées autour de la position de référence, tandis que lorsqu'une loi uniforme est utilisée, les positions simulées se répartissent de manière homogène sur tout l'intervalle.

L'image suivante permet de comparer des simulations réalisées avec un champ de vecteur homogène, ce qui élimine un certain nombre de problèmes [figure 6-29].



**Figure 6-29 : A condition de faire assez de simulations, on vérifie bien que lorsqu'une loi uniforme est choisie (au centre), les simulations sont régulièrement réparties, alors que pour une loi triangulaire (en bas), les simulations sont concentrées autour de la position de référence.**

Les méthodes présentées dans ce paragraphe sont très rapides, vu le peu de nombres aléatoires calculés. Elles permettent de faire varier non seulement la position de la faille, mais aussi son pendage. Dans l'état actuel de l'étude des incertitudes, ces méthodes sont suffisantes. Par contre, elles ne permettent pas de modifier la forme de la faille dans le plan horizontal. Or, pour certaines études, comme l'étude du risque pris lors de l'implantation d'un nouveau puits, il est important de voir l'influence d'un changement local de forme de la faille.

Nous avons donc travaillé sur d'autres types de fonctions aléatoires plus complexes et permettant de modifier la forme de la faille dans la direction horizontale.

### 3.3 Méthodes plus complexes

Ces méthodes sont plus coûteuses en temps, puisqu'il faut calculer de nombreux paramètres, mais donnent plus de souplesse aux simulations.

Une première méthode faisant intervenir des fonctions sinus et cosinus (méthode de Fourier) permet de modifier la forme de la faille de manière non linéaire, tout en contrôlant la sinuosité de la courbe obtenue.

Une deuxième méthode (cokrigage colocalisé) a été essayée afin de faire varier la géométrie de manière non linéaire dans les deux directions principales de la faille. Nous pensons qu'elle n'est pas réellement adaptée à la modification de la géométrie des failles, et qu'il vaudrait mieux revenir à des méthodes plus analytiques pour ce type de simulation.

#### 3.3.1 Fonction constante verticalement et non linéaire horizontalement = modification de la géométrie en carte

##### a) Principe (très simplifié)

Cette méthode a entièrement été codée par Arben Shtuka [95 - Mallet 3], et seuls les résultats de cette fonction ont été utilisés. Elle est inspirée des travaux de Srivastava [96 - Srivastava] qui propose une méthode rapide pour générer des images équiprobables d'hétérogénéités de réservoirs (chenaux, lobes...).

L'idée de la méthode, est d'utiliser une longueur de corrélation donnée dans l'intervalle [0,1] (la *portée*) et une combinaison linéaire de fonctions sinus et cosinus afin d'obtenir une fonction variant de manière assez lisse dans l'intervalle [0, 1]. Plus la portée est grande et plus la fonction obtenue sera lisse (à condition d'utiliser un modèle de variogramme gaussien).

De petites variations  $Z_k$  et  $Z'_k$  sont utilisées afin d'introduire une partie aléatoire dans la fonction cyclique. La formule ci-dessous donne une idée très simplifiée de l'équation utilisée:

$$z(u) = \sum_{k=0}^n \cos(ku) \cdot Z_k(u) + \sin(ku) \cdot Z'_k(u)$$

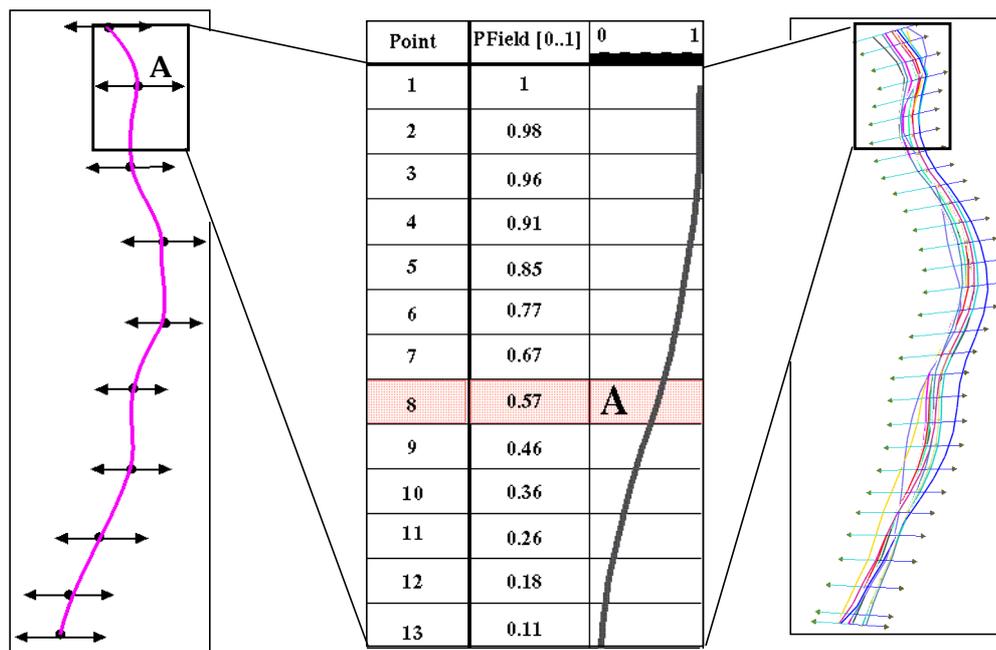


Figure 6-30 : quelques résultats obtenus avec la méthode décrite plus haut.

### b) Influence de la portée sur la sinuosité

Comme nous l'avons dit plus haut, la portée donnée en entrée intervient dans la fonction et influe sur la sinuosité de la courbe dans l'espace  $[0, 1]$ .

Si la portée est proche de 1, toutes les valeurs simulées doivent être très proches de la première valeur calculée. On obtiendra donc une courbe assez plate. Au contraire, si la portée est faible, le calcul de la première valeur a beaucoup moins d'influence sur les valeurs simulées par la suite et la courbe sera plus sinueuse.

Compte-tenu de la forme des failles que nous avons eu à traiter, et à l'échelle à laquelle nous travaillons, les variations suivant le plan horizontal sont relativement faibles. C'est pourquoi une portée assez grande (0.8 ou 0.9) a été choisie.

Ceci dit, l'utilisateur peut décider d'avoir une plus forte sinuosité et donner une plus petite portée qui donnera une forme de faille oscillant beaucoup plus.

Ces petites valeurs de portée sont généralement mieux adaptées pour simuler des trajets de chenaux par exemple [100 - Viseur].

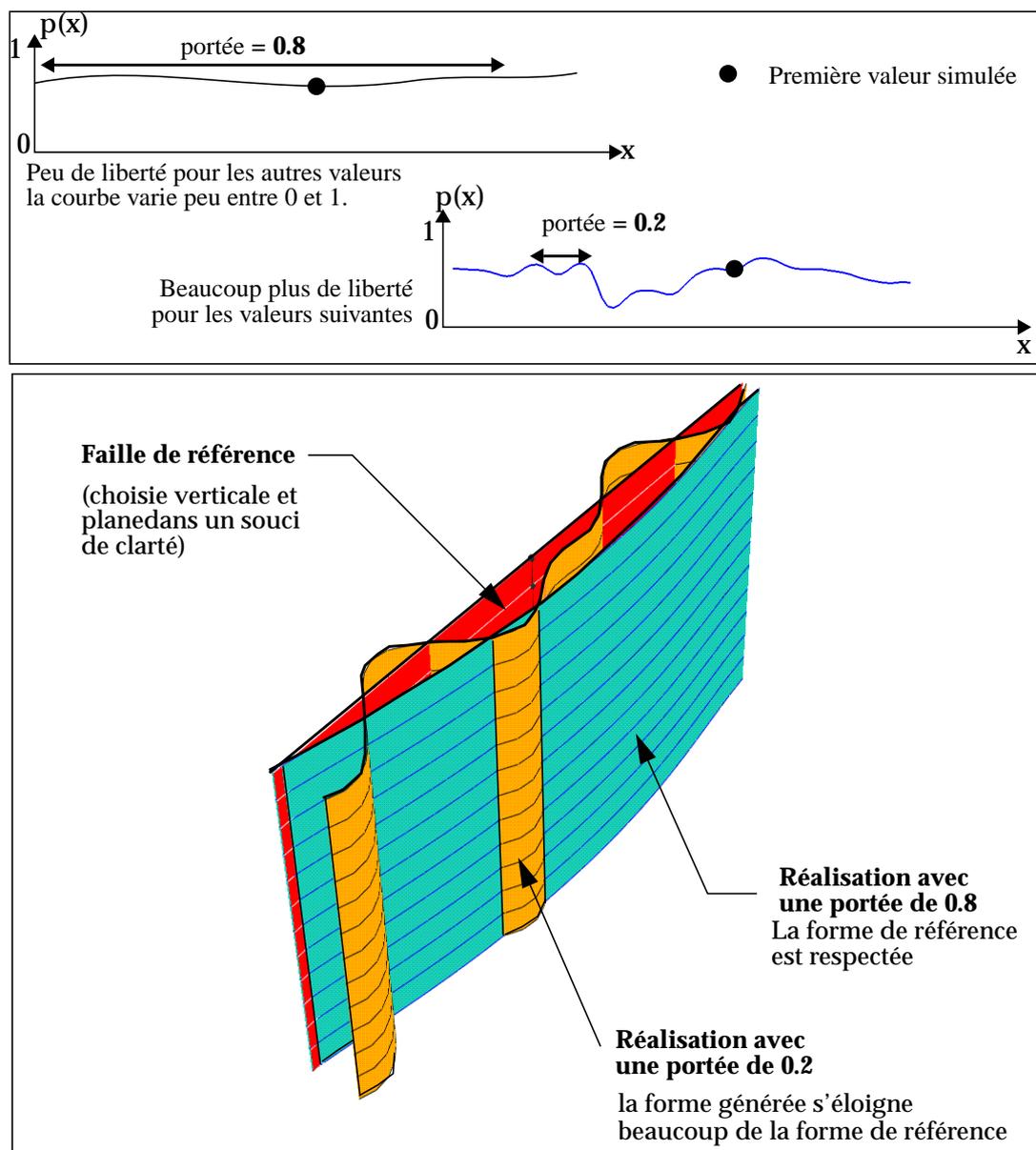


Figure 6-31 : Influence de la portée sur la forme de la courbe obtenue

### 3.3.2 Fonction non linéaire à la fois verticalement et horizontalement

#### a) Principe de la méthode testée

Afin de simuler en même temps, une variation de pendage et une modification de la faille en carte, nous avons essayé d'adapter une méthode actuellement beaucoup utilisée pour estimer les propriétés pétrophysiques des couches à partir des données du puits (assez sûres, mais éparses) et des données sismiques (présentes sur l'ensemble du domaine, mais connues moins précisément).

Cette méthode (le cokrigage colocalisé) consiste donc à mélanger des données considérées comme *sûres mais peu nombreuses* et des données *moins certaines* mais réparties sur tout le domaine lors des simulations, en donnant un coefficient de corrélation spatiale entre les données sûres et les données incertaines par l'intermédiaire d'un variogramme.

Le problème majeur de l'utilisation de cette méthode dans notre cas, est que nous n'avons en général aucune donnée pour guider la simulation (ni "sûre", ni incertaine). Nous avons donc essayé de simuler de telles données dans un premier temps, afin d'orienter les simulations vers le résultat souhaité.

En effet, nous avons pensé que l'on pouvait simuler un premier type de données variant latéralement le long de la faille et devant être respectées; et une tendance générale sur toute la faille forçant la simulation à être monotone suivant l'axe Z.

Cette simulation a été réalisée en construisant une grille autour de la faille, et en affectant des valeurs de propriété le long de trois génératrices de la faille (les génératrices supérieure, médiane, et inférieure). Afin de respecter le pendage initial, les valeurs simulées sur l'axe central sont choisies proches de la position de référence [-0.2; 0.2], tandis que les valeurs des deux autres axes sont calculées dans des intervalles choisis aléatoirement, mais symétriques par rapport à cette position de référence [-1; 0.8] et [0.8; 1] par exemple

La tendance verticale est elle donnée par une fonction variant linéairement avec Z:  $f(x, y, z) = \alpha \cdot z$ . Plusieurs valeurs de  $\alpha$  ont été essayées sans que cela n'ait une grande influence sur le résultat.

Enfin, un variogramme permet de donner la corrélation qui doit être appliquée entre les deux types de données lors de la simulation.

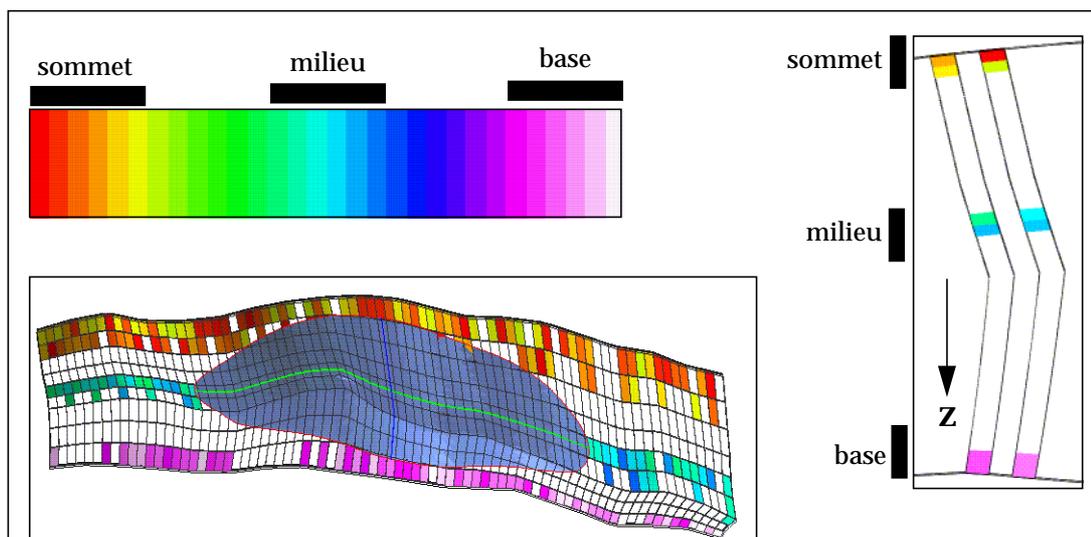


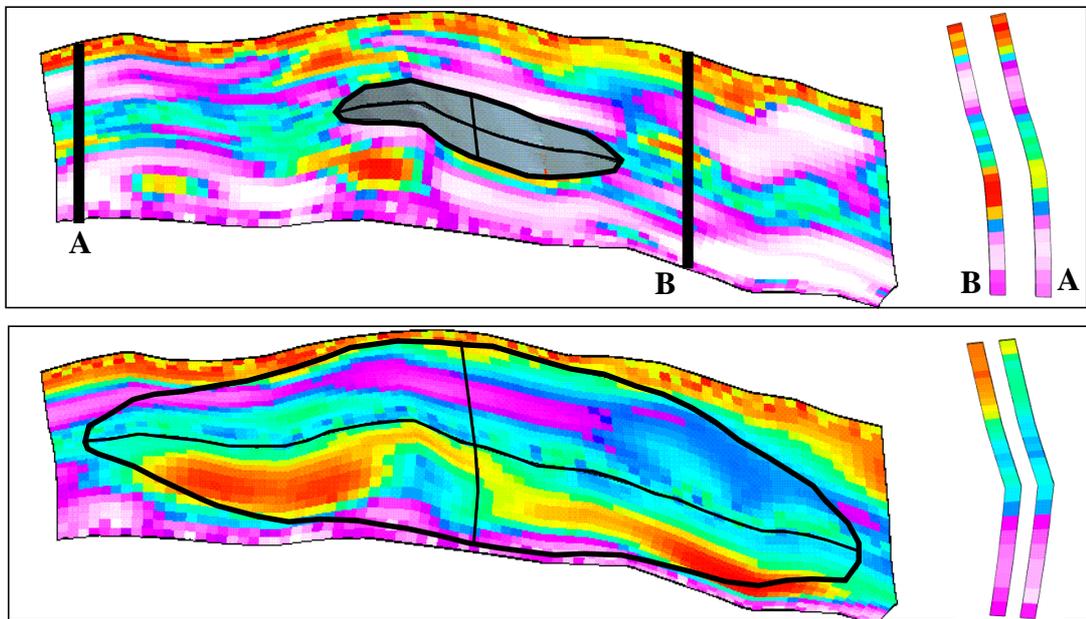
Figure 6-32 : Calcul de la propriété utilisée pour guider la simulation.

### b) Critique des résultats obtenus

On pourrait penser qu'en utilisant une grande portée (surtout selon l'axe vertical), on obtiendra une simulation assez "lisse" pour être monotone sur toutes les verticales de la faille. Pourtant, même si l'utilisation d'un variogramme à grande portée améliore beaucoup les résultats obtenus, on ne peut garantir la monotonie de la fonction aléatoire le long d'une verticale (condition c1 que nous nous sommes fixée au départ).

L'explication est que nous essayons de faire des choses *contradictoires*: nous souhaitons avoir une variabilité non négligeable selon un axe horizontal, et, afin de faire éventuellement varier le pendage de la faille, nous ne voulons pas utiliser une portée égale à 1 dans la direction verticale (sinon, cela revient à utiliser le même nombre aléatoire sur toute la verticale, et donc cela revient à utiliser la méthode présentée dans le paragraphe précédent).

Comme il est impossible de garantir la monotonie d'une simulation en jouant uniquement sur le modèle de variogramme, il est normal que la fonction aléatoire ne soit pas monotone selon l'axe Z.



**Figure 6-33 :** Avec un variogramme à grande portée (en bas), le résultat est meilleur, mais la fonction aléatoire n'est toujours pas garantie monotone suivant l'axe Z.

Cette méthode statistique n'étant pas adaptée au problème que nous traitons, nous avons décidé de nous concentrer sur les trois méthodes précédentes, plus analytiques.

En utilisant ces trois méthodes de base, qui correspondent chacune à un type de mouvement particulier (décalage latéral, modification du pendage et modification de la forme en carte), et en les combinant entre elles, il est possible d'obtenir une palette de fonctions aléatoires plus fournie.

La combinaison de ces trois méthodes ne permet ni de générer des réalisations de type 6, ni de type 8 (voir [figure 6-26]) où la modification du pendage ne se fait pas de manière linéaire. Cependant, cela offre une palette plus variée que ce qui existe actuellement, tout en permettant de contrôler les résultats obtenus.

Si cela est souhaité, il sera toujours possible, plus tard, de proposer d'autres fonctions aléatoires analytiques permettant de modifier la géométrie de manière non linéaire dans les deux directions principales de la faille.

### 3.4 Combinaison des trois méthodes de base

Les trois méthodes présentées précédemment permettent de faire varier séparément la position de la faille, son pendage ou sa forme selon un plan horizontal. Il nous a semblé utile de pouvoir proposer à l'utilisateur de choisir un, deux, voire les trois modes de simulations et de les combiner entre eux.

Ce travail, encore en cours est actuellement réalisé par Cécile Audinet dans le cadre de son D.E.A [86 - Audinet]. Cette section décrit le principe de ce travail, et les problèmes qui restent encore à l'heure actuelle.

#### 3.4.1 Pourquoi combiner les trois méthodes précédentes?

Les deux premières méthodes présentées sont très rapides: au maximum, deux nombres aléatoires sont calculés pour définir la fonction aléatoire. Elles permettent une modification globale de la faille (décalage ou changement de pendage) sans modifier la forme de la faille localement. Elles permettent d'explorer rapidement tout le volume d'incertitude, mais sont loin de donner toutes les solutions possibles.

La troisième méthode qui est plus lente, permet à sa manière d'explorer tout le volume d'incertitude. Elle autorise la modification locale de la géométrie de la faille de référence, mais ne permet pas de modification globale de la faille comme un décalage latéral ou une modification de pendage.

Ces fonctions peuvent suffire lorsque l'on étudie un type d'incertitude particulier :

- incertitude sur la **position** lorsque la *forme* et le *pendage* sont considérés comme sûrs,
- incertitude sur le **pendage** lorsque la *forme* globale et la *position* sont considérées comme sûres,
- incertitude sur la **forme** lorsque la *position* et le *pendage* sont considérés comme sûrs.

Cependant, dans une étude des incertitudes plus globale, il peut arriver que le géologue souhaite **combiner plusieurs types d'incertitudes**, voire utiliser les trois critères lorsque la sismique est très mauvaise et que la faille est vraiment mal définie.

C'est pourquoi, il nous a semblé utile de proposer d'autres fonctions aléatoires combinant les avantages des différentes méthodes: pouvoir à la fois appliquer un décalage latéral et une modification du pendage, ou une modification de la géométrie en cartel et une modification de pendage... etc.

L'ensemble de ces fonctions aléatoires permet de balayer l'étendue des possibilités de manière plus complète.

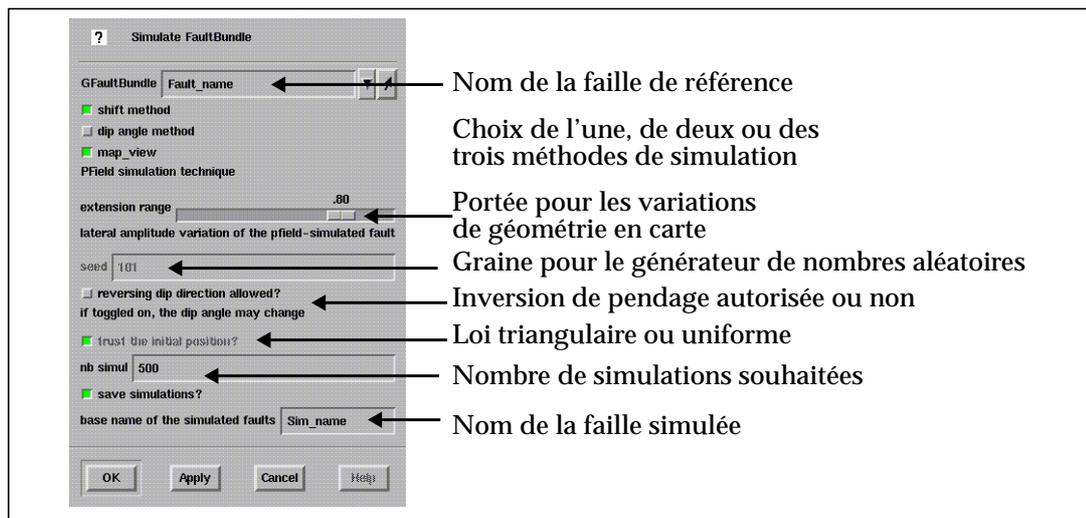
#### 3.4.2 Principe général

Le but de cette nouvelle commande est de pouvoir combiner les avantages des différentes méthodes et de gérer en même temps plusieurs paramètres intervenant dans la simulation d'une nouvelle géométrie pour la faille. Elle utilise le code défini pour les trois types de simulation décrits ci-dessus, mais a aussi entraîné des changements majeurs dans la gestion des simulations des variables aléatoires.

### a) Description de l'interface

Les différents choix proposés à l'utilisateur sont regroupés sous un même panneau de commande. Parmi ces choix, on trouve

- Le type de simulation choisie: décalage latéral *et/ou* changement de pendage *et/ou* modification de la géométrie selon le plan horizontal
- Lorsqu'une modification de pendage est demandée, l'utilisateur peut choisir de conserver le type initial (normal ou inverse) de la faille de référence ou peut autoriser un changement plus radical permettant de passer d'une faille normale à une faille inverse. Ce dernier cas de figure peut arriver lorsque le pendage de la faille est quasi vertical et qu'aucune information géologique supplémentaire ne permet de trancher sur le type de la faille. Cependant, nous pensons que dans la majorité des cas, le type de la faille de référence doit être conservé.
- Lorsqu'une modification de la géométrie en carte est demandée, l'utilisateur peut choisir la valeur de la portée. Par défaut, cette valeur est choisie à 0.8.
- le type de cdf: si l'utilisateur décide de privilégier la position de référence, une loi triangulaire sera attachée à chaque point de la faille. Par contre, si la position de référence n'a pas une plus grande probabilité que les autres positions, une loi uniforme sera attachée aux différents points de la faille. Cette dernière possibilité se produit par exemple lorsque la faille a été construite uniquement à partir des données sismiques, sans aucun apport géologique supplémentaire. Dans ce cas, la faille de "référence" ne correspond à rien de géologique et n'a pas une plus grande probabilité d'apparition que les autres.



**Figure 6-34 : Boite de dialogue permettant de choisir les différents paramètres de simulation pour une faille donnée: le type de fonction aléatoire, le type de loi à inverser localement, autoriser ou non l'inversion de pendage et le choix de la portée pour la modification de la géométrie horizontale.**

### b) Déroulement global des simulations

Les choix de l'utilisateur permettent de calculer le type de simulation. Chaque type de simulation de base (décalage vertical, changement de pendage ou modification horizontale) est codé par un booléen qui vaut 0 si ce type de simulation n'est pas choisi et 1 sinon. La combinaison linéaire suivante permet de calculer le type de simulation qui sera appliqué:

$$\text{code} = \text{décalage} + 2 * \text{pendage} + 4 * \text{horizontal} \text{ (code va donc de 1 à 7)}$$

En fonction du code ainsi calculé, la fonction adéquate est appelée autant de fois que de nombre de simulations demandées par l'utilisateur.

Le calcul des nouvelles positions se fait en deux grandes étapes:

**1. Calcul du champ de probabilités dans l'intervalle [0,1].** Ceci est réalisé en utilisant une fonction aléatoire adaptée au type de simulation choisie. La seule condition est que cette fonction aléatoire suive une loi uniforme dans cet intervalle [0,1]. Nous avons ajouté les deux conditions  $c1$  et  $c2$  énoncées précédemment (monotonie suivant l'axe  $Z$  et grande portée horizontale) (voir page 137).

**2. Utilisation des réalisations obtenues et des *cdf* locales** pour calculer les nouvelles positions d'un certain nombre de points. Nous verrons que dans les cas simples, les *cdf* locales sont directement combinées aux nombres calculés durant la première étape. Dans les types de simulation plus complexes, plus de travail est nécessaire avant de pouvoir réaliser la transformation de l'espace [0,1] vers l'espace réel ( $x,y,z$ ).

En fin de simulation, on est capable d'avoir: les positions de référence pour chacun des points et un vecteur de translation par rapport à cette position de référence pour chaque point. Si cela a été demandé, un objet faille portant le nom préfixe\_ *nom-de-faille\_numéro-de-simulation* est construit en appliquant la bonne translation à chaque point de la faille de référence.

Afin de mieux détailler les sept cas possibles, nous utilisons par la suite un exemple où tous les vecteurs d'incertitude sont égaux tout au long de la faille. Ceci élimine énormément de problèmes. Quelques cas particuliers (champ de vecteur homogène, mais avec les vecteurs droits et gauches inégaux, champ de vecteurs inhomogène), seront abordés en fin de chapitre.

### 3.4.3 Cas où tous les vecteurs d'incertitude sont égaux

#### a) Code = 1: *décalage latéral*

Comme cela a été décrit plus haut, cette fonction aléatoire est la plus simple que nous proposons. Elle consiste à calculer une réalisation  $p$  de la variable aléatoire suivant une loi uniforme dans l'intervalle [0,1]. Ce nombre  $p$  est ensuite utilisé pour inverser toutes les *cdf* locales. Ces *cdfs* peuvent être triangulaires si la position de référence est privilégiée ou uniforme si toutes les positions sont considérées comme équiprobables [figure 6-35] (a).

#### b) Code = 2: *changement de pendage*

Dans ce cas, nous avons décidé de modifier le pendage de la faille, de manière *centrée* par rapport à la faille de référence. Si ceci est facilement réalisable dans le cas particulier où nous nous plaçons actuellement, cela pose plus de problème lorsque les vecteurs ne sont pas égaux de part et d'autre de la faille.

Cette fonction aléatoire consiste à calculer un premier nombre  $p1$  suivant une loi uniforme dans l'intervalle [0,1]. La deuxième étape consiste à calculer le symétrique de ce point:  $p2=1-p1$ . Si l'utilisateur n'autorise pas de modification de type pour la faille, il faut de plus vérifier que  $p1$  soit inférieur à  $p2$  (pour une faille normale) [figure 6-35] (b).

Le nombre  $p1$  est utilisé pour inverser toutes les *cdfs* de la génératrice composant le sommet de la faille, tandis que le nombre  $p2$  est utilisé pour inverser toutes les *cdfs* des points constituant la base de la faille. Entre ces deux points, une simple interpolation linéaire est réalisée.

c) **Code = 3:** *changement de pendage + décalage latéral*

Dans ce cas, la fonction est très similaire au cas précédent. La seule différence est que les nombres  $p1$  et  $p2$  ne doivent pas être symétriques par rapport à la position centrale  $0.5$ .

Ils sont tous les deux choisis aléatoirement suivant une loi uniforme dans l'intervalle  $[0,1]$ . Seule la condition  $p1 < p2$  est vérifiée lorsque la faille doit rester normale [figure 6-35] (c).

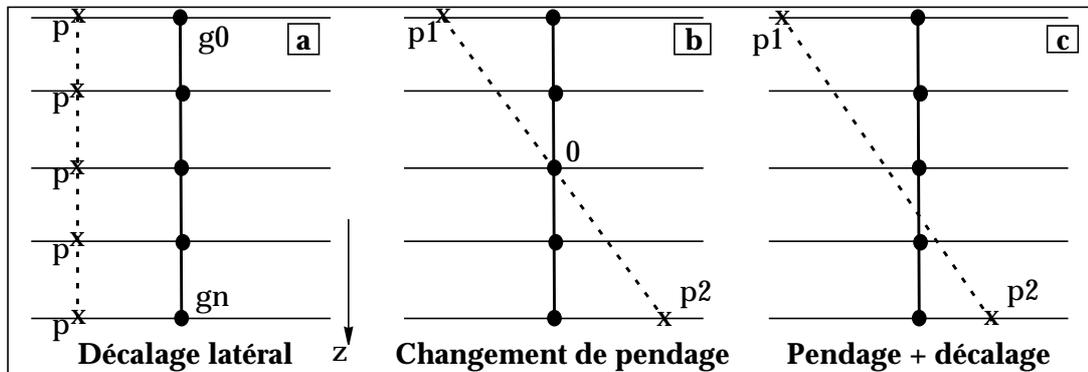


Figure 6-35 : Fonctions aléatoires préservant la forme dans le plan horizontal

d) **Code = 4:** *modification de la géométrie en carte uniquement*

Pour cette méthode, un champ de probabilités est calculé à l'aide de la fonction aléatoire  $P3(u)$  (voir page 142). Pour chaque génératrice, ce tableau est ensuite directement utilisé pour inverser les différentes *cdf*s locales. Tous les points portant l'indice  $0$  utiliseront  $tableau[0]$ , tous les points portant l'indice  $1$  utiliseront  $tableau[1]$ , et ainsi de suite jusqu'à l'indice  $n$  [figure 6-37] (a).

e) **Code = 5:** *modification en carte + décalage latéral*

A partir de cette fonction aléatoire, le code devient plus complexe à gérer. Chaque fonction correspondant aux codes 5 à 7 commence de la même manière: le calcul du tableau de champ de probabilités dans l'espace  $[0,1]$  à partir de la fonction aléatoire  $P3(u)$ . En fonction du type simulation choisi, ce tableau est ensuite transformé de l'intervalle  $[0,1]$  vers un intervalle plus approprié à la méthode choisie et inclus dans cet intervalle  $[0,1]$  [figure 6-36].

Le calcul de cet intervalle selon les étapes suivantes:

1. calcul de certains éléments dans l'espace  $[0, 1]$ , afin de définir l'intervalle restreint: la borne inférieure de l'intervalle restreint pour la génératrice supérieure ( $p1_t$ ), pour la génératrice médiane ( $p1_m$ ) et pour la génératrice inférieure ( $p1_b$ ), ainsi que les bornes supérieures correspondantes ( $p2_t$ ,  $p2_m$ ,  $p2_b$ ). Afin d'éviter toute distorsion, les bornes supérieures sont choisies de manière à ce que la longueur des intervalles soit égale pour toutes les génératrices de la faille.
2. Pour chaque point  $point_k$  de la génératrice considérée, la valeur de probabilité  $p_k$  est transformée de l'intervalle  $[0,1]$  vers le nouvel intervalle  $[p1, p2]$  en  $p_k'$ .
3. C'est cette valeur  $p_k'$  qui est ensuite utilisée pour inverser la *cdf* locale. Comme la fonction aléatoire  $P(u)$  suivait une loi uniforme dans l'intervalle  $[0,1]$ , la fonction aléatoire transformée  $P'(u)$  suit une loi uniforme dans l'intervalle restreint (une simple anamorphose a été appliquée).

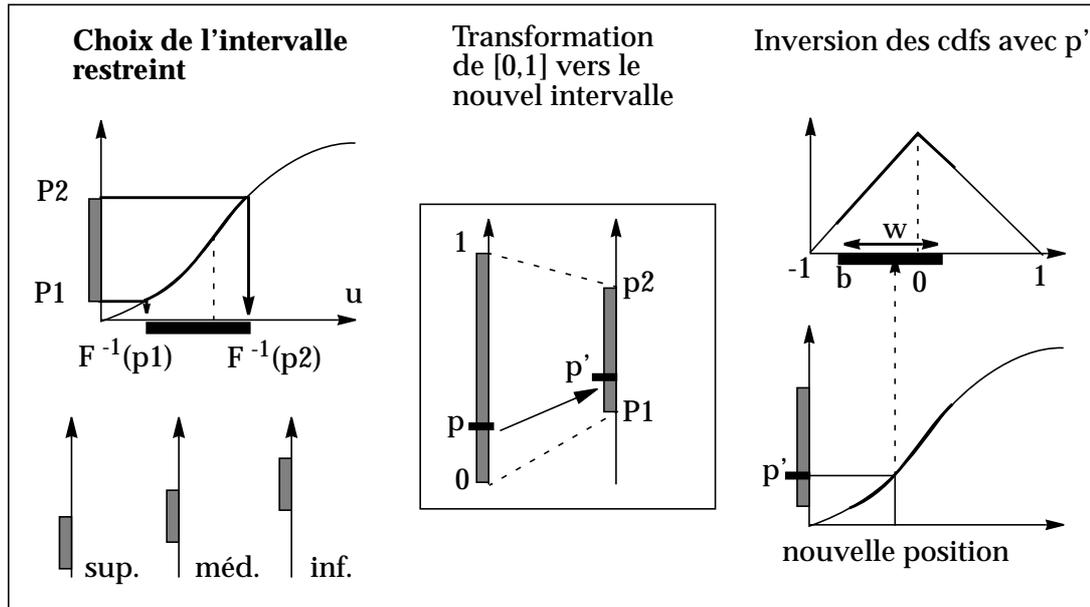


Figure 6-36 : Transformation du tableau de nombres aléatoires de l'intervalle  $[0,1]$  vers un intervalle plus approprié. L'image du point  $p_1$  dans l'espace réel est noté:  $F^{-1}(p_1) = b$ .

Lorsqu'un décalage latéral est demandé en plus de la modification en carte, les éléments suivants sont calculés:

- $p_1$  est choisi suivant une loi uniforme dans l'intervalle  $[0, 1]$ .
- $p_2$  est choisi aléatoirement selon une loi uniforme dans l'intervalle  $[p_1, 1]$
- $w = p_2 - p_1$

Ensuite, l'algorithme suivant est appliqué [figure 6-37] (b):

*Exemple de code 6.1. modification en carte et décalage latéral*

```

1  Si nb_atom représente le nombre de génératrices
2  et nb_points le nombre de points par génératrice
3  Pour toutes les génératrices de  $i=0$  à nb_atom
4    Pour tous les points de  $k=0$  à nb_points
5      calculer  $pk' = pk*w + p_1$ 
6      inverser la cdf locale à l'aide de  $pk'$ 

```

**f) Code = 6:** *modification en carte + changement de pendage*

Lorsqu'un changement de pendage est demandé en plus de la modification horizontale, les éléments suivants sont calculés (cas d'une faille normale, avec obligation de conserver le type de pendage):

- $w$  est choisie selon une loi uniforme dans  $[0, 1]$
- $p_{1m} = 0.5 - w/2$
- $p_{2m} = 0.5 + w/2$
- $p_{1t}$  est choisi suivant une loi uniforme dans  $[0 ; p_{1m}]$
- $p_{2t} = p_{1t} + w$
- $p_{1b} = 1 - p_{1t}$
- $p_{2b} = p_{1b} + w$

Ensuite, l'algorithme suivant est appliqué [figure 6-37] (c):

*Exemple de code 6.2. modification horizontale et changement de pendage*

```

1  Si nb_points représente le nombre de points par génératrice
2
3  Pour la génératrice supérieure
4      Pour tous les points de k=0 à nb_points
5          calculer  $p_k' = pk*w + p1_t$ 
6          inverser la cdf locale à l'aide de  $pk'$ 
7
8  Pour la génératrice médiane
9      Pour tous les points de k=0 à nb_points
10         calculer  $pk' = pk*w + p1_m$ 
11         inverser la cdf locale à l'aide de  $pk'$ 
12
13 Pour la génératrice inférieure
14     Pour tous les points de k=0 à nb_points
15         calculer  $pk' = pk*w + p1_b$ 
16         inverser la cdf locale à l'aide de  $pk'$ 
17
18 Pour les autres points une interpolation linéaire est réalisée
19

```

---

**g) Code = 7:** *modification horizontale + décalage latéral + changement de pendage*

Lorsque les trois types de simulation sont demandés en même temps, les éléments suivants sont calculés:

- $w$  est choisi suivant une loi uniforme dans l'intervalle  $[0, 1]$ .
- $p1_t$  est choisi suivant une loi uniforme dans l'intervalle  $[0 ; 1-w]$
- $p1_b$  est choisi suivant une loi uniforme dans l'intervalle  $[p1_t, 1-w]$

Ensuite, l'algorithme suivant est appliqué [figure 6-37] (d):

*Exemple de code 6.3. trois types de modification simultanés*

```

1  Si nb_points représente le nombre de points par génératrice
1
2  Pour la génératrice du haut de la faille
3      Pour tous les points de k=0 à nb_points
4          calculer  $pk' = pk*w + p1_t$ 
5          inverser la cdf locale à l'aide de  $pk'$ 
6
7  Pour la génératrice du bas de la faille
8      Pour tous les points de k=0 à nb_points
9          calculer  $pk' = pk*w + p1_b$ 
10         inverser la cdf locale à l'aide de  $pk'$ 
11
12 Pour les autres points une interpolation linéaire est réalisée
13

```

---

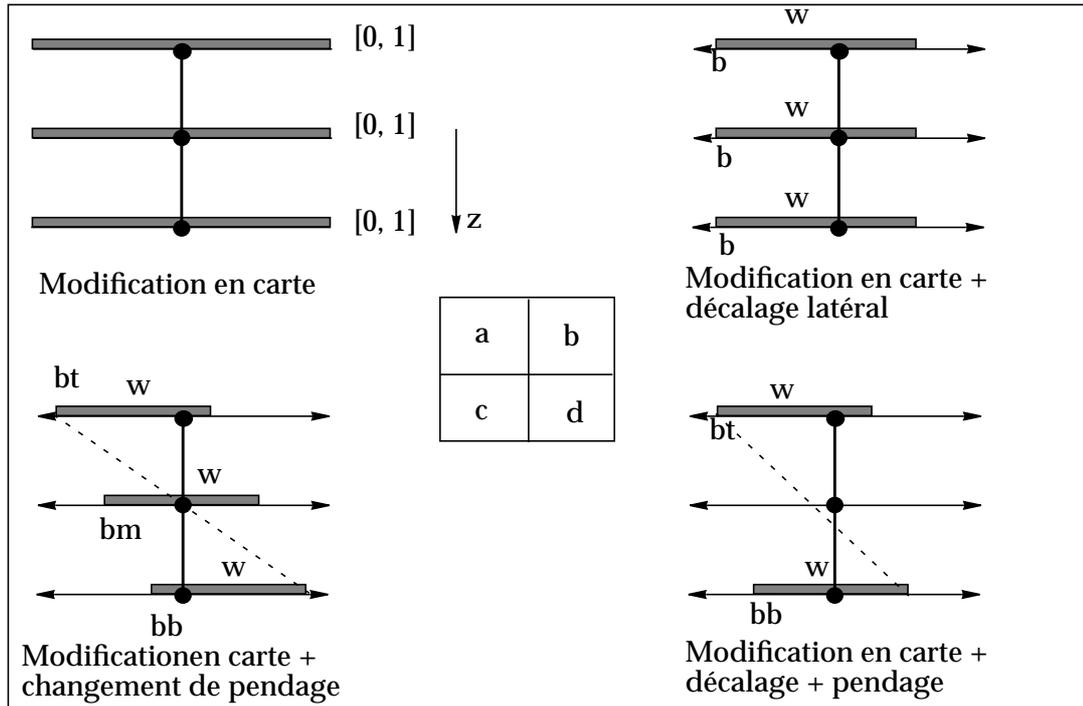


Figure 6-37 : Sélection des intervalles en fonction du type de simulation sélectionné

#### 3.4.4 Limites de validité de la méthode et cas «pathologiques»

Comme cela a été annoncé plus haut, le seul conditionnement de ces méthode est fourni par les estimations de l'incertitude de part et d'autre de la faille.

Tout ce qui a été décrit dans le paragraphe précédent n'est valable que dans le cas où tous les vecteurs d'incertitude sont égaux le long de la faille (ou au moins lorsque le champ de vecteurs est homogène). Lorsque ce n'est pas le cas, certaines des méthodes de simulation conduisent à des configurations contradictoires avec les données d'incertitude fournies par l'utilisateur.

Ce paragraphe décrit quelques cas posant des problèmes d'incompatibilité entre les vecteurs d'incertitude fournis et le mode de simulation demandé. Les différentes solutions envisageables sont ensuite discutées.

##### a) Premier cas «pathologique»

Lorsque les vecteurs d'incertitude sont égaux tout au long de la faille, mais avec des vecteurs *gauche* et *droite* différents, toutes les simulations avec une modification du pendage uniquement posent problème.

En effet, dans notre formalisme, la méthode qui consiste à modifier uniquement le pendage de la faille, correspond à une rotation de la faille autour de l'axe constitué par la génératrice médiane. Ceci est du au fait que dans le paragraphe précédent, nous pouvions considérer que les deux réalisations  $p1$  et  $p2$  étaient symétriques par rapport à la valeur  $0.5$ , autant pour une loi uniforme que pour une loi triangulaire, puisque dans le cas particulier étudié, ces lois étaient symétriques par rapport à la position de référence.

Dès que le rapport entre les vecteurs *gauche* et *droite* est différent de 1, on obtient une cassure le long d'une verticale, si l'on utilise le code décrit précédemment (figure 6-38).

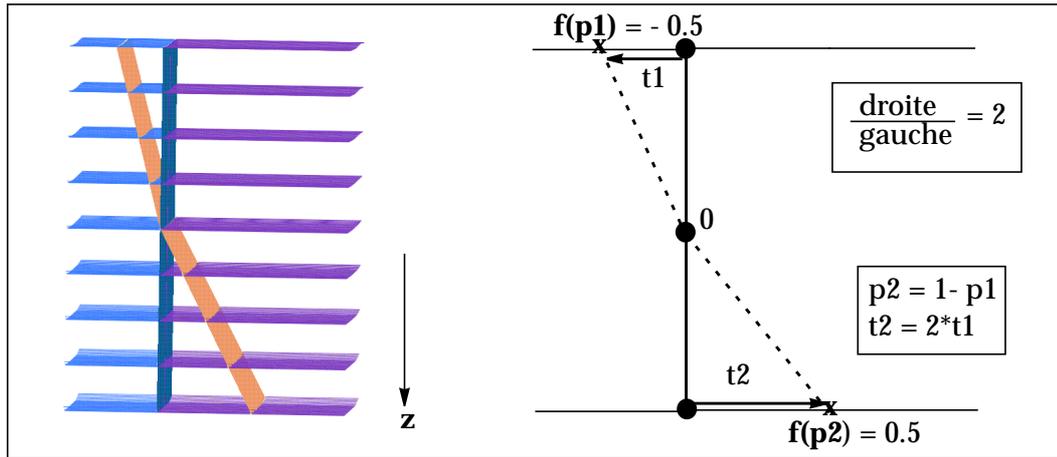


Figure 6-38 : Problème de cassure lorsque le rapport  $u_{gauche}/u_{droite}$  est différent de 1

Une solution consiste à calculer d'abord la valeur  $p_0$  correspondant à l'image du point de référence dans l'espace  $[0,1]$ , puis à utiliser ce point  $p_0$  comme centre de symétrie, au lieu de la valeur 0.5 actuelle [figure 6-39].

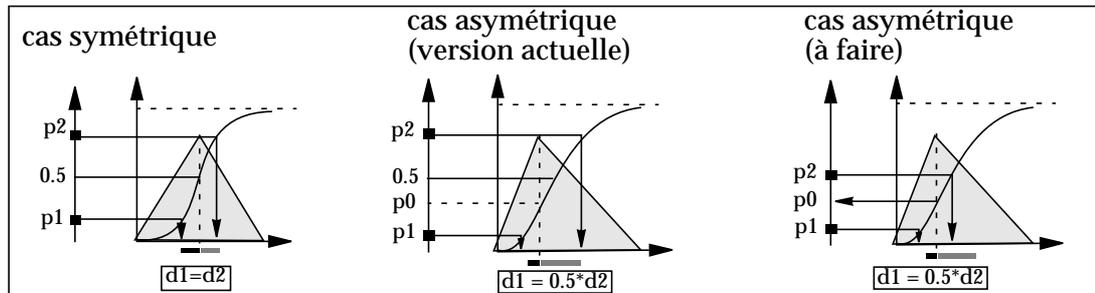


Figure 6-39 : Modification à apporter pour des lois triangulaires asymétriques sur une même verticale.

Si l'on peut considérer que tous les vecteurs *droite* (resp. *gauche*) sont égaux le long de la faille, il suffit de calculer  $p_0 = p(0)$  et de calculer  $p_1$  et  $p_2$  symétriques par rapport à ce  $p_0$ . On obtient donc:

$$p_2 = p_0 + (p_0 - p_1) \quad (0 < p_1 < p_0) \text{ si } 0 < p_0 < 0.5$$

$$p_1 = p_0 - (p_2 - p_0) \quad (p_0 < p_2 < 1) \text{ si } 0.5 < p_0 < 1.$$

Cette solution est rapide mais peut poser un problème théorique, puisque l'on est obligé de réduire la zone d'incertitude afin de conserver la symétrie et que la fonction aléatoire obtenue n'est plus une fonction uniforme dans l'intervalle  $[0,1]$  [figure 6-40].

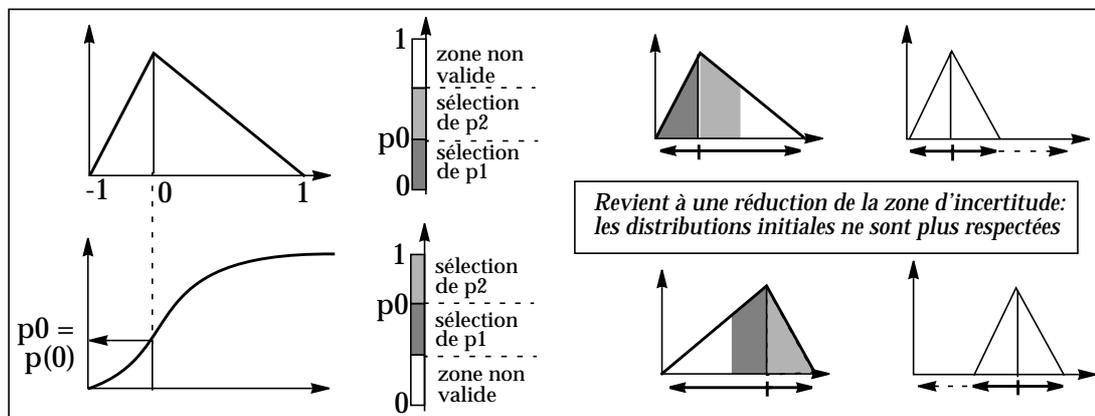


Figure 6-40 : Solution rapide pour tenir compte du cas  $u_{gauche}/u_{droite} \neq 1$

### b) Deuxième cas «pathologique»

Dans le cas particulier où les vecteurs sont identiques sur une même verticale, mais varient selon l'axe horizontal, certaines des méthodes de simulation n'ont pas de sens.

La figure suivante montre sur un cas un peu extrême les problèmes pouvant apparaître si l'on utilise la méthode correspondant au décalage latéral. Si les vecteurs d'incertitude sont asymétriques de part et d'autre de la faille, et que l'asymétrie est inversée entre les deux points extrêmes, il est impossible de proposer un simple décalage latéral à l'utilisateur (fonction aléatoire constante).

En effet, la méthode utilisant la même réalisation  $p$  pour toutes les  $cdf$  locales, il peut arriver que la géométrie simulée soit d'un côté de la faille de référence à l'une des extrémités, et de l'autre côté de la faille de référence au niveau de la deuxième extrémité [figure 6-41].

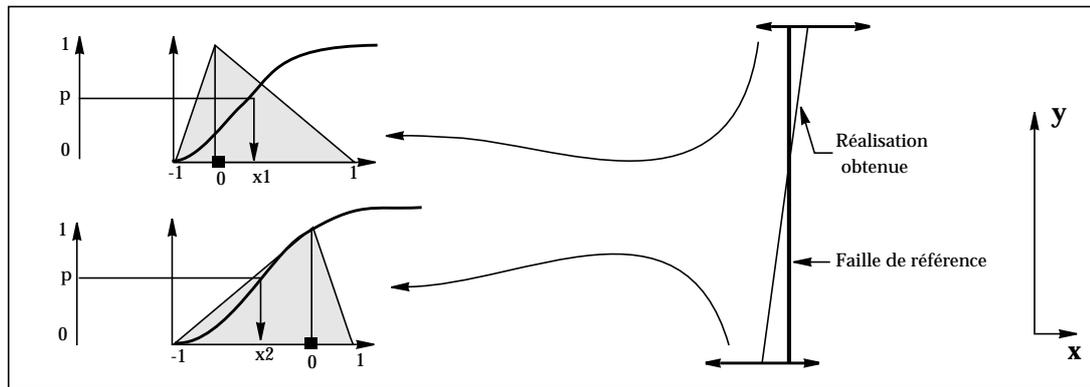


Figure 6-41 : Problème lors que les vecteurs d'incertitude varient selon l'axe horizontal de la faille.

L'une des solutions consiste à calculer la zone «valide» dans laquelle le décalage latéral est possible. Mais une fois encore, cette méthode, qui a le mérite de proposer une solution à l'utilisateur, est contestable du point de vue théorique, puisque la distribution finale ne correspond en aucune manière à la distribution correspondant aux vecteurs d'incertitude fournis au départ.

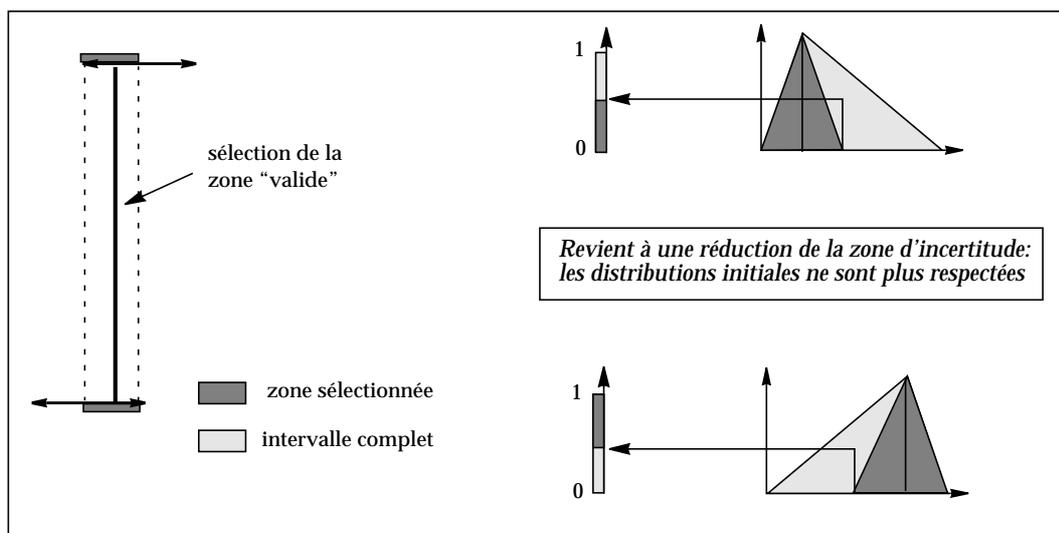


Figure 6-42 : Sélection de la zone valide dans le cas de vecteurs asymétriques.

### c) Troisième cas «pathologique»

Lorsque les vecteurs d'incertitude sont trop irréguliers le long de la faille, aucune des méthodes présentées n'est satisfaisante.

Par exemple, le cas extrême qui consiste à demander des modifications de la géométrie en conservant un point fixe (passage d'un puits pas exemple), en s'éloignant le moins possible de la géométrie de référence, tout en ayant des vecteurs d'incertitudes conséquents par ailleurs, est une demande impossible à satisfaire.

Sans aller jusqu'à cet extrême, si les vecteurs d'incertitude ne sont pas cohérents, il est difficile de proposer une solution acceptable dans de tels cas.

C'est pourquoi une attention particulière doit être apportée lors de l'estimation de ces vecteurs d'incertitude, pour que le champ de vecteurs obtenu ne conduise pas à ces cas pathologiques.



Figure 6-43 : Cas conduisant à une forme on géologique (très exagéré).

Une solution consisterait à travailler beaucoup plus sur les vecteurs d'incertitude et en fonction du type de simulation demandée, de sélectionner un intervalle "valide" pour chaque point (cette solution rejoint un peu aux solutions présentées [figure 6-40] et [figure 6-42]).

Elle consisterait à *sélectionner d'abord les parties des vecteurs d'incertitude valides* pour le type de simulation demandé, et de choisir ensuite ce sous-domaine comme nouvel espace de travail. Une fois ce domaine sélectionné, les méthodes précédentes pourraient être appliquées plus facilement sur les vecteurs d'incertitude "valides". Les *cdfs* seraient calculées en fonction de ces vecteurs d'incertitude "réduits", et on serait assuré d'avoir des fonctions aléatoires qui varient uniformément dans l'intervalle [0,1].

La difficulté de cette méthode est qu'un domaine réduit pour un type de simulation, n'est pas forcément valide pour un autre type de simulation. En fonction des vecteurs d'incertitudes fournis au départ, on se retrouve devant un très grand nombre de cas particuliers à traiter [figure 6-44].

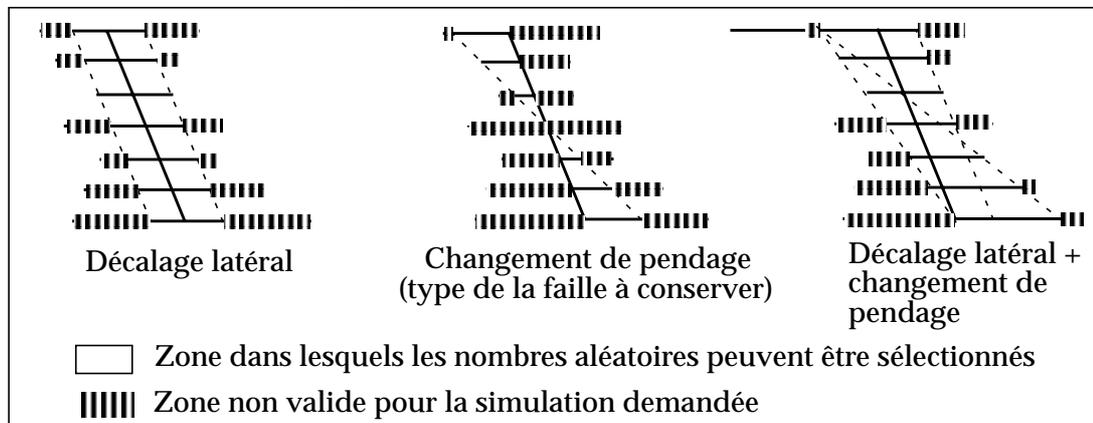


Figure 6-44 : Sélection de la partie valide des vecteurs d'incertitude, en fonction du type de simulation demandé.

### 3.4.5 Conclusion sur les méthodes de simulation

Le mixage des trois fonctions aléatoires de base est utilisable dans des cas relativement simples et courants actuellement.

Cependant, ces fonctions ne permettent pas encore de balayer l'ensemble des possibilités. Nous avons en effet choisi quelques restrictions afin d'éviter des géométries de failles que nous considérons comme non valides.

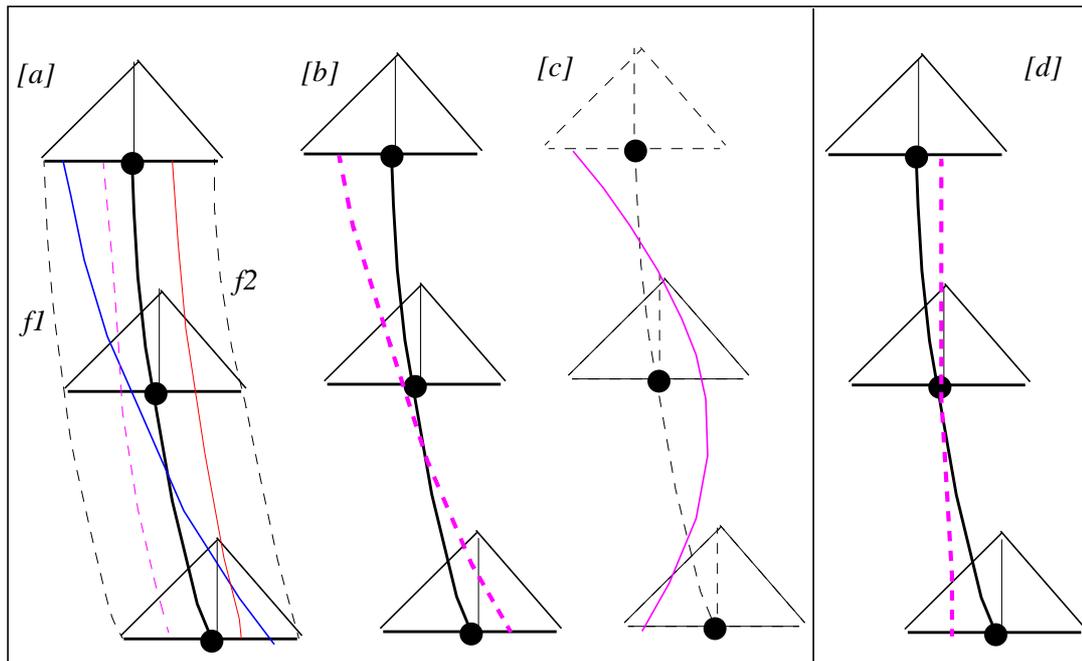
Ainsi, lorsque la combinaison (*géométrie horizontale + pendage*) est choisie, nous réduisons l'intervalle  $[0,1]$  dans lequel les nombres aléatoires sont sélectionnés à trois intervalles échelonnés le long de la faille:  $[p1]$ ,  $[p1t+w]$ ,  $[p1m, p1m+w]$  et  $[p1b, p1b+w]$ . Par défaut,  $p1t$ ,  $p1m$  et  $p1b$  sont choisis de telle sorte que  $p1t < p1m < p1b$  lorsque la faille est une faille normale.

Ceci nous permet de balayer toute la zone d'incertitude (voir figure 6-39 (a)):

- si  $p1t = p1m = p1b = 0$  et  $w=0$ , on obtient le cas extrême  $f1$
- si  $p1t = p1m = p1b = 1$  et  $w=0$ , on obtient le cas extrême  $f2$

Cette restriction autorise l'augmentation du pendage par rapport à la faille de référence (voir figure 6-39 (b)) et nous permet d'éviter les inversions de courbure le long d'une même verticale (voir figure 6-39 (c)). La condition c1 "monotonie de la fonction aléatoire le long d'une verticale" est bien vérifiée.

Cependant, elle est encore un peu forte, et écarte certaines solutions qui pourraient être considérées valides. Ainsi, la diminution du pendage est assez limitée avec ce type de simulation (voir figure 6-39 (d)).



**Figure 6-45 :** Les méthodes de simulations actuelles permettent d'explorer tout le volume [a]. Il est possible d'augmenter le pendage par rapport à la faille de référence [b]. La condition "fonction monotone suivant l'axe Z" est respectée et évite d'obtenir des inversions de courbure [c]. Par contre, les diminutions de pendage sont encore limitées à cause de conditions trop restrictives.

Certaines de ces restrictions sont donc laissées au choix de l'utilisateur: il peut décider de ne pas en tenir compte, au risque d'obtenir certaines géométries inacceptables du point de vue géologique. Ainsi, dans l'exemple précédent, le géologue peut autoriser une inversion de pendage pendant les simulations, et balayer un espace de solutions possibles plus important.

Dans ce chapitre, nous avons vu tous les aspects concernant la simulation de la géométrie d'une faille isolée: le stockage et l'interpolation des estimations de l'incertitude ainsi que les différentes techniques pour simuler de nouvelles géométries.

Le **stockage des incertitudes** permet la prise en compte de tous type de vecteurs. Ceux ci peuvent être estimés de différentes manières. Nous considérons qu'ils doivent être fournis par les géophysiciens, et travaillés afin d'être cohérents le long de la faille.

Les **méthodes de simulation** simples sont au point, la méthode de modification de la géométrie en carte demande plus de travail. Toutes les méthodes sont basées sur le même principe: l'attribution d'une *cdf* locale en chacun des points, et le calcul d'une *fonction aléatoire corrélée* dans l'espace pour inverser ces *cdf* locales. Lorsque le champ de vecteur est trop hétérogène, on ne peut pas le gérer à l'heure actuelle. De nouvelles fonctions aléatoires sont à l'étude, mais n'ont pas encore fourni assez de résultats pour être considérées comme robustes.

Même si les méthodes existantes marchent bien dans des cas simples (vecteurs d'incertitudes assez homogènes), il reste un gros travail de fond à faire afin de proposer soit des méthodes qui permettent de traiter les cas «pathologiques», soit qui informent l'utilisateur afin que ces derniers soient modifiés.

Nous avons commencé à réfléchir à ces solutions, mais elles sont loin d'être triviales compte tenu du nombre de cas particuliers à traiter.

De plus, il faut donc être très prudent avec ce genre de solution et prévoir un processus qui prévient l'utilisateur du fait qu'il demande un type de simulation contradictoire avec les vecteurs qu'il a fourni en entrée, et qu'il ne pourra en aucun cas retrouver une distribution des failles simulées correspondant aux vecteurs d'incertitudes donnés au départ.

Cependant, si l'on considère ce que représentent ces incertitudes et la manière dont elles sont calculées, on peut se rassurer en se disant que:

1. actuellement, un vecteur d'incertitude **global** est appliqué à la faille. On peut envisager que lors d'une étude plus précise, un horizon soit particulièrement étudié. Dans ce cas, les vecteurs d'incertitude affectés à cet horizon seront transférés aux autres génératrices, et on retrouve le cas particulier où l'étude peut se faire verticale par verticale.
2. les vecteurs d'incertitude ne doivent pas énormément varier le long de la faille, ou au moins varier de manière " lisse ". Même lorsqu'un puits donne une incertitude très faible en un point donné, cette incertitude faible doit se retrouver autour de la zone du puits et diminuer les vecteurs d'incertitude tout au long de la faille
3. si une autre fonction aléatoire est proposée plus tard pour résoudre ces cas plus complexes, la suite du processus de simulation n'est pas modifiée, du moment qu'en fin de simulation, on est capable d'affecter un vecteur de translation à chacun des points de la faille

Comme pour les parties précédentes, notre but étant de proposer des solutions simples pour chacune des parties de l'algorithme global permettant de simuler un réseau de faille, nous avons considéré que les méthodes actuelles étaient suffisantes pour poursuivre l'étude au niveaux du réservoir.

Ces méthodes de simulations, pourront être reprises plus tard, si le besoin s'en fait sentir.



---

---

# Chapitre 7 Construction de modèles volumiques

## Nécessité de préserver les contacts lors des simulations

---

---

Nous avons montré dans la deuxième partie comment simuler une nouvelle position pour chacun des points d'une *faille isolée*, compte tenu de vecteurs d'incertitude fournis par le géophysicien. Mais dans les études réelles, les géologues n'ont jamais à étudier une faille isolée.

A moins d'étudier une faille bordière particulière, dans la plupart des cas, il existe *plusieurs failles* reliées entre elles, et dans tous les cas, cette faille (ou ces failles) est (sont) reliée(s) à des *horizons*. Comme le montre la première partie de ce chapitre, il est indispensable de **rétablir tous les contacts** après chaque simulation, afin de réaliser une étude des variations de volumes lorsque la géométrie des failles est modifiée.

La deuxième partie de ce chapitre décrit comment cette gestion des contacts peut être faite lorsque le modèle est très simple (deux ou trois failles et autant d'horizons). Cette méthode a été utilisée au début de la thèse, pour tester la faisabilité de l'algorithme global sur un cas très simple. Mais elle est beaucoup trop lourde en gestion pour être utilisée sur des gros jeux de données.

Une autre méthodologie plus appropriée pour étudier les cas complexes sera présentée dans les deux derniers chapitres. Cette deuxième méthode utilise les fonctionnalités d'une nouvelle structure de données spécialement créée pour détecter et stocker les contacts dans un modèle structural: l'objet «STRUCTURALMODEL».

### 1. Modélisation du sous-sol: méthodes couramment utilisées

#### 1.1 Différents types de modèles utilisés pour représenter le sous-sol

##### 1.1.1 But de la modélisation

Les zones réservoirs étudiées par les compagnies pétrolières sont situées en profondeur, et ne sont connues la plupart du temps que par des *méthodes indirectes* (images sismiques par exemple). Afin de déterminer quelles sont les zones les plus intéressantes, il faut donc être capable de fournir un modèle géologique de la zone étudiée. Ce modèle ne constitue qu'une *interprétation*, c'est à dire qu'il constitue toujours une *description simplifiée de la réalité*.

Pour représenter cette partie de l'espace à 3 dimensions (ou espace 3D), on peut utiliser différentes représentations en deux dimensions (cartes ou sections verticales) ou, plus récemment, en 3 dimensions (groupe de surfaces, découpage de l'espace 3D à partir des surfaces précédentes, grilles régulières, grilles déformées qui suivent la stratigraphie).

La sous-section suivante expose les différents types de modèles, leurs avantages et leurs inconvénients. Ceci permet de justifier le choix que nous avons fait pour représenter le modèle du sous sol: le *modèle volumique continu*.

### 1.1.2 Modèles en deux dimensions

Depuis longtemps, les études géologiques ont été réalisées en deux dimensions, plus ou moins dessinées par les géologues structuraliste.

Dans cette catégorie de modélisation du sous-sol, on rencontre deux types de modèles [figure 7-1]:

- **les cartes** (représentation en «*vue de dessus*»), avec la troisième dimension représentée sous formes de courbes d'isovaleurs de l'altitude. Ce type de représentation est suffisant pour un œil exercé, mais n'est pas utilisable lorsque la surface est multivaluée dans la direction verticale (comme le sont souvent les lentilles de sel). De plus, devant l'augmentation du nombre de données à utiliser, il devient de plus en plus difficile et risqué de se contenter de cette représentation simple.
- **les coupes verticales** qui permettent de vérifier les relations entre les différents composants du modèle, mais selon un seul axe à la fois, ce qui limite la représentation que l'on peut avoir lorsque les zones sont trop complexes. Cependant, elles sont très utiles pour vérifier localement que l'interprétation des données est géologiquement correcte (en montrant moins d'information, elles permettent de se concentrer sur une zone délicate).

Avec les progrès informatiques, des méthodes de modélisation en trois dimensions sont apparues. Il est de plus en plus reconnu que la modélisation directement en trois dimensions apporte une plus grande précision, donne une meilleure vue d'ensemble et permet de déterminer plus d'incohérences, surtout lorsque les données sont de mauvaise qualité [135 - Hoffman],[107 - Lasseter], [109 - Orlic].

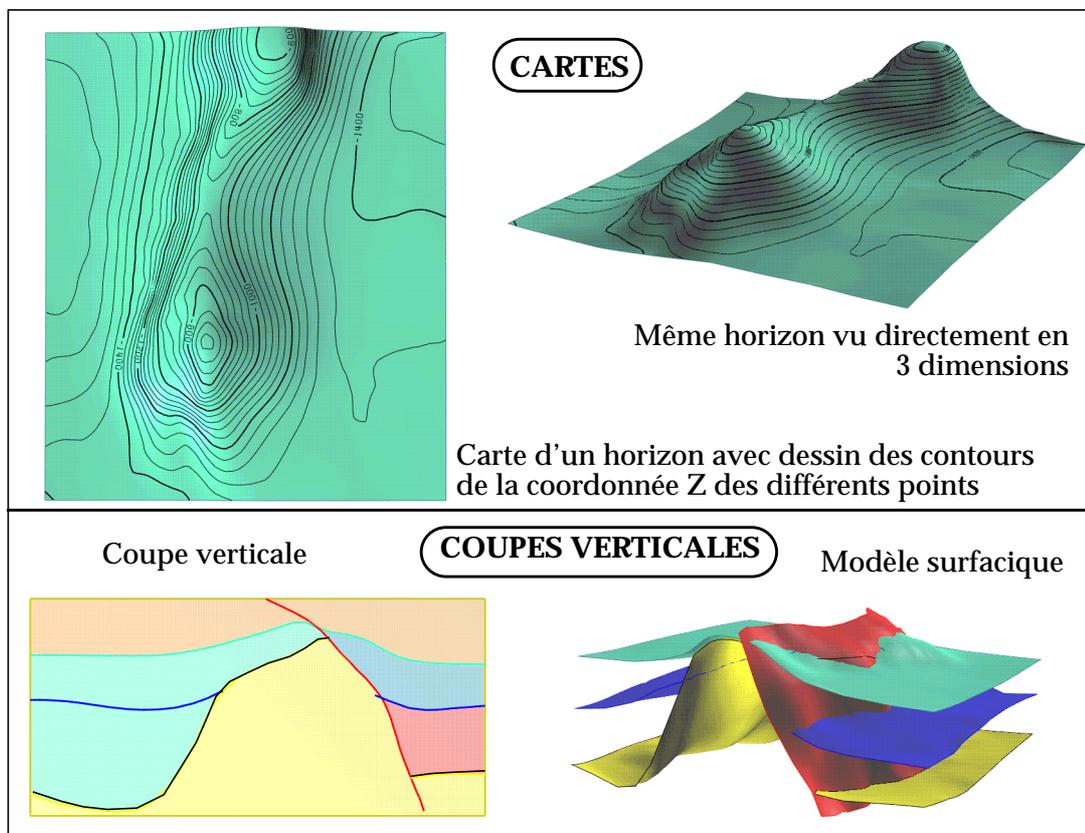


Figure 7-1 : Apport de la modélisation en trois dimensions

### 1.1.3 Modèles en trois dimensions

Les modèles en trois dimensions peuvent être classés en 3 catégories [109 - Orlic]:

- \* les **modèles surfaciques** constitués d'un ensemble de surfaces
- \* les **modèles volumiques continus**: qui séparent le domaine d'études en zones ayant des propriétés semblables.
- \* les **modèles volumiques discrets** qui permettent d'affecter des valeurs de propriétés (porosité et perméabilité notamment) en chaque nœud de la grille, de faire des simulations géostatistiques...

#### a) Modèles surfaciques

Lorsque l'on veut représenter une partie du sous-sol, la première étape consiste souvent à construire un ensemble de surfaces représentant les différents horizons et failles du modèle. Ces surfaces sont construites indépendamment les unes des autres dans un premier temps. Ensuite, un gros travail d'interprétation intervient afin d'assurer la cohérence de l'ensemble.

Il existe de nombreux types de représentations pour les surfaces [figure 7-2]:

- **les surfaces triangulées**: elles permettent d'avoir un maillage très souple, qui peut être optimisé localement en fonction de la complexité de la zone. Elles sont donc bien adaptées à la modélisation de surfaces complexes. C'est pourquoi elles sont très utilisées pour modéliser les objets naturels. Les principaux inconvénients de ces surfaces par rapport aux surfaces paramétriques sont que les opérations sont plus longues sur elles, qu'elles ne sont pas lisses au niveau des connections entre les triangles et qu'elles occupent plus de place mémoire.
- **les surfaces paramétriques**: ce sont des approximations des surfaces à l'aide de fonctions polynomiales. Elles permettent de générer des surfaces lisses, mais sont beaucoup moins flexibles que les surfaces triangulées. De plus il est plus difficile de représenter les discontinuités (failles) avec ce type de surface.
- **les grilles plus ou moins régulières**: elles sont souvent en deux dimensions, avec la troisième coordonnée stockée sous forme de propriété en chacun des points. Leur structure très simple permet d'avoir des algorithmes très rapides. Le revers de la médaille, est que ces surfaces sont difficilement adaptables dans des zones complexes où la géométrie varie beaucoup, et que pour compenser, il faut utiliser un grand nombre de cellules.

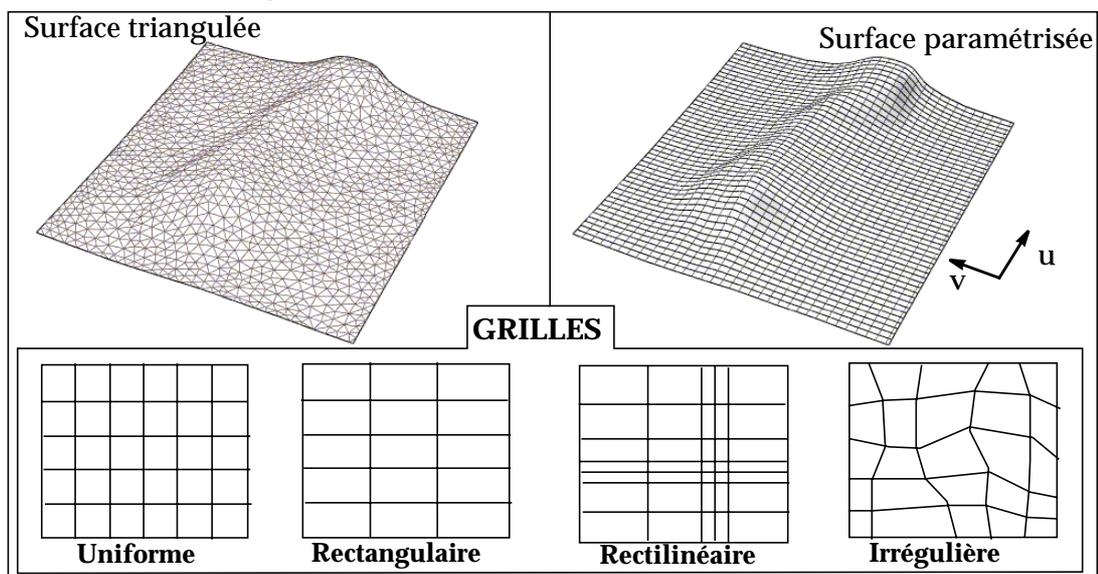


Figure 7-2 : Différents types de représentations pour les surfaces (d'après [109 - Orlic]).

Cette représentation surfacique est assez limitée puisqu'elle ne modélise que des objets individuels, sans stocker les relations qui existent entre eux. De plus, ce type de modélisation ne permet pas de séparer la zone d'étude en sous-domaines ayant les caractéristiques communes et pouvant être traités de la même manière.

C'est pourquoi des techniques de modélisation volumique sont utilisées en complément.

### b) Modèles volumiques continus (ou solides)

Ce paragraphe est très largement inspiré de la thèse de Nicolas Euler [106 - Euler 2]. Une revue beaucoup plus complète des techniques de modélisation volumique y est présentée et l'on pourra s'y référer pour plus de précisions.

Il existe de nombreuses méthodes pour construire un modèle volumique. Un premier groupe de techniques suppose que la région à modéliser peut être décomposée à l'aide de primitives géométriques comme des plans, des sphères, des cylindres, des axes de symétrie.

Ces méthodes ont deux principaux inconvénients:

- cette hypothèse de décomposition par des éléments simples est rarement applicable sur des objets naturels comme les réservoirs pétroliers
- dans ce type de modèle, il est très difficile de définir des relations explicites entre les différentes parties du modèle

C'est pourquoi en géologie, une autre technique est plus couramment employée: la **description des volumes par leurs éléments frontières**. Cette représentation, connue aussi sous le nom de *modèle de Weiler* [111 - Weiler] ou *B-Rep Model* («Boundary Representation»), est la plus souple, puisqu'elle ne dépend que des différentes surfaces qui composent les éléments frontière (horizons et failles principalement).

Ce type de modèle permet d'avoir une très bonne définition des relations topologiques entre les différents éléments du modèle. Il permet en outre de subdiviser l'espace en sous-volumes appelés *régions*. Il est ensuite possible de travailler de manière indépendante dans chacune des régions, ou dans un groupement de régions (par exemple deux régions correspondant à un même faciès géologique, mais séparées par une faille pourront être regroupées en une même «couche géologique»).

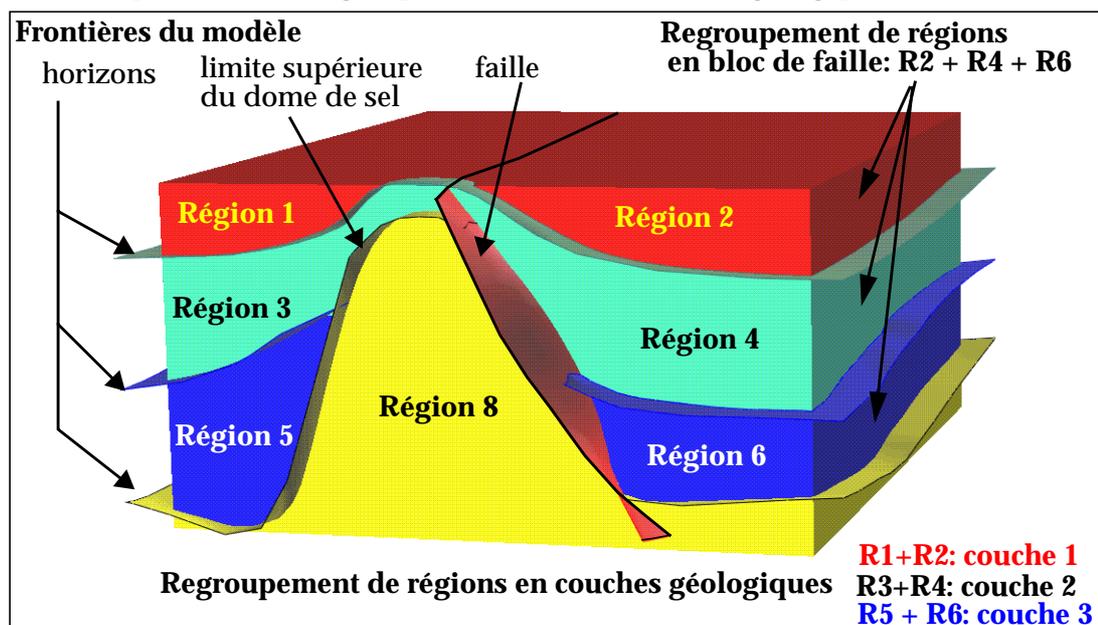


Figure 7-3 : Les différents composants du type de modèle volumique utilisé dans gOcad: le modèle de Weiler.

C'est avec ce type de modèle que nous avons décidé de travailler. En effet, il est **indispensable** lors de la phase d'exploration. Ses applications sont nombreuses, et parmi elles, on trouve:

- le calcul d'appartenance d'un point ou d'une courbe à une région de l'espace. Cette possibilité est utilisée par exemple pour vérifier que le tracé d'un puits (notamment d'un *puits dévié*) passe bien dans les zones qui ont été sélectionnées.
- la vérification de la cohérence des différents composants. En effet, comme nous l'avons vu dans les chapitres précédents, les données permettant de construire le modèle géologique (horizons et failles) sont souvent imprécises. La construction d'un modèle volumique continu permet de vérifier que ces surfaces modélisées individuellement forment un ensemble cohérent [figure 7-4].
- des propriétés peuvent être attachées dans les différentes régions du modèle, de manière indépendante. Cette possibilité est beaucoup utilisée pour créer des modèles de vitesses sismiques. En effet, chaque région correspond généralement à un faciès particulier. Il est donc possible d'estimer une vitesse de propagation des ondes sismiques dans chacune des couches. Ce modèle de vitesse est ensuite utilisé pour passer du domaine "*temps*" au domaine "*profondeur*".
- enfin, ces modèles volumiques sont utilisés pour faire des calculs de volumes précis. D'après le théorème de Gauss-Ostrogradsky, il est possible de transformer une intégrale de volume en intégrale de surface qui donne un résultat très précis. La connaissance de l'aire des triangles constituant les différentes frontières du modèle est donc suffisante pour calculer le volume de façon précise.

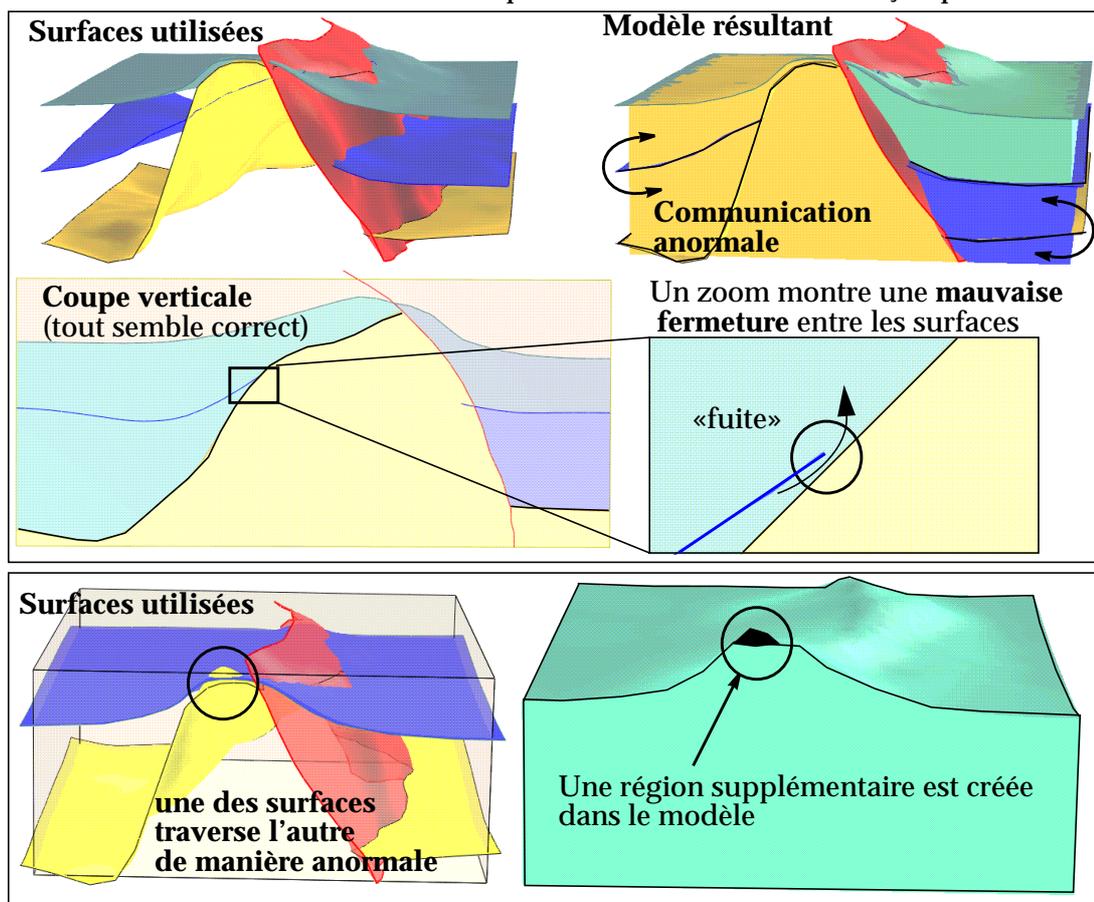


Figure 7-4 : Deux cas où la construction d'un modèle volumique continu permet de détecter des incohérences: la fuite entre deux régions lorsqu'il existe une communication anormale [en haut]; la création d'une région supplémentaire lorsque deux surfaces s'intersectent anormalement [en bas].

### c) Modèles volumiques discrets

Contrairement au modèle précédent, où la définition de propriétés se fait globalement dans une région donnée (constante, gradient vertical ou fonction plus complexe), les modèles volumiques discrets permettent d'attacher une valeur de propriété en chacun des nœuds du modèle.

Le problème de ce type de méthode, comme toutes les décompositions, est qu'il faut utiliser un *grand nombre de cellules* si l'on veut obtenir une bonne précision. Ceci est très gourmand en mémoire et plus long à utiliser.

Ces modèles discrets sont constitués d'un ensemble d'éléments de base (tétraèdres ou hexaèdres le plus souvent) [figure 7-5]:

- **grilles hexaédrales régulières (a)**: elles sont relativement faciles à construire, et la simplicité de leurs éléments de base permet de faire des calculs rapides pour affecter une valeur de propriété en chacun des nœuds. Cependant, il faut un très grand nombre de cellules pour décrire le modèle avec une précision acceptable. Cet inconvénient diminue beaucoup les avantages dus à la simplicité de l'élément de base. De plus, même avec un grand nombre de cellules, elles constituent toujours une approximation du découpage de l'espace, ce qui rend le calcul précis des volumes impossible.
- **grilles hexaédrales à éléments de base variables (b)**: ces grilles sont plus précises, puisqu'elles peuvent suivre la stratigraphie (d'où une autre appellation: *grilles stratigraphiques*). Ces grilles sont particulièrement adaptées pour faire des simulations de propriétés comme la porosité et la perméabilité, puisqu'elles représentent bien la géométrie du réservoir, tout en étant discrétisées. Pour cette raison, ces grilles font le lien entre l'étude géométrique du réservoir, et la partie qui consiste à estimer les direction d'écoulement des fluides à l'aide de simulateurs d'écoulement. Ces grilles sont déjà utilisées dans le logiciel JACTA et c'est pourquoi il a été décidé, au début de la thèse, que la représentation volumique du modèle après chaque simulation se ferait avec ces grilles stratigraphiques. Cependant, autant de telles grilles sont faciles à construire à partir d'un ensemble d'horizons, autant l'introduction de failles rend le calcul de ces grilles très difficile (apparition de *cellules dégénérées* par exemple). De nombreux progrès ont été réalisés par d'autres personnes de l'équipe sur la construction de grilles faillées, mais les résultats sont encore inutilisables pour réaliser de nombreuses simulations de la géométrie du réservoir. C'est pourquoi nous avons choisi de travailler avec des modèles volumiques continus, qui sont plus simples à utiliser et qui sont suffisants pour faire des calculs de volumes.
- **grilles à cellules tétraédrales (c)**: ce type de grilles est beaucoup utilisé pour réaliser des études de tracé de rayons (simulation du passage des rayons sismiques dans le modèle) Ce type d'opération est réalisé lorsque l'on veut vérifier la validité d'un modèle: une image sismique synthétique est réalisée à partir du trajet des rayons sismiques et comparée avec les vraies données sismiques. Malheureusement, ces grilles, sont beaucoup moins adaptées aux simulations de propriétés par les méthodes géostatistiques classiques (du moins pour l'instant). C'est pourquoi nous ne les avons pas utilisées dans notre travail.

Afin de trouver le meilleur compromis entre ce qui est réalisable actuellement, et les objectifs fixés au départ, c'est donc le **modèle volumique continu** qui a été utilisé pour représenter le modèle su sous-sol après chaque simulation. Il faut toutefois garder à l'esprit que le plus tôt possible, il faudra être capable de générer aussi des grilles stratigraphiques afin de rester compatible avec le logiciel JACTA.

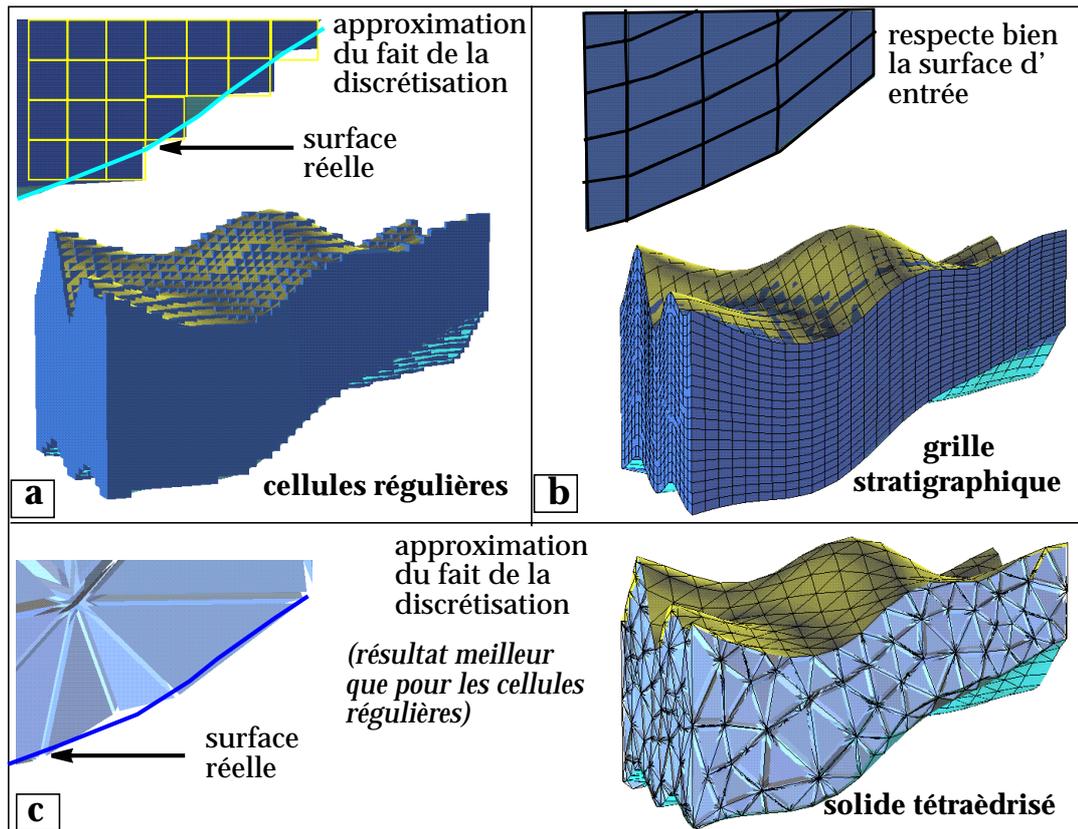


Figure 7-5 : Exemples de modèles volumiques discrets (un zoom permet de vérifier ce qui se passe dans chacun des cas).

## 1.2 Méthode de construction classique de solides

### 1.2.1 Principe

#### a) Conditions à respecter

Pour pouvoir construire un tel modèle, il faut vérifier un certain nombre de conditions, définies pour la plupart par Requicha [110 - Requicha]:

- il faut être capable de *différencier l'intérieur de l'extérieur*. Pour cette raison, toutes les surfaces frontières doivent être orientées et fermées
- le modèle obtenu doit occuper une *partie finie de l'espace*
- il doit avoir une *forme invariante* quelles que soient sa position et son orientation

Les différents éléments qui le composent doivent eux aussi vérifier deux conditions importantes:

- **Non intersection:** les différents volumes n'ont le droit de s'intersecter qu'au niveau de leurs éléments frontière (ou faces), les faces n'ont le droit de s'intersecter qu'au niveau de leurs bords, les courbes n'ont le droit de s'intersecter qu'au niveau de leurs extrémités et les points doivent être distincts. Lorsque l'une des règles n'est pas respectée l'élément en cause doit être redécoupé.
- **Finitude:** les points doivent avoir une position finie dans l'espace, les courbes doivent avoir une longueur finie, les faces doivent avoir une aire finie, les volumes doivent être fermés et avoir un volume fini.

## b) Méthode de construction

La méthode de construction classique est simple en théorie.

La première étape consiste à orienter tous les triangles de la même manière sur une même surface. Ceci est relativement aisé en utilisant le modèle de Weiler [figure 7-6] (a). Pour plus d'informations, on pourra se reporter à [106 - Euler 2].

La deuxième étape consiste à étendre toutes les surfaces du modèle afin d'être sûr qu'elles s'intersectent (pour éviter les problèmes de "fuite" comme le montre la [figure 7-4]). Comme nous le verrons dans le paragraphe suivant, cette étape peut poser quelques problèmes, surtout au niveau des contacts horizon-faille [figure 7-6] (b).

Une fois cette opération réalisée, la troisième étape consiste à calculer toutes les lignes d'intersection entre les différentes surfaces du modèle. Cette opération de découpage est réalisée par l'intermédiaire de calculs d'intersection sur les éléments de base comme les points, les segments et les triangles, et en modifiant localement les maillages des deux surfaces qui s'intersectent afin qu'elles partagent exactement les mêmes points au niveau de l'intersection. Les surfaces sont séparées en deux parties distinctes au niveau de chaque ligne d'intersection [figure 7-6] (c).

La dernière étape consiste à construire le modèle volumique. Chaque région du modèle est créée à partir d'un ensemble de faces orientées qui respectent la close de non intersection grâce au découpage précédent.

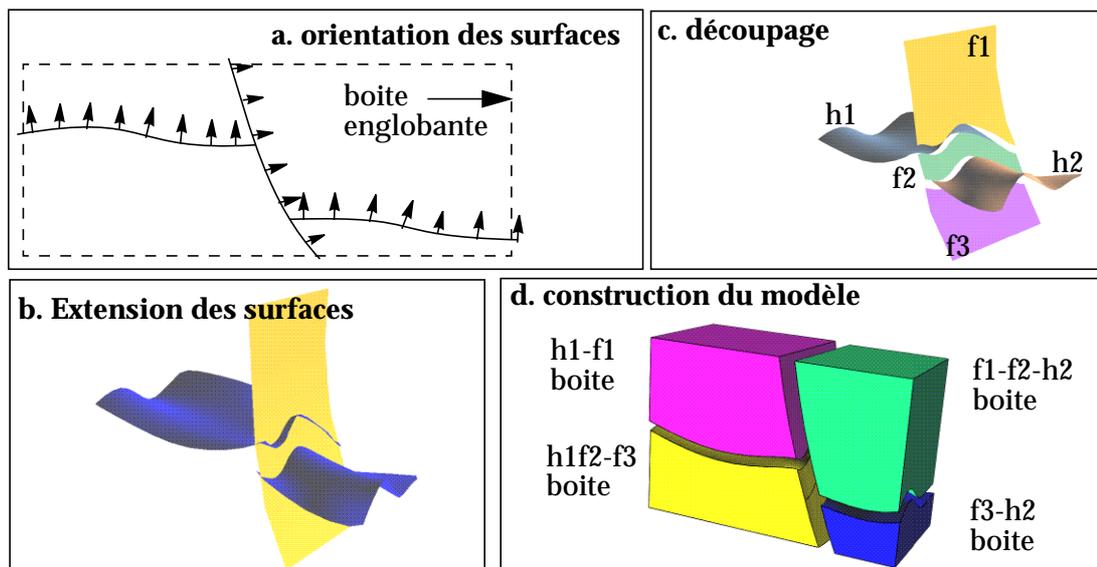


Figure 7-6 : Les grandes étapes de la construction d'un modèle volumique continu.

Au démarrage de la thèse, cette méthode classique de construction de modèle volumique posait de nombreux problèmes, dus essentiellement à des imprécisions numériques. Grâce au travail réalisé par Nicolas Euler, la plupart de ces problèmes ont été résolus et la construction de ce type de modèle est beaucoup plus sûre.

### 1.2.2 Problèmes de la méthode initiale

#### a) Problèmes dus à la phase d'extension

La première étape qui consiste à étendre toutes les surfaces latéralement, afin d'être sûr d'obtenir une ligne d'intersection ne peut pas être utilisée sur toutes les surfaces. Dans certaines configurations (faille se terminant au milieu d'un horizon par exemple), cette extension peut poser des problèmes [figure 7-7]. Il faut parfois utiliser une autre technique pour calculer la ligne d'intersection.

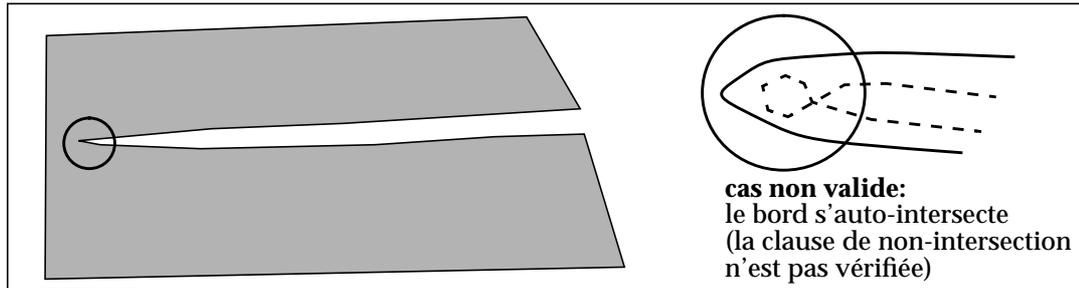


Figure 7-7 : Formation de “nœuds papillon” lors de l’extension de certains bords (d’après [106 - Euler 2]).

### b) Problème rencontrés lors du calcul de la ligne d’intersection

Cette méthode pose de nombreux problèmes, essentiellement dus à des *imprécisions de calcul* dans l’espace arithmétique flottant (problèmes de débordement ou de perte de chiffres significatifs).

Le calcul d’intersection entre deux surfaces dans l’espace à trois dimensions est une opération très complexe. La difficulté d’une opération de découpage réside dans le juste équilibre et le respect de trois critères: la *robustesse* (être sûr que toutes les intersections ont été détectées), la *précision* (les intersections calculées correspondent-elles vraiment à la réalité), et la *rapidité* de l’algorithme.

Si l’on cherche une précision “absolue”, on obtient des temps de calcul qui sont inacceptables dans un cadre industriel.

Par contre, si la précision n’est pas assez grande, les problèmes numériques résultants font que certaines intersections ne sont pas détectées, provoquant des “fuites” entre les deux surfaces en contact. Ce phénomène est encore plus accentué lorsque la densité du maillage est très différente entre les deux surfaces, ou lorsque l’une des deux surfaces présente une forte courbure localement.

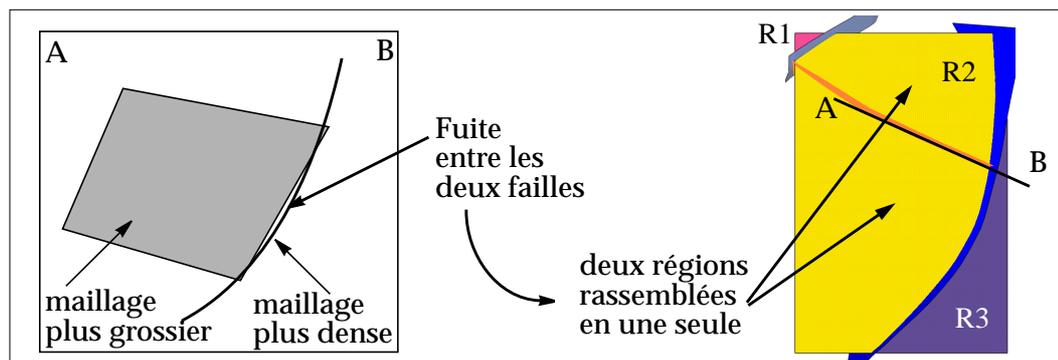


Figure 7-8 : Exemple où l’intersection n’est pas détectée. Ceci peut être dû à la forte courbure de la faille de droite, au fait que la faille de gauche n’a pas été étendue assez, ou à la différence de densité de maillage entre les deux surfaces.

Cette étape de découpage, ainsi que les problèmes qui lui sont liés sont détaillés dans [106 - Euler 2] et l’on pourra s’y reporter pour plus de précisions sur les problèmes rencontrés durant cette étape primordiale.

### c) Problèmes de maillage

Le dernier problème rencontré lors de la construction de ce type de modèle est dû au *corafinage local des surfaces* le long de la ligne d'intersection. En effet, la construction d'un modèle volumique valide ne peut se faire que si les frontières des surfaces et donc les bords contigus sont les mêmes.

Il faut donc que la ligne d'intersection calculée lors de l'étape précédente soit partagée par toutes les surfaces concernées.

Si des précautions ne sont pas prises, le maillage se dégrade très vite lorsqu'une surface de faille est découpée par plusieurs horizons, rendant cette surface inutilisable pour des calculs de propriétés par la suite (pendage par exemple) [figure 7-9].

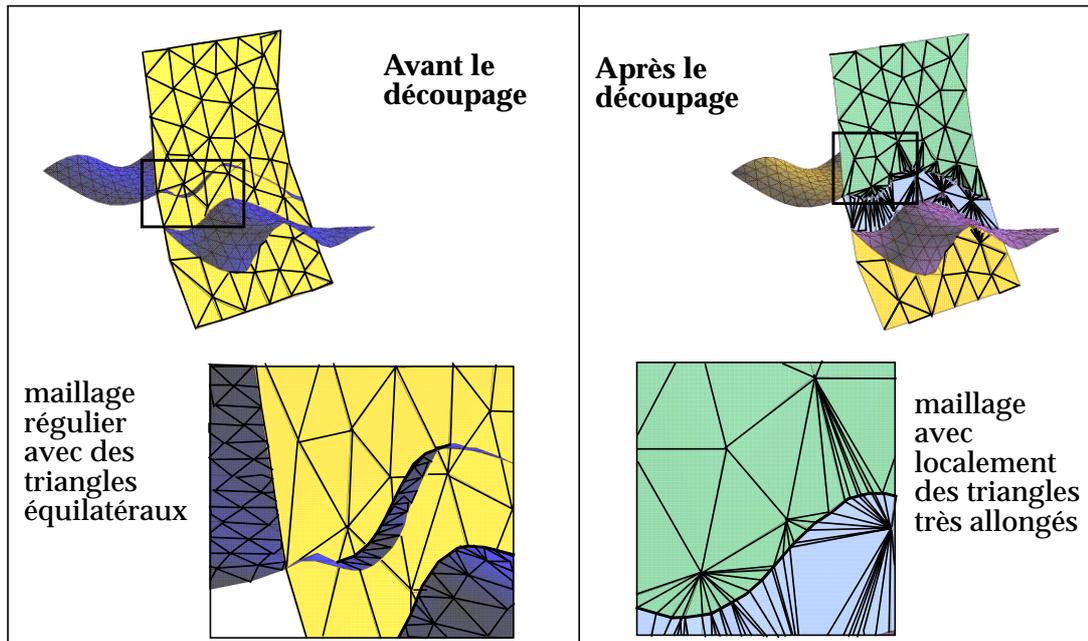


Figure 7-9 : Au fur et à mesure que les surfaces sont découpées, la qualité du maillage se dégrade.

### 1.2.3 Améliorations apportées

Une méthode originale a permis à Nicolas Euler de résoudre une grande partie des problèmes présentés ci-dessus [105 - Euler 1].

#### a) Découpage sous contrainte

Cette méthode ne fait aucune extension de surface, afin d'éviter les problèmes de nœuds papillon. Elle nécessite d'installer deux types de contraintes géométriques [figure 7-10]:

- la contrainte «Bord sur Surface» (ou *On Tsurf*): qui force le bord de l'horizon à se coller sur la faille lors de l'interpolation
- la contrainte «Coin de bord sur Bord» (ou *Border On Border*) qui force un point extrême du bord de l'horizon à coulisser le long du bord de faille lors de l'interpolation.

Ces deux contraintes sont utilisées pour calculer les lignes d'intersection entre les deux surfaces, ainsi que les points à insérer dans les deux maillages afin d'être sûr que le contact entre ces deux surfaces sera sans fuite. Mais même le contact semble correct après interpolation, il reste souvent des zones de fuite [figure 7-11].

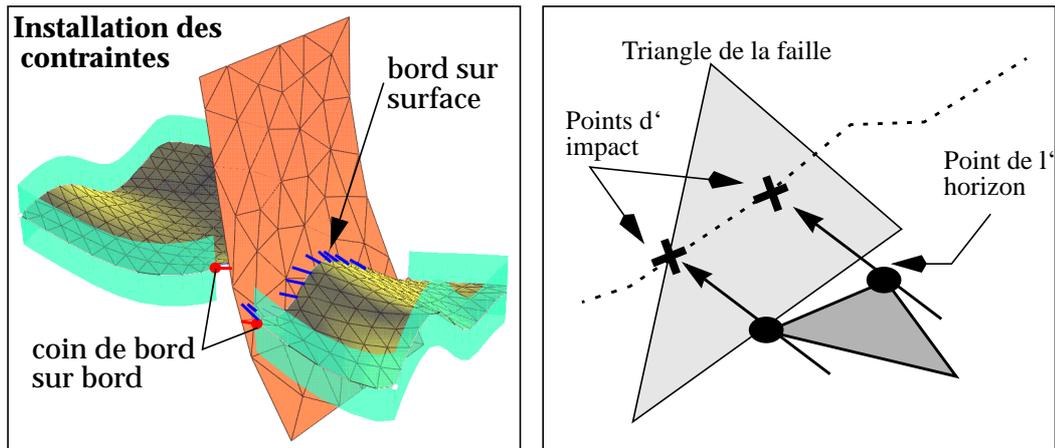


Figure 7-10 : Utilisation des contraintes installées pour calculer la ligne d'intersection: les points d'impacts trouvés par la méthode d'interpolation sont insérés dans la ligne de contact.

### b) Elimination des trous

Après l'opération précédente, le contact peut être correct du point de vue topologique (tous les points du bord de l'horizon se sont déplacés sur la surface de faille), mais il peut rester des zones de fuite à cause de la différence de discrétisation en triangles des deux surfaces.

En effet, si deux points d'impact consécutifs ( $p1$  et  $p2$ ) ne se trouvent pas sur le même triangle de la faille (ce qui arrive fréquemment), et que les deux triangles contigus ( $T1$  et  $T2$ ) forment un angle concave (resp. convexe), un trou sera créé (resp. un dépassement de la surface d'horizon à travers la surface de faille). Ces problèmes, souvent invisibles à l'œil nu, doivent absolument être résolus si l'on veut pouvoir construire le modèle volumique.

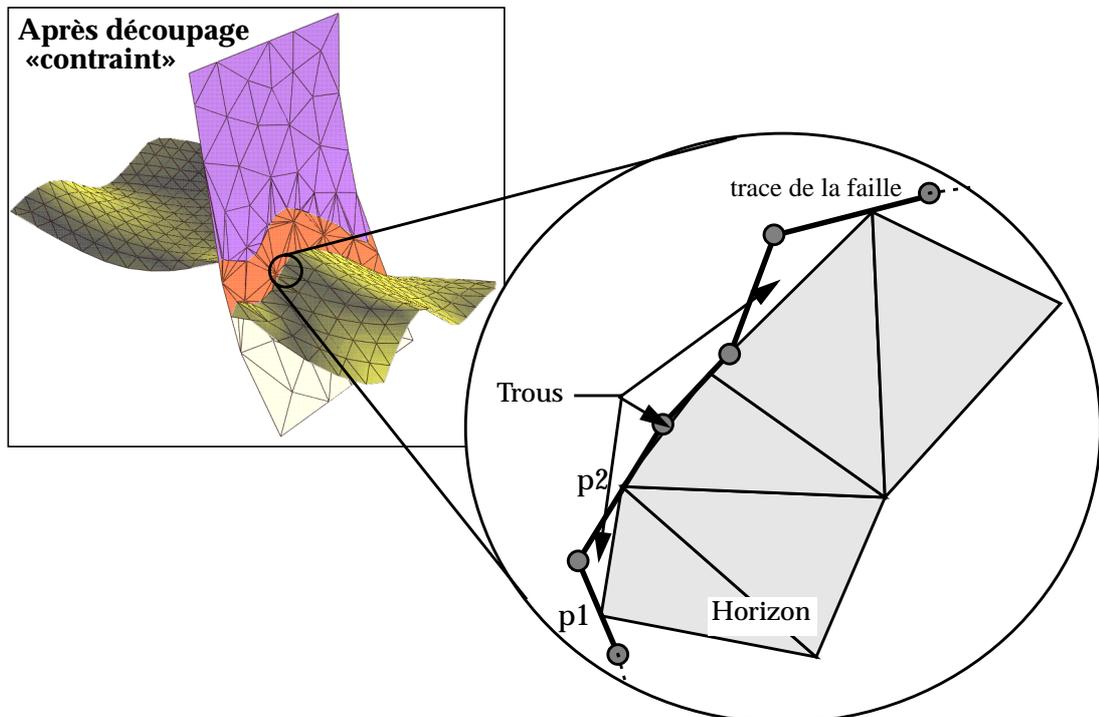


Figure 7-11 : Problème de création de trou (d'après [106 - Euler 2]).

Ce problème est résolu en calculant l'intersection entre l'arête commune entre les deux triangles T1 et T2 de la surface de faille, et la projection du segment formé par les deux points d'impact p1 et p2 sur la surface de faille. Ces points d'intersection sont ensuite ajoutés à la ligne de contact et assurent un contact parfait tant du point de vue topologique que du point de vue géométrique [figure 7-12].

Ainsi, l'information géométrique n'est utilisée qu'en fin d'algorithme de découpage, une fois que l'exploitation complète de l'information topologique a été utilisée.

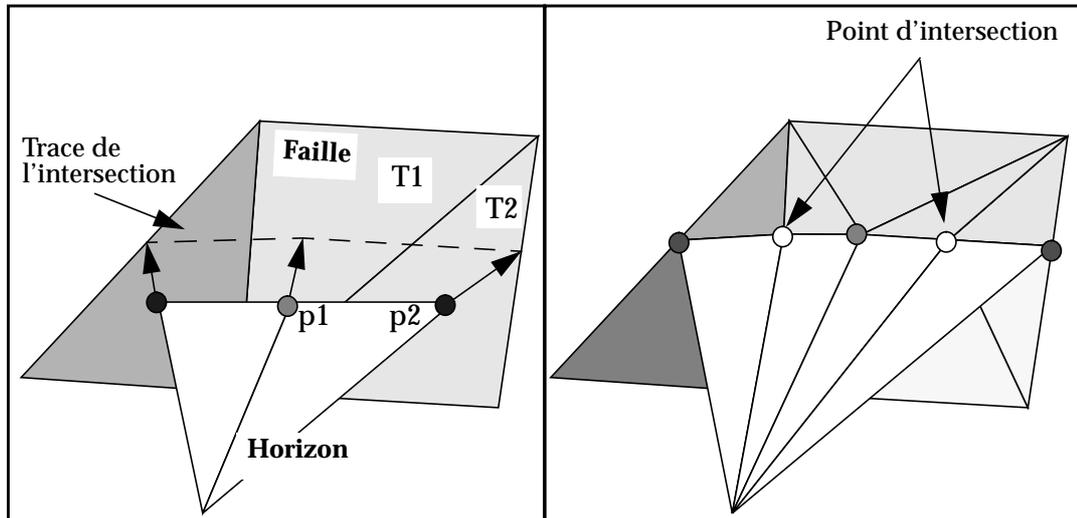


Figure 7-12 : Elimination des vides par projection (d'après [106 - Euler 2]).

### c) Amélioration du maillage

Les travaux réalisés ont en outre permis d'améliorer énormément le maillage, en faisant une analyse des points situés sur la ligne d'intersection, et en ne conservant que les points réellement significatifs pour les deux bords.

Cette opération consiste à trouver la discrétisation la moins fine possible (minimiser le nombre de segments nécessaires pour représenter au mieux les bords contigus) mais la plus proche de la réalité (minimiser l'erreur entre le bord original et le bord simplifié). Cet algorithme ne se contente pas de réduire le nombre de nœuds le long du bord, il permet aussi d'obtenir des triangles les plus équilatéraux possible, en écartant un peu les points situés trop près de la ligne d'intersection. Ainsi la plupart des triangles plats sont éliminés [figure 7-13].

Pour plus de précisions, on pourra se reporter à [106 - Euler 2].

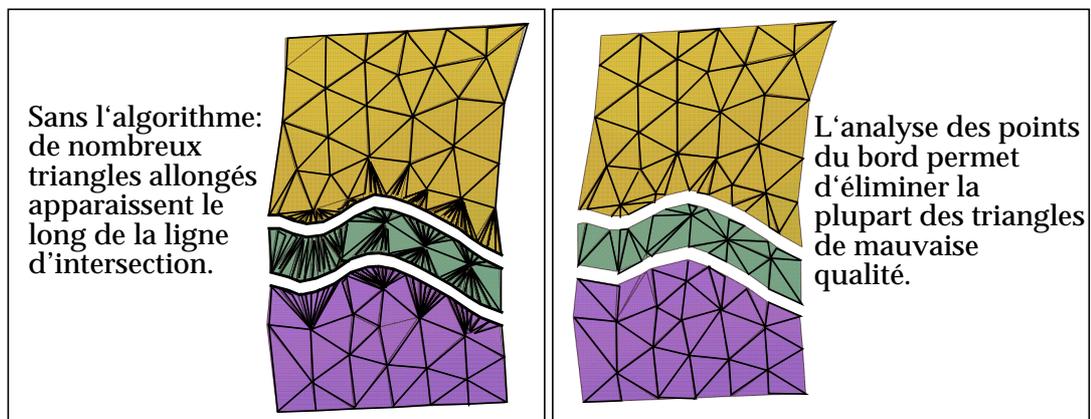


Figure 7-13 : Amélioration de la qualité du maillage

### 1.3 Méthode de construction à partir des coupes sériées

Comme cela a été souligné dans la section précédente, le plus gros problème auquel on est confronté lorsque l'on crée un modèle volumique continu, est le calcul de *lignes d'intersection* correctes non seulement du point de vue topologique, mais aussi du point de vue géométrique (contact «parfait»).

Durant sa thèse, Olivier Mariez [108 - Mariez], a mis au point une nouvelle méthode pour construire de tels solides à partir d'un ensemble de coupes géologiques. Cette méthode présente comme originalité de calculer d'abord toutes les lignes de contact affectant une même surface (faille ou horizon), et seulement ensuite de recalculer un maillage régulier incorporant toutes ces lignes de contact. Les nœuds des lignes de contact étant partagées par toutes les surfaces concernées par cette intersection, le contact géométrique est assuré. La triangulation étant recalculée tout à la fin, le maillage est beaucoup plus homogène que dans la méthode classique où des modifications sont apportées à une triangulation initiale, afin d'insérer les lignes de contact.

#### 1.3.1 Construction de surfaces à partir de courbes frontières

De la même manière que le modèle volumique constitue un découpage d'une portion de l'espace 3D en régions, une coupe géologique permet de définir un découpage d'une portion de l'espace 2D. Une structure spécifique, ayant des racines communes avec celle qui gère le modèle volumique décrit les relations entre les différentes lignes de la coupe. Cette même structure peut en outre servir à construire une surface triangulée à partir de toutes les lignes d'intersection où elle intervient.

Ce modèle en deux dimensions doit obéir à des règles d'orientation des lignes, afin de pouvoir regrouper toutes les courbes qui entourent une même région. C'est pourquoi une normale au plan global de la coupe permet d'orienter toutes les lignes d'une manière homogène. Une fois cette opération réalisée, il est possible de construire un morceau de surface triangulée dans chacune des régions. Ces morceaux étant créés à partir des lignes de contact, on est sûr que ces dernières sont insérées de manière acceptable dans la triangulation de la surface.

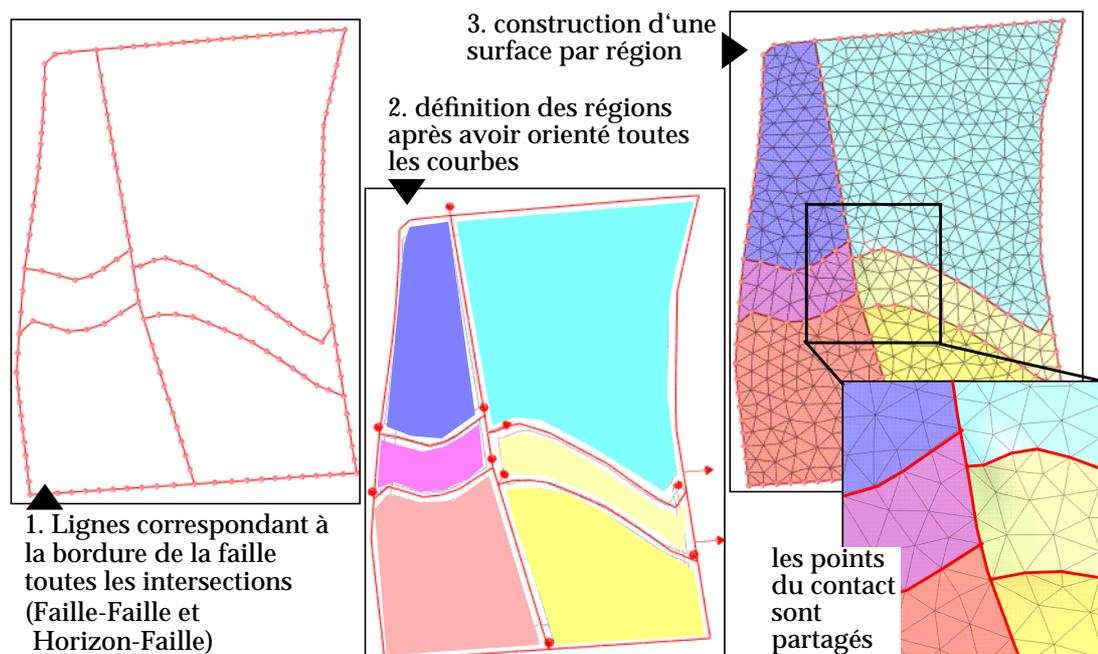


Figure 7-14 : Une structure qui permet de construire des surfaces à partir de leurs lignes d'intersection avec d'autres interfaces du modèle.

### 1.3.2 Principe de la construction

Les données utilisées pour construire un modèle de cette manière sont constituées par une série de coupes parallèles entre elles (coupes sériées). La seule hypothèse qui est faite est que les lignes correspondant à un même horizon sont corrélées d'une coupe à l'autre et portent le même nom [figure 7-15].

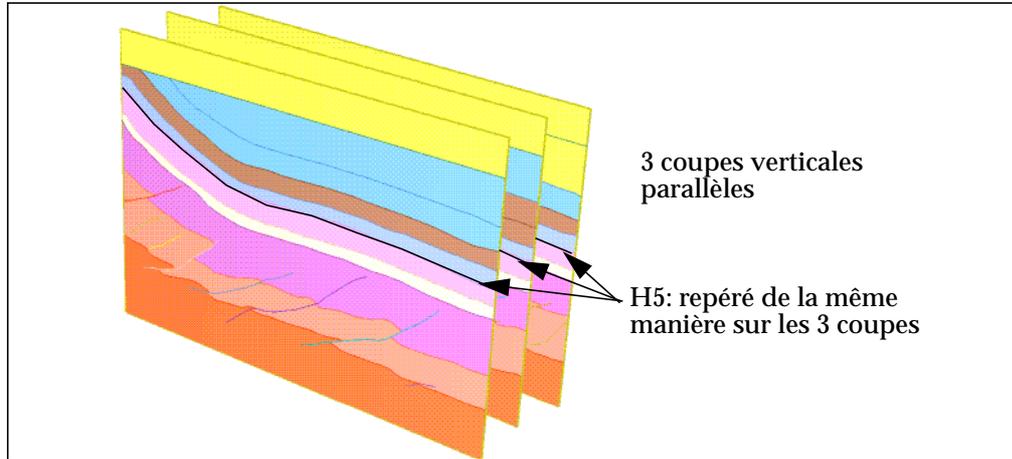


Figure 7-15 : Données utilisées: trois coupes géologiques parallèles.

Cette méthode de construction se déroule en plusieurs étapes:

1. **Appariement des failles d'une coupe à l'autre** en utilisant des critères de ressemblance entre leurs rejets respectifs, ainsi que les horizons qu'elles recoupent. Ces appariements sont stockés à l'aide de liens qui permettent de retrouver rapidement soit tous les liens correspondant à une même faille, soit tous les liens correspondant à une même horizon [figure 7-16]

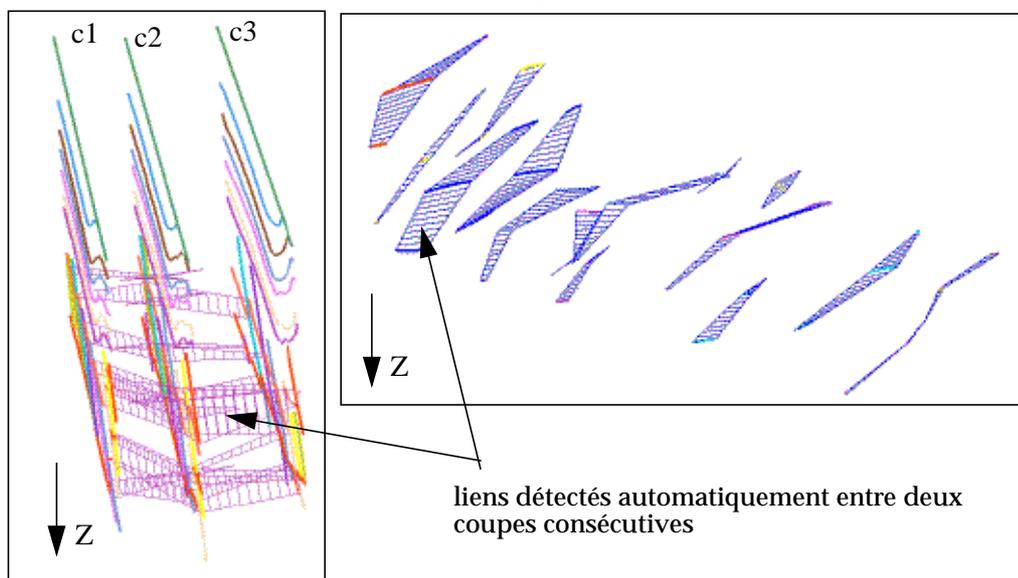


Figure 7-16 : Appariement des traces de failles d'une coupe à l'autre

2. **Création des squelettes de failles** regroupant toutes les courbes d'intersection les concernant, ainsi que la bordure de la faille (enveloppe convexe des précédentes).
3. **Création des squelettes d'horizons** regroupant tous les liens qui les traversent, ainsi que la bordure de l'horizon (enveloppe convexe des précédentes)

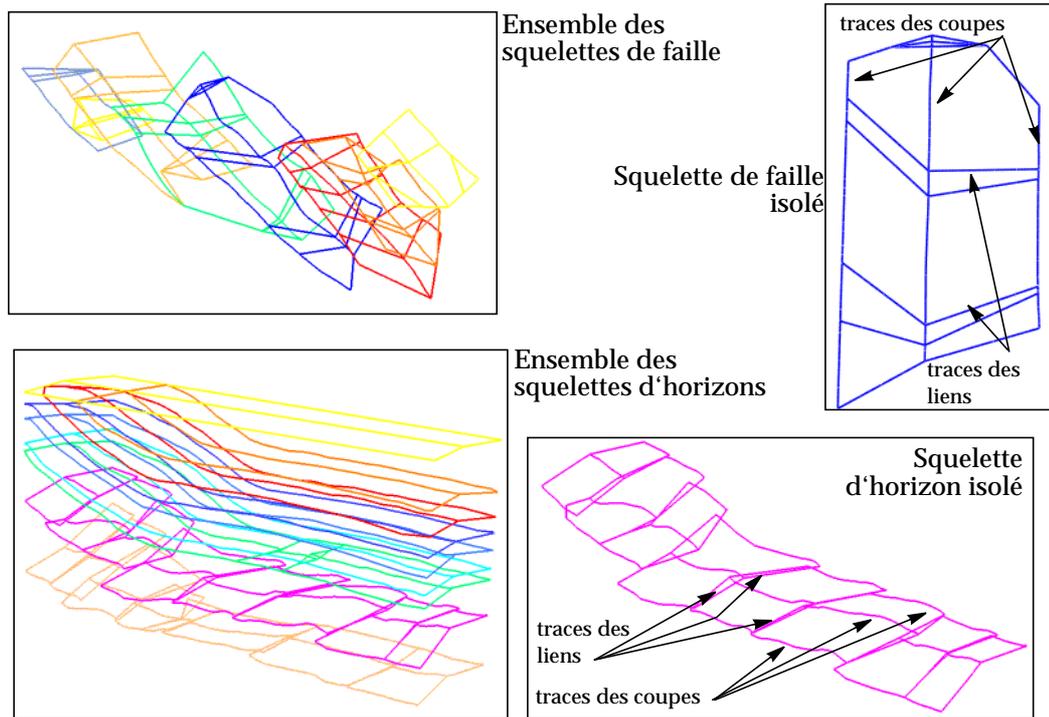


Figure 7-17 : Squelettes de failles et squelettes d'horizons.

4. **Création de surfaces «pré-découpées»** prenant compte de toutes les lignes d'intersection dans leur maillage. En fin d'étape, les morceaux qui se trouvent de part et d'autre des lignes de contact sont fusionnés (le contact est inclus dans le maillage et cela suffit à construire un modèle volumique fermé) [figure 7-18].

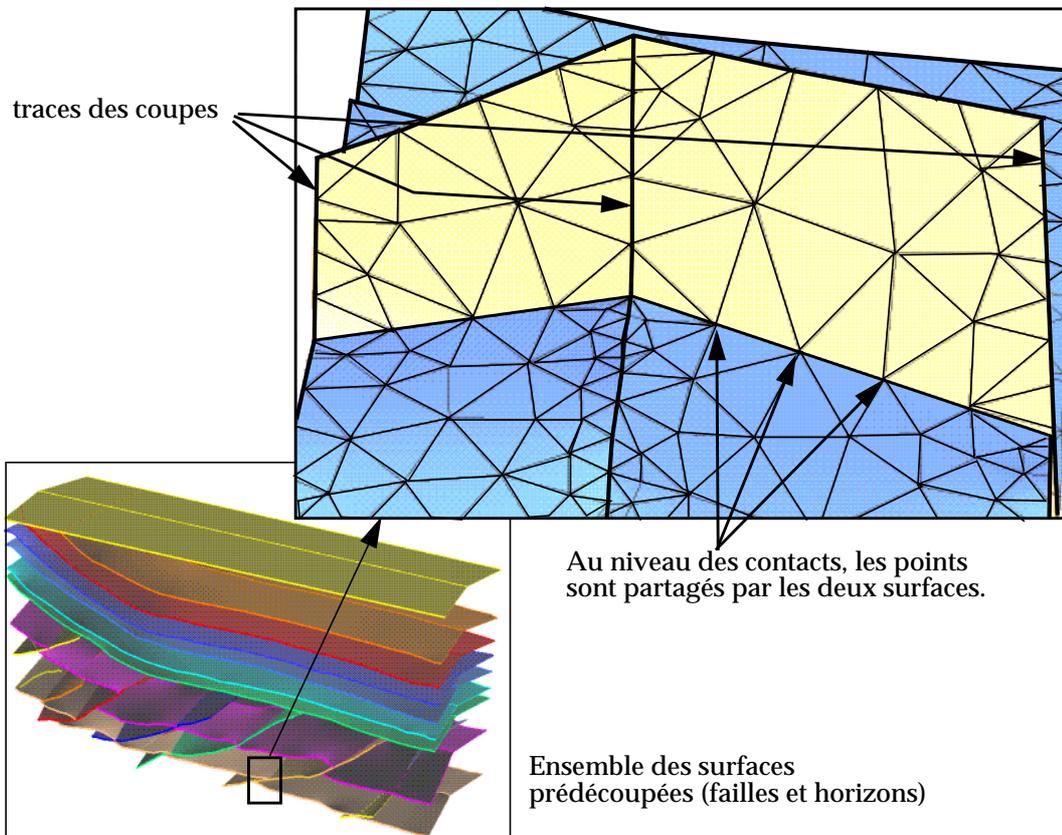


Figure 7-18 : Surfaces incorporant les lignes de contact dans leur maillage

5. **Création du modèle volumique** à partir de toutes ces surfaces «pré-découpées», après avoir défini la portion de l'espace à modéliser (boîte englobante) [figure 7-19].

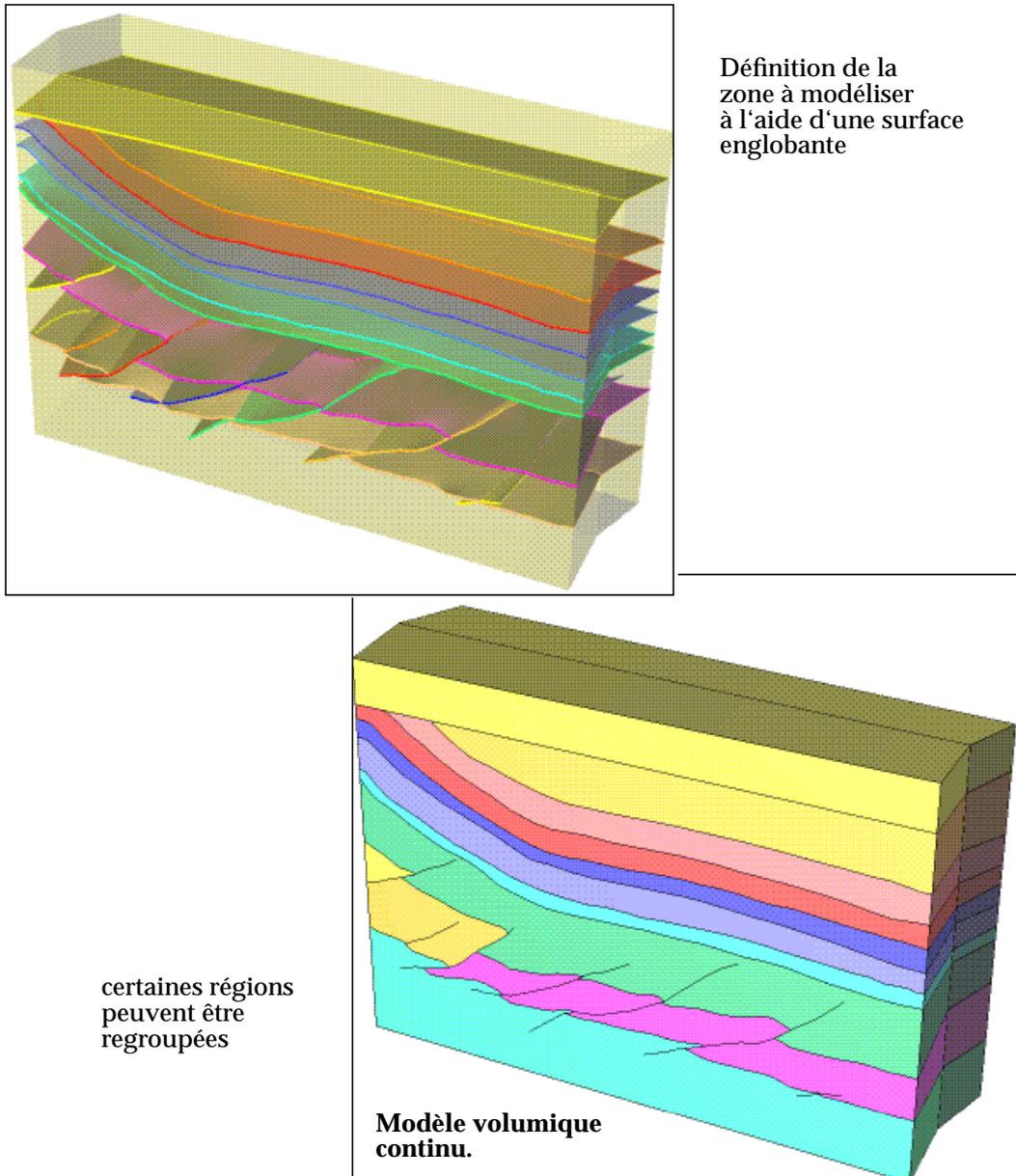


Figure 7-19 : Construction du modèle à partir des surfaces pré-découpées.

### 1.3.3 Bilan de cette méthode

#### a) **Avantage majeur: le contact parfait est assuré par le pré-découpage**

Toutes les surfaces partageant une ligne de contact partagent une copie de chaque nœud de la ligne de contact (utilisation des propriétés de la classe VERTEX: le même point géométrique est partagé par un ensemble d'objets). Ainsi on assure que le contact entre toutes ces surfaces est parfait autant du point de vue topologique que du point de vue géométrique.

Cette méthode, en évitant tout calcul géométrique pour détecter les lignes d'intersection, n'est donc pas sujette aux imprécisions numériques qui rendent la méthode classique difficile à utiliser dans certaines configurations complexes (même en utilisant le système de contraintes).

### b) Inconvénient: choix d'une direction privilégiée

Le principal inconvénient de cette méthode est qu'elle travaille selon une direction privilégiée (perpendiculaire à la série de coupes), et que certains contacts ne peuvent être détectés de cette manière (disparition d'une faille entre deux horizons par exemple). Même en utilisant une série de coupes perpendiculaires, on n'est pas sûr de résoudre tous les cas: il manque l'information fournie le long des bords d'horizons.

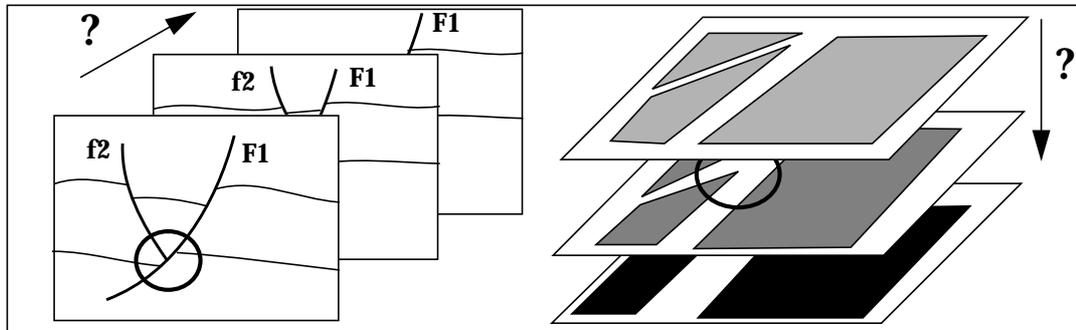


Figure 7-20 : Difficile de détecter d'une terminaison de faille lorsqu'une seule direction est privilégiée.

### c) Idée: utiliser la même technique à partir des horizons

Comme nous le verrons dans le chapitre suivant, la coopération avec Isabelle Duvinage a permis de mettre au point une méthode basée elle aussi sur le pré-découpage des surfaces de faille. Contrairement à la méthode présentée ici, l'axe principal n'est pas l'axe perpendiculaire à la série de coupes, mais l'axe vertical, puisque les appariements entre failles se font d'un horizon à l'autre, en utilisant l'information contenue au niveau des horizons eux-mêmes.

A terme, on peut imaginer un mixage de ces deux méthodes qui permettrait d'obtenir tous les contacts pouvant être détectés de manière automatique.

## 2. Connexions entre failles lors des simulations sur un modèle simple

Cependant, cette méthode consistant à calculer des contacts parfaits partagés par toutes les surfaces en contact n'était pas disponible au début de nos travaux. Nous avons donc utilisé la méthode de *découpage sous contraintes* qui nécessite l'installation de contraintes géométriques entre les failles et entre les failles et les horizons.

### 2.1 Principe de la contrainte "Fault On Fault" (FOF) et "Horizon On Fault" (HOF)

Comme cela a été présenté dans le chapitre 4, un mécanisme permet d'ajouter des contraintes sur les objets faille, équivalentes des contraintes "bord sur surface" déjà définies pour les surfaces triangulées. Cette contrainte force un bord de surface triangulée à se déplacer le long de la surface triangulée "contrôleur". Nous avons créé deux contraintes équivalentes pour l'objet faille, suivant que l'objet contrôlé est un horizon (HOF) ou un autre objet faille (FOF)

Dans le premier cas, 5 associations entrent en jeu: une association entre l'objet faille contrôleur et sa surface triangulée cachée, deux associations pour créer le lien entre la surface de faille cachée et la surface d'horizon, ainsi que deux associations pour représenter le lien entre l'objet faille et l'horizon ([figure 7-21] (a)).

Dans le second cas, 6 associations entrent en jeu: une association entre chaque objet faille et sa surface interne, deux associations pour représenter le lien entre les deux surfaces triangulées, ainsi que deux associations pour représenter le lien entre les deux objets faille ([figure 7-21] (b)).

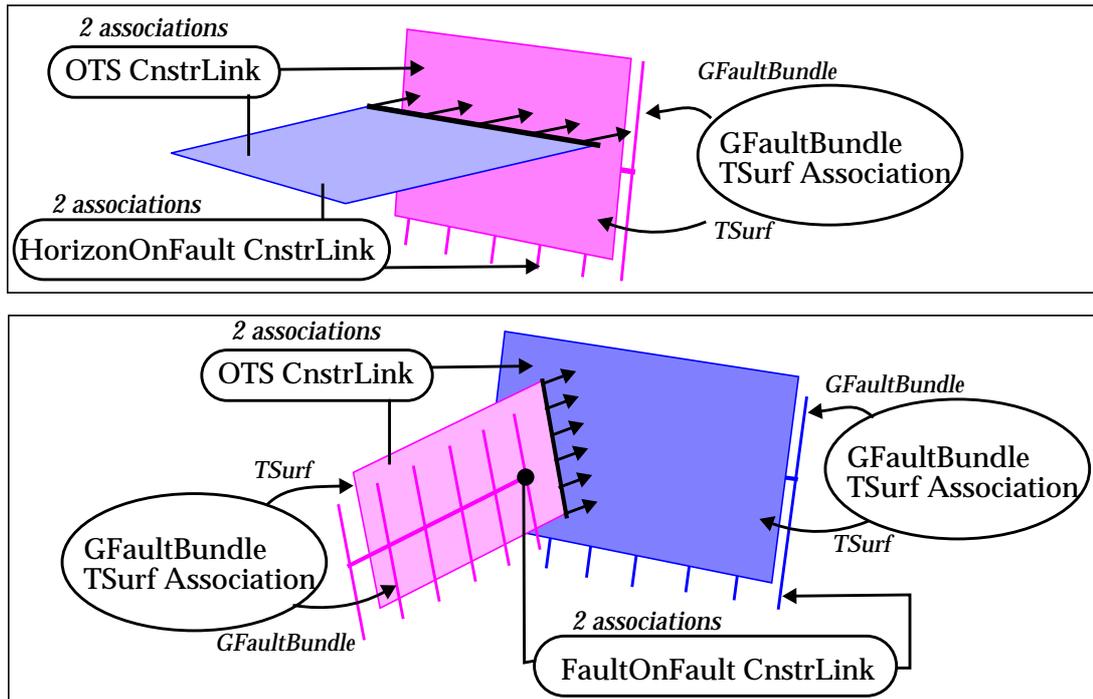


Figure 7-21 : L'utilisation de ces contraintes crée de nombreuses associations.

Une fois installées, ces contraintes permettent de fermer non seulement le réseau de failles, mais aussi les différents horizons sur les failles. L'interpolation se réalise d'abord sur les surfaces triangulées contrôlées. Elle permet de souder les endroits où la faille secondaire (resp. l'horizon) passe à travers la faille principale, ou au contraire, les endroits où il existe un espace vide entre les deux failles (resp. entre l'horizon et la faille). Une fois l'interpolation réalisée sur les surfaces triangulées, les nouvelles positions sont ensuite transférées sur les objets faille correspondants, par l'intermédiaire des associations ObjetFaille-SurfaceCachée ([figure 7-22]) et [figure 7-23].

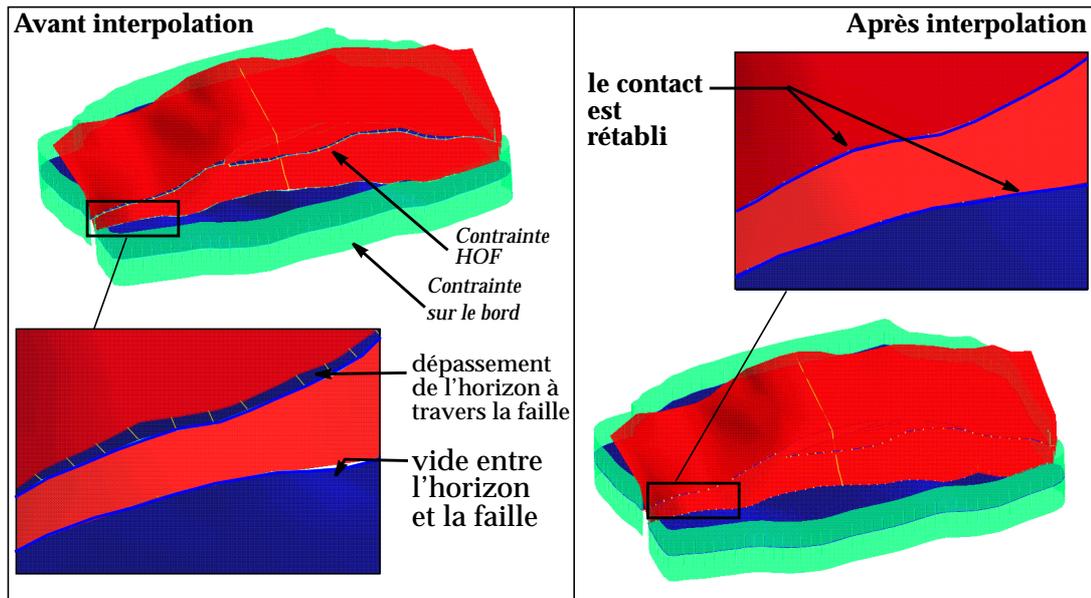


Figure 7-22 : Respect des contacts entre l'horizon et la faille en utilisant la contrainte HorizonOnFault (HOF), par l'intermédiaire de la surface interne de l'objet faille

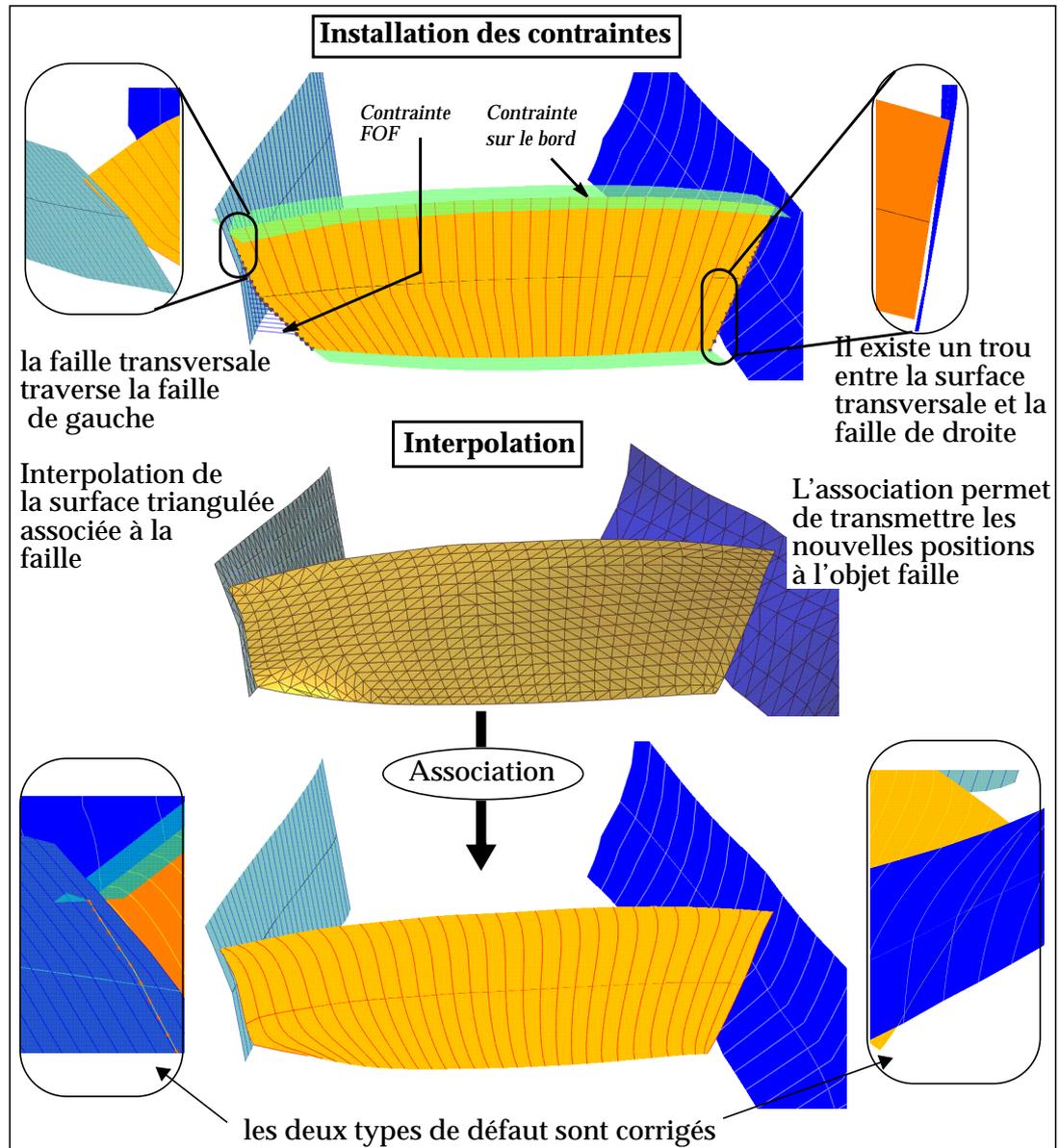


Figure 7-23 : Fermeture du réseau de failles en utilisant la contrainte Fault On Fault (FOF), par l'intermédiaire de la surface interne de l'objet faille.

## 2.2 Gestion des contraintes lors des simulations

Afin de simplifier le travail par la suite, je suppose que les contraintes reliant les différentes failles, ainsi que celles reliant les failles et les horizons doivent être pré-installées sous forme de contrainte "Fault On Fault" ou de contrainte "Horizon On Fault". De même, je suppose que les vecteurs d'incertitude doivent être estimés pour chacune des failles à simuler et déjà stockés sur les différents objets faille.

Comme on peut l'imaginer à partir de la [figure 7-21], même sur un modèle très simple contenant peu de failles et peu d'horizons, le nombre d'associations à gérer explose très rapidement.

Mais cette sous-section montre comment les choses se compliquent encore, du fait que l'on veut générer des réalisations équiprobables, et qu'il ne faut pas modifier les failles de référence.

## 2.2.1 Schéma général d'une simulation

Le principe des simulations est assez simple:

1. l'utilisateur sélectionne le *mode de simulation* pour chacune des failles et le nombre global de simulations à réaliser [figure 7-24]
2. pour chaque itération, les *nouvelles positions* des failles sont calculées à l'aide de la fonction aléatoire adéquate
3. après chaque itération, une *interpolation* est appliquée *sur chaque faille* pour fermer le réseau de failles
4. une fois le réseau de failles fermé, une *interpolation* est réalisée *sur chaque horizon*, afin de fermer le modèle

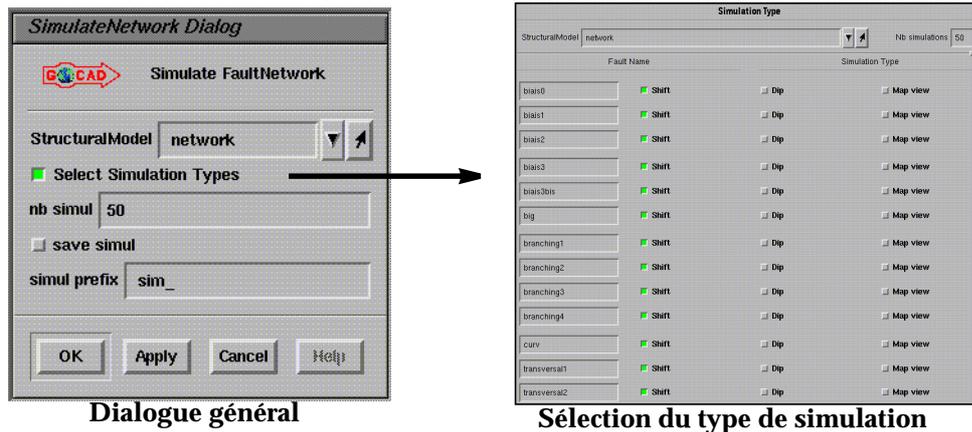


Figure 7-24 : Boite de dialogue permettant à l'utilisateur de sélectionner le mode de simulation pour chacune des failles du modèle.

## 2.2.2 Transfert des contraintes initiales

Comme nous voulons obtenir des simulations *indépendantes* les unes des autres, nous devons repartir de la **situation initiale** à chaque simulation. Nous ne pouvons donc pas simuler directement les failles de référence. En effet, les interpolations ne sont *pas réversibles*, et nous ne sommes pas sûrs de retrouver la configuration initiale en ré-interpolant en sens inverse.

En début de simulation, un gestionnaire de simulation par faille est créé (FAULTSIMULATOR). Afin d'obtenir des réalisations indépendantes, ce FAULTSIMULATOR crée une copie de la faille de référence dont il s'occupe. Puis, pour chaque contrainte installée sur la faille de référence (autant *FOF* que *HOF*) il crée la contrainte correspondante entre la copie de la faille et l'objet concerné. Les contraintes initiales sont ensuite désactivées afin de ne pas interférer avec les nouvelles contraintes durant la phase d'interpolation.

Cette opération est simple pour les contraintes reliant un bord d'horizon à un objet faille (il suffit de créer une nouvelle contrainte entre la copie de la faille, et le même bord d'horizon). Par contre, elles sont beaucoup plus complexes lorsqu'il s'agit de transférer une contrainte liant deux objets faille. Il faut en plus retrouver les correspondance entre le bord initialement contrôlé et le nouveau bord à contrôler.

Afin de diminuer un peu la complexité des simulations, et du nombre d'associations à créer, la copie de la faille de référence n'est donc pas créée sous forme d'objet faille, mais correspond à la copie exacte de la surface triangulée cachée dans l'objet faille. Ainsi, les contraintes sont directement créées entre deux surface triangulées (surface d'horizon et copie de surface de faille, ou deux copies de surfaces de failles). Il ne faut plus que deux associations par contrainte (au lieu de 5 voire 6 avec les objets faille) [figure 7-25].

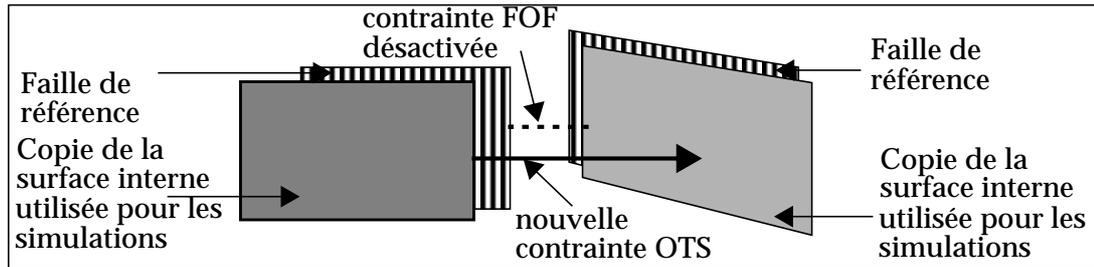


Figure 7-25 : Exemple de transfert pour une contrainte "Fault On Fault".

### 2.2.3 Description plus précise d'une simulation

L'algorithme de simulation est donc décomposé en trois phases:

- une phase d'**initialisation** durant laquelle les différents FAULTSIMULATORS sont créés. Ceux-ci se chargent de créer et stocker une surface triangulée correspondant à la faille de référence, de transférer les contraintes de cette faille de référence vers la copie, et de désactiver les contraintes initiales.
- Une fois que l'initialisation des nouvelles contraintes est réalisée, le processus de simulation peut commencer. Une **itération** peut être détaillée comme suit:
  1. le calcul des *nouvelles positions* est réalisé sur la faille de référence qui contient les vecteurs d'incertitude. En fin de simulation, chaque point stocke une position située dans l'intervalle  $[-1,1]$
  2. les vecteurs de *translation*  $t_k$  sont calculés et appliqués aux points de la copie de la faille (a). Si la position calculée (pos) est négative, le vecteur de translation est  $t_k = pos_k * right_k$  sinon  $t_k = pos_k * left_k$  (k représentant l'indice du point).
  3. l'*interpolation* pour fermer le réseau de failles est réalisée sur les copies des failles de référence (b). Ce sont les contraintes créées durant la phase d'initialisation qui sont utilisées à chaque boucle.
  4. l'*interpolation des horizons* suit les copies des failles de référence (c). Ici aussi, les contraintes utilisées lors de l'interpolation sont celles qui ont été copiées.

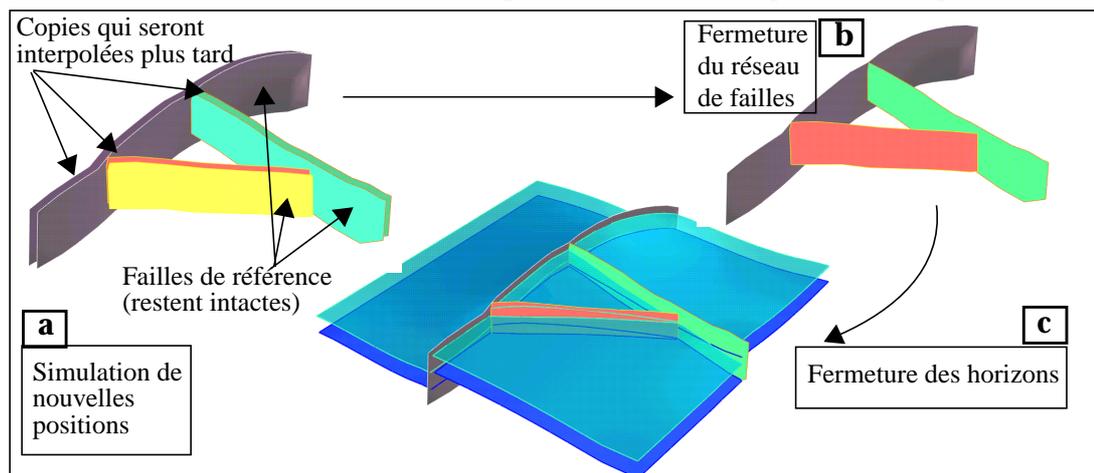


Figure 7-26 : Une simulation (modification puis mise à jour des contacts)

- la destruction de toutes les copies (surfaces triangulées et contraintes) et réactivation des contraintes désactivées, afin de retrouver l'état initial.

C'est cette utilisation des contraintes géométriques qui guidé l'utilisation d'un logiciel comme gOcad pour réaliser des simulations sur les failles. En effet, une fois installées, ces contraintes contiennent toute l'information nécessaire pour mettre à jour les lignes de contact entre les différentes surfaces du modèle.

### 2.3 Résultats

Sur un exemple très simple constitué d'une faille et de deux horizons, on peut vérifier qu'après chaque simulation, les contacts entre les horizons et la faille sont recalculés correctement [figure 7-27].

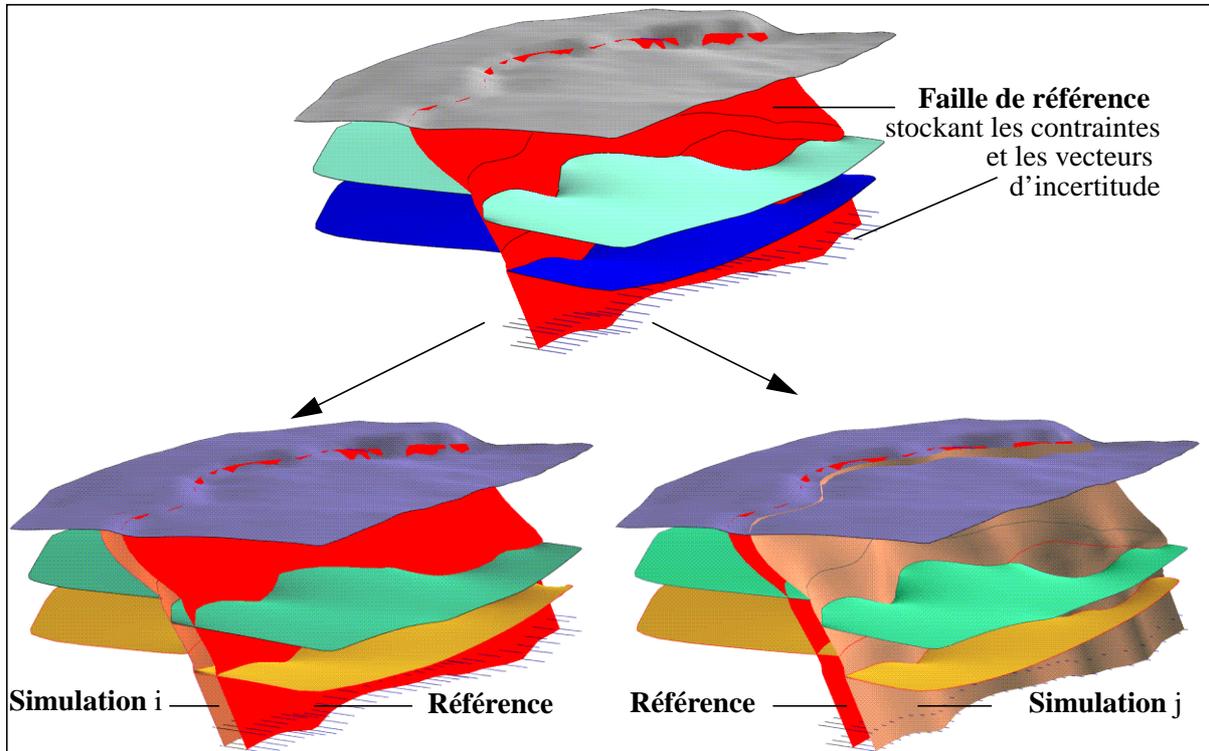


Figure 7-27 : Vérification du maintien des contacts sur deux simulations.

Comme cela a été montré, le fait d'obtenir des contacts bien soudés permet de construire des modèles volumiques. Le calcul des volumes de chacune des régions définies par les 3 surfaces est donc envisageable après chaque simulation.

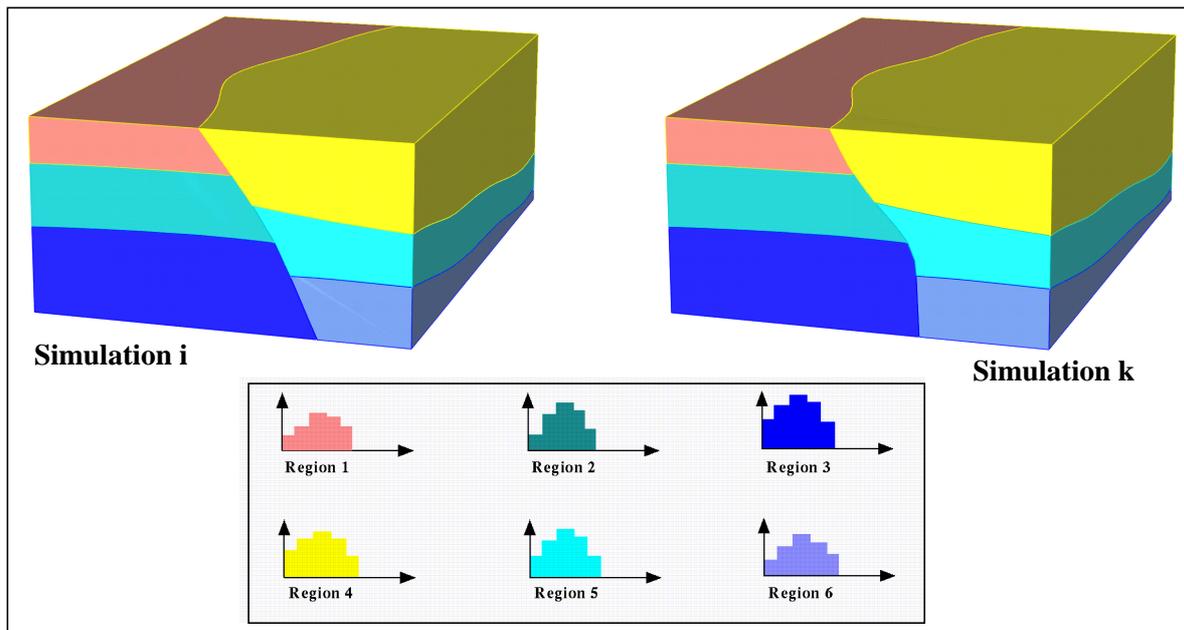
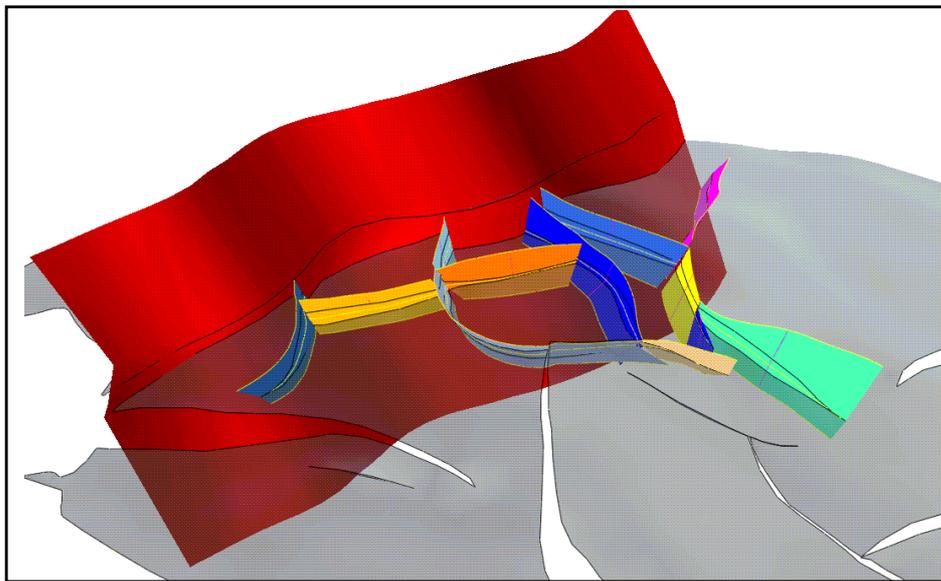


Figure 7-28 : Lorsque les contacts sont soudés, il est possible de construire le modèle surfacique, et donc de calculer les volumes de chacune des régions.

La gestion des contacts présentée ici est suffisante sur des exemples simples. Cependant, elle présente un certain nombre d'inconvénients:

- les contraintes disponibles sur les objets faille sont *beaucoup moins nombreuses* que sur les surfaces triangulées. Plutôt que de reporter toutes les contraintes existant sur les surfaces triangulées vers l'objet faille, nous pensons qu'il vaut mieux conserver un objet faille le plus léger possible pour réaliser ce pour quoi il a été conçu (les simulations de la géométrie grâce à une paramétrisation simple), et n'utiliser que des surfaces triangulées pour préserver les contacts (à l'aide des contraintes).
- dès que le nombre de failles augmente, cette méthode devient trop lourde à gérer. Le nombre de relations entre elles risque en effet d'augmenter et la recherche des contraintes à copier augmente d'autant.



**Figure 7-29 :** Exemple de modèle (encore assez simple) sur lequel les copies de surfaces et de contraintes deviennent trop lourdes à gérer de cette manière.

- Enfin, l'installation des contraintes avant le lancement des simulations n'est pas automatique et devient très pénible dès que le nombre de failles augmente. Si l'on est capable de les installer d'une manière automatique, cela rend le processus beaucoup plus facile pour l'utilisateur par la suite.

Même si cette partie de transfert des contraintes ne se produit qu'une fois pour toutes, avant de lancer les simulations, elle est lourde à mettre en place et le code résultant est difficile à maintenir.

C'est pourquoi, même si elle est plus récente et ne marche pas encore parfaitement, nous avons utilisé la méthode mise au point grâce à un objet spécialement conçu pour gérer les contacts: le STRUCTURAL MODEL. Les techniques utilisées à partir de cet objet sont très proches de celles étudiées par Olivier Mariez.

Le chapitre suivant décrit l'organisation et le fonctionnement de cet objet. Le chapitre 9, lui, décrit plus précisément comment cette structure peut servir à simuler des réseaux de failles plus complexes.



---

---

## Chapitre 8 Le Structural Model

### Un gestionnaire de contacts pour le modèle structural

---

---

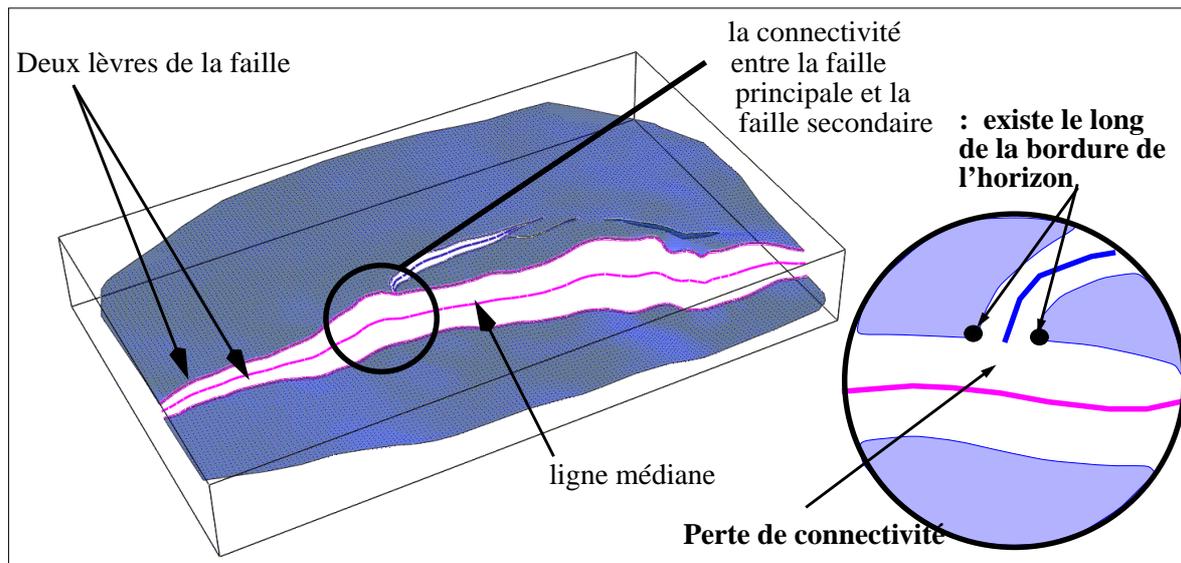
Cette partie du travail a été réalisé collaboration avec Isabelle Duvinage et Jean-Claude Dulac. En effet, elle correspond à la définition d'une nouvelle structure de données qui permet de résoudre des problèmes communs aux deux thèses: *la gestion des contacts dans un modèle structural*.

J'ai participé à la description et l'organisation des différentes classes, mais la majeure partie des algorithmes a été réalisée par Isabelle Duvinage. C'est pourquoi je ne les décrirai pas dans le détail et que pour plus de précisions, on pourra se reporter à son mémoire de thèse [103 - Duvinage 3].

Ce chapitre décrit plutôt la structure de données, les choix qui ont été faits, et l'utilisation de cette structure.

Les buts de ce travail sont multiples:

- **gérer les objets plus globalement**: en effet, plutôt que de construire les objets indépendamment, puis de construire un modèle volumique, il paraît plus judicieux d'utiliser toutes les informations disponibles dès le départ. Il est en outre moins fastidieux pour un utilisateur de réaliser en une fois la même opération sur un ensemble de surfaces lorsque cela est possible, plutôt que d'avoir à la répéter autant de fois qu'il existe de surfaces dans le modèle.
- **utiliser chaque représentation pour sa spécificité**, mais sans que l'utilisateur en soit gêné. Cette question a déjà été abordée en signalant que les objets faille ne devaient être responsables que de la partie simulation géométrique, et que la gestion des contraintes devait se faire par l'intermédiaire des surfaces triangulées. Cependant, il faut que ceci soit *transparent* à l'utilisateur: il doit travailler avec une Faille, sans forcément savoir par quel type d'objet de base elle est représentée pour tel ou tel algorithme (points, lignes, surface triangulée...)
- **gérer les contacts entre les différents composants du modèle structural**, de la manière la plus *automatique* possible, tout en permettant à l'utilisateur d'intervenir si cette solution automatique n'est pas satisfaisante. Le fait de stocker tous les objets dans une même structure permet d'avoir plus d'information. Ainsi, l'information contenue dans la succession des bords d'un horizon est très complète pour définir les relations entre les failles qui le recourent. Cette information est beaucoup plus difficile à retrouver si l'on construit les failles indépendamment, sans se servir des horizons [figure 8-1].
- **construire des modèles pré-découpés**, à la manière de ce qui a été proposé par Olivier Mariez pour construire un modèle volumique à partir de coupes sériées [108 - Mariez]. Ici, les corrélations pour détecter les lignes de contact horizon-faille (et éventuellement faille-faille), se feront d'un horizon à l'autre. Le fait de travailler dans une direction perpendiculaire à celle des coupes, en fait une méthode complémentaire de la méthode précédente, qui peut énormément améliorer l'automatisation de construction des modèles volumiques dans le logiciel.



**Figure 8-1 :** L'information concernant la connectivité entre les failles est perdue lorsqu'elles sont construites indépendamment, et sans tenir compte des horizons.

## 1. Gérer différentes représentations pour un même objet géologique

### 1.1 Les différents composants du modèle structural

Comme nous l'avons vu dans le chapitre précédent, nous travaillons avec un type de modèle volumique défini par ses éléments frontière ou interfaces.

En fonction de la zone étudiée, ou de l'entreprise qui réalise le modèle, les données disponibles pour représenter ces différentes interfaces peuvent être très variées. Cependant, du point de vue géologique, le modèle est toujours constitué des mêmes éléments: des failles, des horizons et les limites du domaine à modéliser.

#### 1.1.1 Les failles

Les données qui permettent de représenter les failles sont de natures très variées. En effet, ces dernières sont difficiles à détecter, et des méthodes très différentes sont utilisées pour arriver à les représenter au mieux [figure 8-2]:

- ensembles de points correspondant à l'interprétation de nombreuses sections sismiques dans les trois directions de l'espace. Ces données sont souvent incohérentes, si elles n'ont pas fait l'objet d'un gros travail de mise en cohérence en trois dimensions.
- polygones de failles: ces données ne sont pas disponibles systématiquement, et sont même inexistantes dans certaines compagnies. Pourtant, elles constituent un type de données souvent plus travaillé que les précédentes, et donc, a priori plus fiable.
- interprétations de coupes sériées (données extraites à partir des coupes en 2D)
- surfaces triangulées ou objets faille: ces données correspondent au cas idéal, où un travail de modélisation a déjà été fourni. Elles correspondent au type de données le plus abouti (ce qui n'empêche pas certaines incohérences)

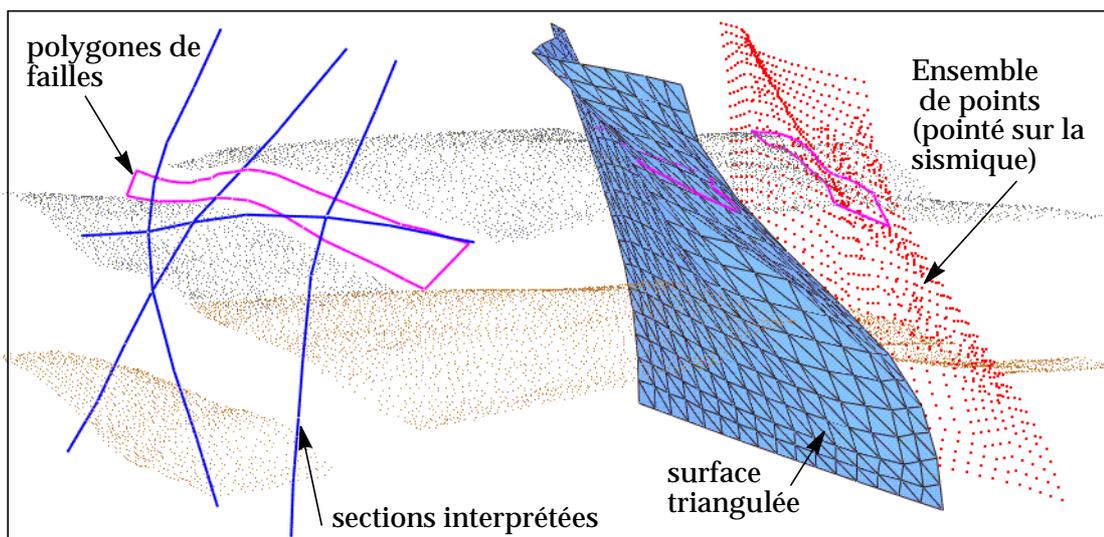


Figure 8-2 : Les représentations possibles pour modéliser les failles

### 1.1.2 Les horizons

Les données permettant de modéliser les horizons sont moins variées et souvent plus sûres, puisqu'elles sont directement observables sur les données sismiques. Parmi les différentes représentations disponibles, on trouve:

- des ensembles de points migrés profondeur. Ces ensembles de points sont souvent «pollués» dans les zones faillées, et nous supposons dans la suite qu'ils ont été nettoyés. Ainsi, tous les points contenus dans les données seront utilisés pour la modélisation. Afin de nettoyer les données, il est possible d'utiliser des outils de morphologie mathématique qui donnent souvent de très bons résultats [77 - Duvinage 1].
- des bords d'horizons (lignes polygonales). Il est assez rare de partir de telles données. Par contre, elles peuvent être obtenues facilement à partir des précédentes en utilisant des outils de morphologie mathématique [77 - Duvinage 1]
- des surfaces triangulées: comme pour les failles, ce type de données correspond au cas idéal où un travail de modélisation et de réflexion a déjà été réalisé en collaboration avec un structuraliste.

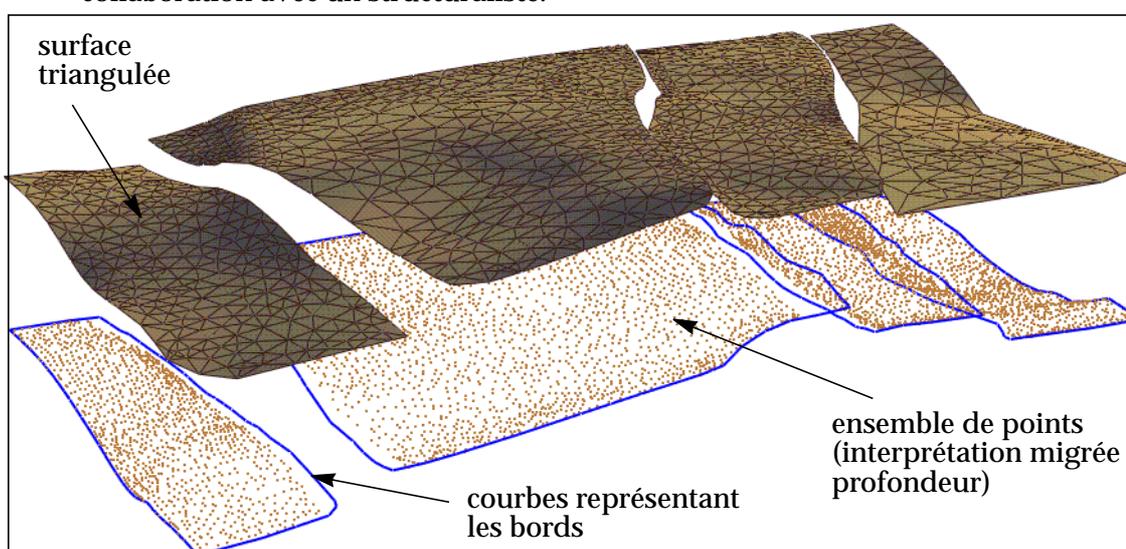


Figure 8-3 : Les représentations possibles pour modéliser les horizons

### 1.1.3 La boîte englobante

Enfin, comme cela a été souligné dans le chapitre précédent, le modèle volumique correspond à une *portion finie de l'espace 3D*. Il faut donc matérialiser les limites de la zone à modéliser. En général, cette boîte englobante est représentée par une unique surface triangulée fermée, ou par un ensemble de surfaces triangulées constituant un volume fermé.

Dans notre cas, c'est la deuxième solution qui a été choisie: elle permet de pouvoir traiter chacune des faces de la boîte de la même manière que les failles dans la gestion des contacts qui sera faite par la suite.

Actuellement, la boîte est calculée à partir du volume occupé par l'ensemble des interfaces du modèle. On peut aussi imaginer que l'utilisateur sélectionne une zone en dessinant une courbe, et que la boîte soit calculée par extrusion verticale de cette courbe [figure 8-4].

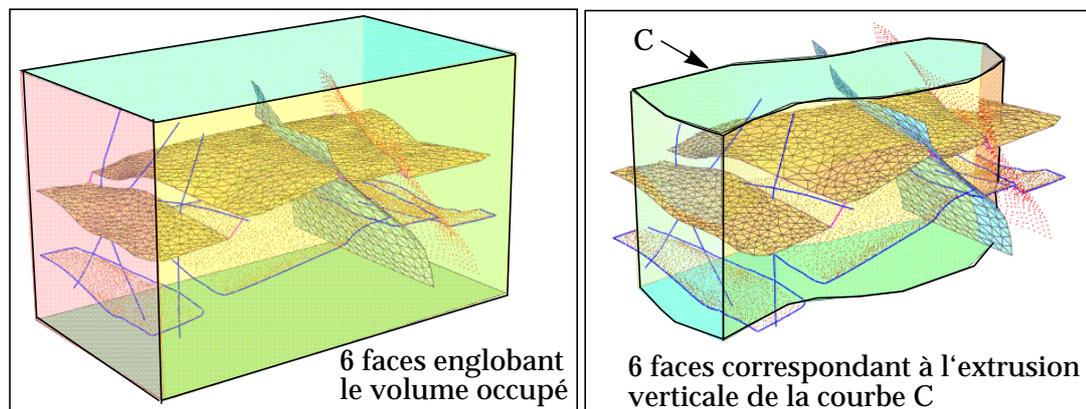


Figure 8-4 : Représentation des limites du domaine à modéliser.

## 1.2 Séparation objet géologique / représentations de cet objet

Comme cela a été montré, les représentations sont très variées pour un même objet géologique. Il faut donc séparer l'objet géologique de sa liste de représentations.

En effet, lorsque l'utilisateur travaille avec la faille 'F1', il n'a pas à savoir que la représentation ligne est utilisée pour un algorithme et la représentation surface pour un autre. Il doit pouvoir demander des actions sur la seule chose qui est géologique: la faille F1.

Si l'on est capable de séparer l'objet géologique de ses différentes représentations, et de choisir en interne la représentation la plus appropriée pour chaque algorithme, la communication avec les géologues utilisateurs devient beaucoup plus facile.

C'est pourquoi nous avons séparé la définition de l'objet géologique (avec son nom, son âge, le stockage de ses relations avec ses interfaces) de la définition des différentes représentations de cet objet.

### 1.2.1 la classe InterfaceModel et ses classes dérivées

Une première classe de base (INTERFACEMODEL) permet de définir les fonctionnalités communes à toutes les interfaces du modèle. Elle contient:

- \* le *nom* de l'objet,
- \* son *âge* géologique,
- \* sa *couleur* (afin que toutes les représentations soient codées par la même couleur)
- \* la *liste des représentations* existantes pour cet objet,
- \* la *liste des bords* de cet objet (ce sont eux qui interviennent dans les contacts)
- \* la *liste des contacts* entre cet objet géologique et les autres interfaces du modèle.

Cette classe est ensuite dérivée en deux classes qui permettent de définir les fonctions spécifiques aux différents objets géologiques: HORIZONMODEL pour les horizons et FAULTMODEL pour les failles et les faces de la boîte englobante.

Cette organisation permet de réduire énormément le code, puisque toutes les fonctions et variables communes sont définies dans la classe mère. Ainsi, il est plus facile de maintenir et de corriger le code [figure 8-5].

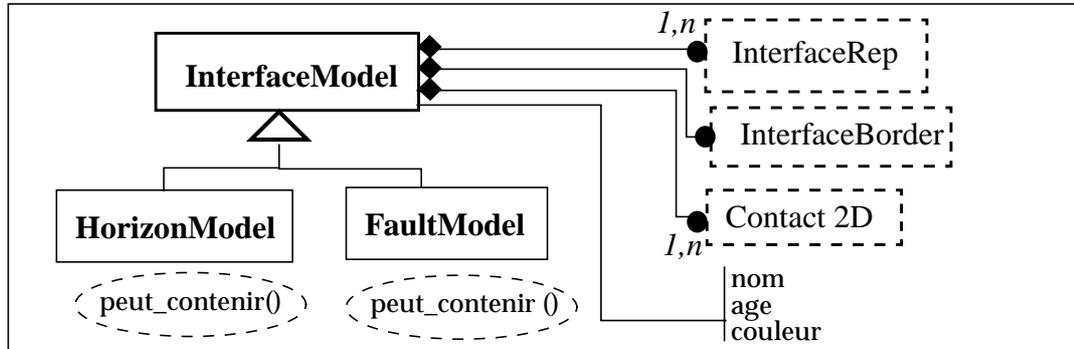


Figure 8-5 : Relations entre les classes InterfaceModel, HorizonModel, FaultModel et InterfaceModelRep (les conventions utilisées pour ce schéma et les suivants sont décrites dans la figure 6-9, page 122).

### 1.2.2 la classe InterfaceRep et ses classes dérivées

De la même manière, une classe générique (INTERFACEREP) permet de définir toutes les variables et fonctions communes aux différentes représentations:

- \* l'InterfaceModel qui la contient,
- \* le type de représentation qui a servi à construire toutes les autres (le «parent»)

Cette classe permet en outre de retrouver la représentation des bords de l'objet qui correspondent à son type de représentation de l'objet donné (des extrémités de génératrices pour la représentation ObjetFaille, des lignes pour la représentation surface triangulée...). Ceci sera développé un peu plus loin.

Comme les représentations peuvent être identiques entre les failles et les horizons (points, surfaces, lignes par exemple), celles-ci sont définies au niveau de la classe parente. Par contre, les failles ont potentiellement plus de représentations différentes (les objets faille ne peuvent représenter que des failles par exemple). Chaque classe dérivée possède donc une fonction «garde-fou» qui teste si la représentation que l'on veut créer est compatible avec le type d'interface sur lequel on travaille

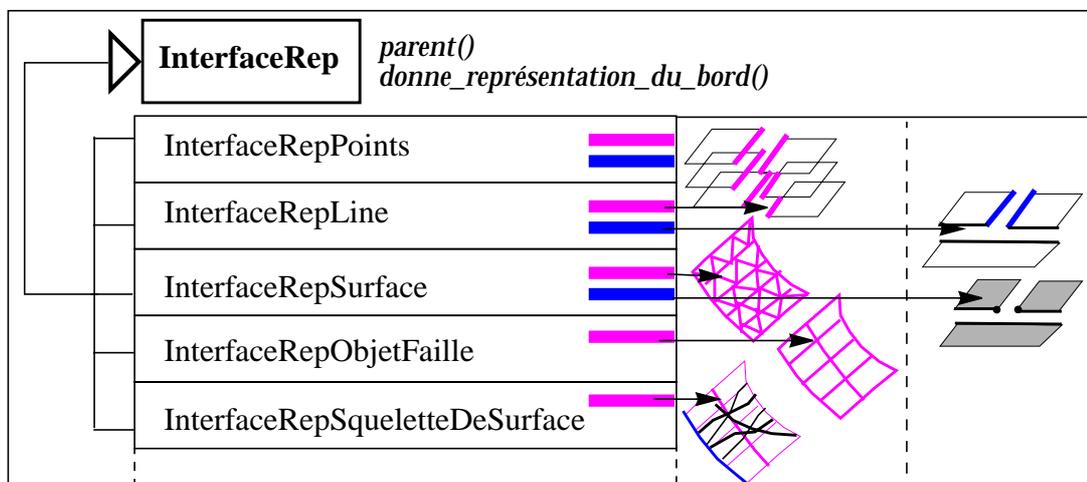


Figure 8-6 : Les différentes représentations possibles pour les interfaces du modèle structural: points, lignes, surfaces triangulées...

### 1.3 Interface avec l'utilisateur

Cette organisation des classes qui sépare l'objet géologique des différents types d'objets qui peuvent le représenter permet d'améliorer énormément la communication avec l'utilisateur. Ainsi la communication se fait en utilisant les noms géologiques des objets (mieux connus des structuralistes), ce qui est beaucoup moins repoussant pour lui que de travailler avec des «TSurf», des «PLines», etc...

De plus, le fait d'avoir un objet global (le StructuralModel) qui stocke de manière ordonnée les différents objets géologiques, ainsi que toutes leurs représentations permet de regrouper de nombreuses commandes et de rendre moins fastidieuse la phase de modélisation surfacique.

#### 1.3.1 L'interaction se fait avec les objets géologiques, et non avec leur représentation

La séparation entre l'objet géologique et ses représentations est complètement *transparente* à l'utilisateur. Il ne travaille qu'avec les objets géologiques, quasiment sans avoir à savoir quel type de représentation est utilisé pour réaliser la tâche qu'il a demandée [figure ].

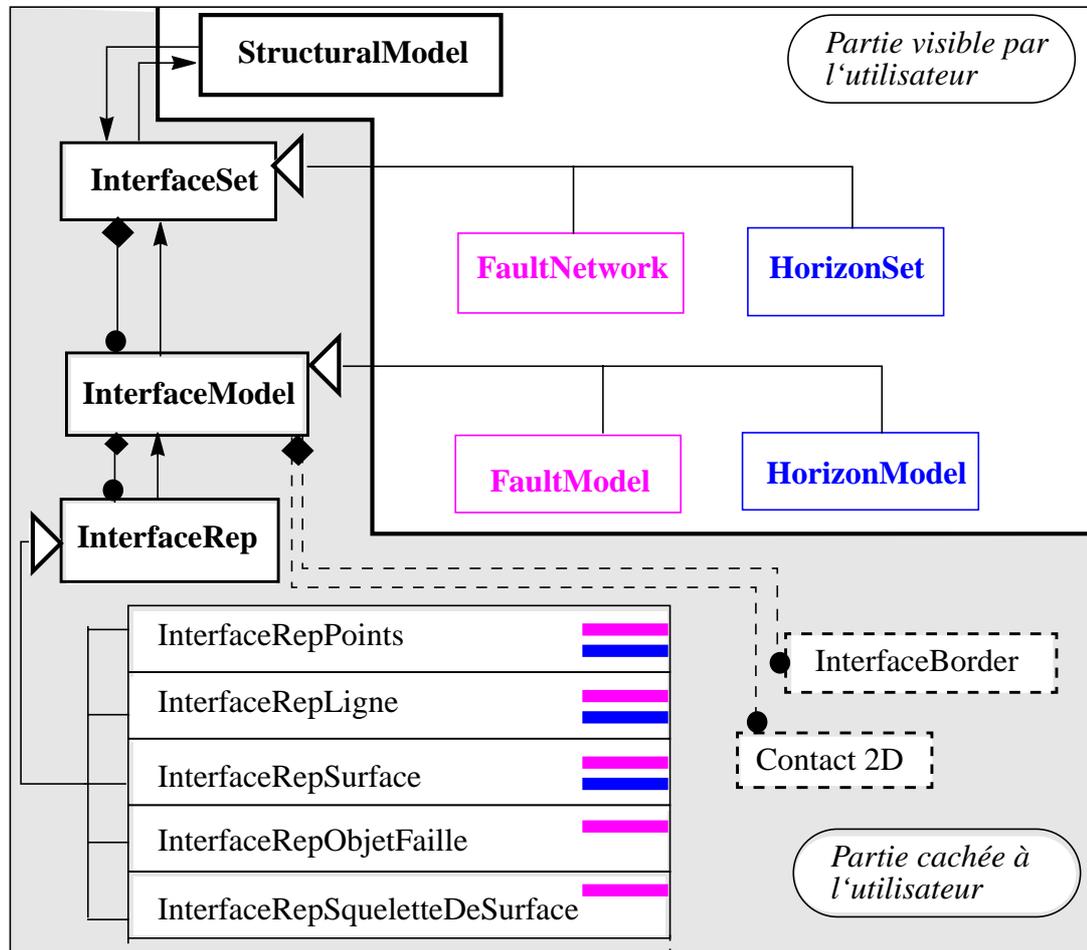


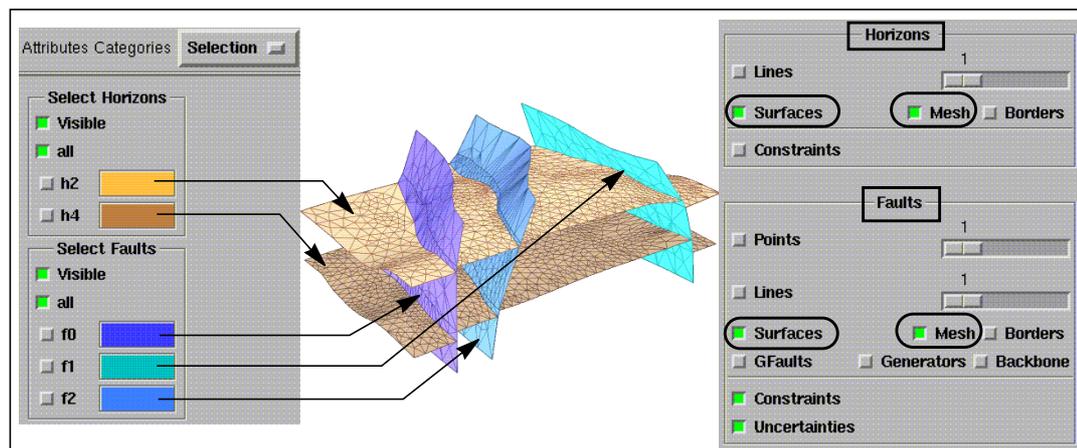
Figure 8-7 : Une petite partie seulement est visible par l'utilisateur.

En interne, les objets sont ordonnés en deux groupes, en fonction de leur *type géologique*. Un premier ensemble (FAULTNETWORK ou réseau de failles) ne contient que des objets de type «faille», tandis que le deuxième ensemble (HORIZONSET ou ensemble d'horizons) ne contient que des objets de type «horizon». Comme ces deux objets dérivent de la même classe, il est toujours possible pour certains algorithmes de travailler de la même manière sur les deux types d'objets, lorsque le traitement est indifférent pour les failles et les horizons, ce qui évite les duplications de code.

De la même manière, ce stockage des objets en fonction de leur type géologique, permet de faciliter la sélection des objets à visualiser.

Un premier panneau permet de sélectionner quels sont les objets à dessiner à l'écran (tous les horizons, ou une sélection de l'utilisateur et/ou toutes les failles ou une sélection de l'utilisateur).

Un second panneau permet de sélectionner de manière globale la représentation choisie pour un type d'objet géologique. Ainsi, les horizons pourront être visualisés sous la représentation «surface triangulée», tandis que les failles peuvent être vues sous la forme «objet faille» [figure 8-8].



**Figure 8-8 :** Le contrôle de la visualisation des objets du modèle se fait de manière plus globale et plus ordonnée. Ici, il a été décidé de visualiser tous les horizons sous forme surface triangulée, avec le maillage visible, et toutes les failles sous forme de surface triangulée, avec le maillage visible.

### 1.3.2 Gestion plus globale des objets géologiques

Ce regroupement des objets en fonction de leur type géologique permet aussi de regrouper certaines commandes.

Ainsi, plutôt que d'appliquer la commande qui permet de stocker un âge géologique sur les différents objets, un par un (ce qui fait 5 lancement de commandes sur ce petit exemple, mais qui peut devenir beaucoup plus fastidieux sur un gros jeu de données), il suffit d'utiliser une seule commande qui permet de travailler globalement sur toutes les interfaces du modèle [figure 8-9]. Ceci permet en outre de vérifier la cohérence des différents âges qui sont assignés aux interfaces du modèle.

Et ceci peut s'appliquer à toutes les commandes où la définition des paramètres est valable soit pour tous les objets d'un type donné, soit pour un groupe de ces objets. Il suffit juste de construire une commande globale qui appelle en interne autant de fois la commande de base que d'objets à traiter. Ceci ne demande pas beaucoup de travail, mais peut simplifier énormément la vie de l'utilisateur.

On peut ainsi:

- étendre toutes les failles en même temps, si la représentation Objet Faille a été créée
- densifier le maillage de toutes les surfaces triangulées (horizons et/ou failles)
- installer les contraintes sur toutes les failles (ou sur tous les horizons)
- étendre toutes les failles et/ou tous les horizons en une fois, à l'aide de l'interpolateur, afin de pouvoir construire un modèle fermé
- ...

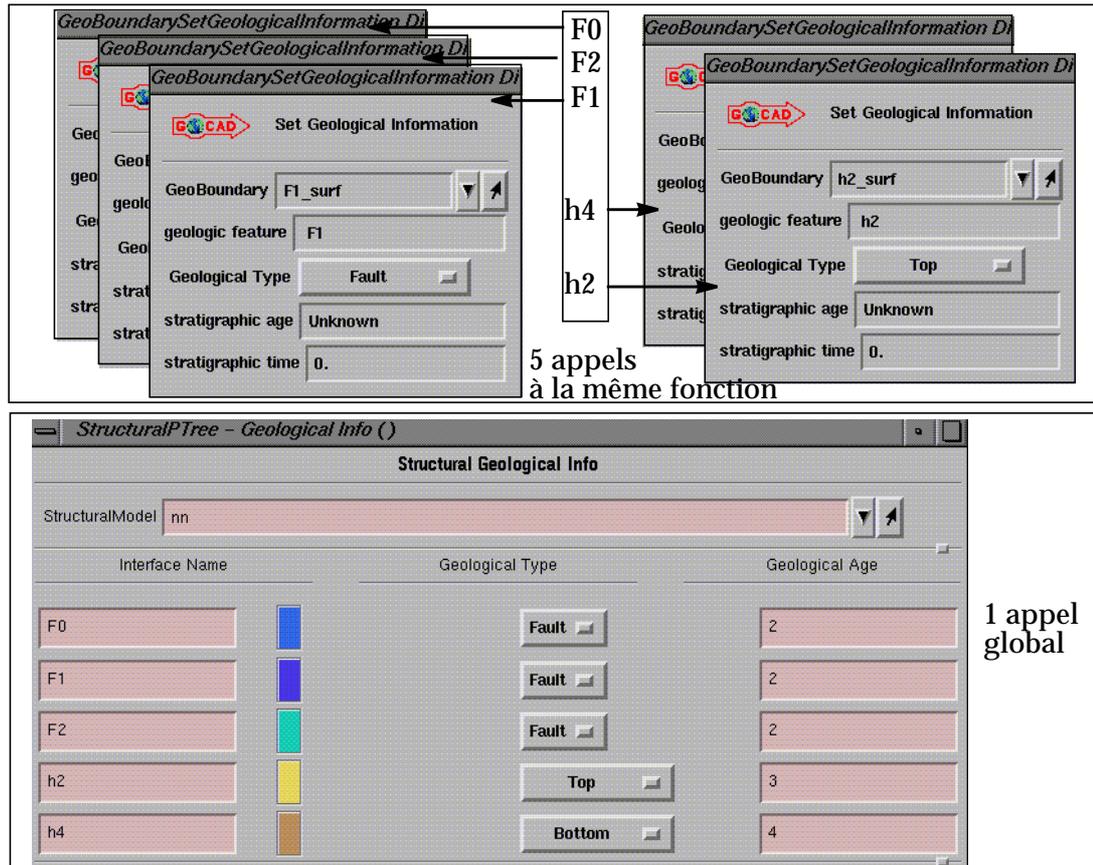


Figure 8-9 : Regroupement de cinq appels à une même commande en une commande globale (aucun algorithme nouveau, c'est juste une question d'interface).

## 2. Les relations entre les différents composants du modèle

Comme cela a été montré dans le chapitre précédent, un modèle structural n'est pas composé uniquement d'une juxtaposition d'interfaces (failles et horizons). Il représente aussi les *relations* entre ces différents objets géologiques. Ceux-ci peuvent être de différents types.

### 2.1 Différents types de contacts

#### 2.1.1 Bord sur surface: contacts 2D

Le premier groupe de contacts met en relation un (ou plusieurs) bord(s) d'interface (faille ou horizon) avec une (ou plusieurs) autre(s) interface(s) (faille, horizon, ou boîte englobante).

Nous avons séparé ce groupe en trois sous-familles, en fonction du nombre de bords et du nombre d'interfaces qui interviennent dans le contact [figure 8-10]:

- **le contact en T**: c'est le plus courant (faille-faille, faille-bord, horizon-faille, horizon-bord, horizon-horizon). Comme nous le verrons plus loin, tous ces contacts ne sont pas encore détectés automatiquement à l'heure actuelle.
- **le contact multiple** (en K, en X, ou plus complexe): il correspond le plus souvent à un ensemble de failles qui se rejoignent le long d'un de leurs bords.
- **le contact multiple** se terminant en T: il combine les deux précédents.

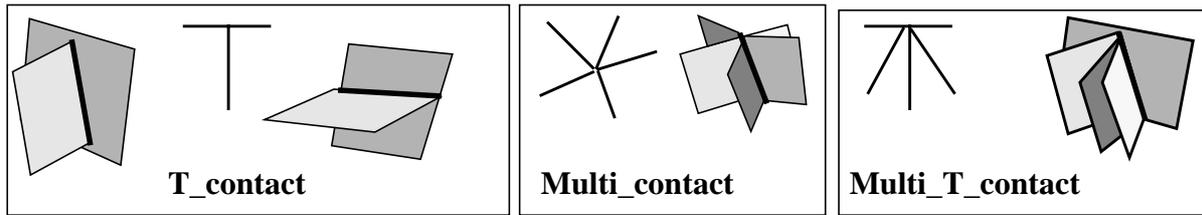


Figure 8-10 : Les différents types de contact «2D».

#### a) Le contact en T «simple»

Ce type de contact est très fréquent, en tous cas dans les jeux de données que nous avons eu à traiter. Il ne met en relation que deux interfaces à la fois et correspond le plus souvent à une connection entre un bord d'horizon et une faille (ou la boîte englobante). Il peut aussi correspondre à une connection entre deux failles (ou entre un bord de faille et la boîte englobante).

La ligne de contact peut être soit sub-horizontale (connection entre deux failles ou entre deux horizons) soit sub-verticale (connections entre deux failles ou entre une faille et la boîte englobante). Comme cela sera montré plus loin, nous ne pouvons détecter que les contacts sub-verticaux de manière automatique, à partir de l'information extraite des bords d'horizons. Mais la structure de donnée est exactement la même, quelles que soient les deux interfaces en contact, et quelle que soit l'orientation de la ligne d'intersection. Ainsi, cela laisse la porte ouverte à un mixage de la méthode présentée dans ce chapitre, et de la méthode étudiée par Olivier Mariez à partir des coupes géologiques [108 - Mariez] (qui ne détectent que les contacts sub-horizontaux).

En effet, ce type de contact contient deux pointeurs vers des objets de type INTERFACEMODEL (un pour l'objet qui se branche [*branching*] et un pour l'objet principal [*main*]), ainsi qu'un pointeur sur le bord de l'objet secondaire en contact avec l'objet principal [figure 8-11]. Cette description est donc très générale, et toutes les combinaisons mettant en jeu deux interfaces seulement seront prises en compte de la même manière par cette structure.

Cette classe est responsable de l'installation d'une contrainte de type «*bord sur surface*» entre les représentations surfaces triangulées des deux interfaces en contact. Le contrôleur sera la surface triangulée de l'interface "*main*", et le contrôlé le bord de l'interface "*branching*".

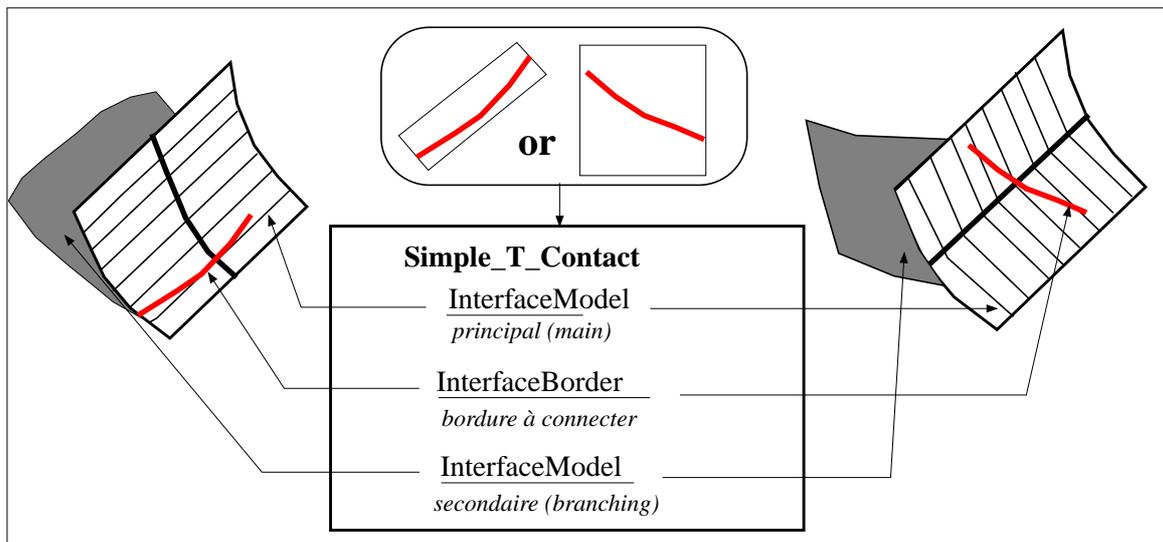


Figure 8-11 : Structure d'un contact en T simple

### b) Le contact multiple (Multi-contact)

Ce contact est beaucoup plus rare sur les jeux de données que nous avons traités, et ne concerne en général qu'un ensemble réduit de failles. Mais, de la même manière que pour le contact en T simple, il est défini de façon très générale, afin de pouvoir prendre en compte des contacts que nous n'avons pas encore rencontrés.

La gestion de ce type de contact est plus complexe que la précédente, puisqu'il faut calculer le bord commun entre toutes les interfaces en contact. Devant le grand nombre de cas particuliers à traiter, différentes stratégies sont proposées pour calculer cette ligne commune:

- ligne correspondant au *barycentre* entre tous les bords en contact
- ligne correspondant à un *bord particulier* (le plus grand par exemple)
- *ligne dessinée* par un structuraliste.

Cette classe se contente de stocker une liste de pointeurs vers les bords d'interface qui sont en contact, ainsi que la ligne commune. Elle est responsable de l'installation d'une contrainte qui permet d'attirer tous les bords en contact par la ligne commune.

**Remarque:** les positions des nœuds de la ligne commune seront recalculées après chaque simulation, sans que cette ligne soit détruite puis qu'une nouvelle ligne soit reconstruite. Ainsi, la contrainte installée au départ peut être utilisées durant tout le cycle de simulations.

### c) Le contact multiple se terminant en T (Multi-T-Contact)

Ce type de cas peut se produire lorsque deux failles du modèle précédent ont été regroupées en une seule, sur laquelle toutes les autres doivent se brancher. Cette classe étant une combinaison des deux précédentes, elle est définie à l'aide d'un *héritage multiple*.

Par l'héritage de la classe T\_CONTACT, elle permet de stocker l'interface principale sur laquelle les autres failles se rattachent. Par l'héritage de la class MULTI\_CONTACT, elle permet de stocker la liste des bords en contact, et de calculer la ligne commune.

De plus, elle projette cette ligne commune sur la faille principale, afin de concilier les deux types de contact. Ceci est réalisé en installant une contrainte spéciale qui force la distance entre la ligne commune et l'interface principale à être nulle. Le fait d'installer une telle contrainte assure qu'elle sera respectée automatiquement lorsque l'interpolation des différentes interfaces sera appliquée (sans avoir à se rappeler qu'il faut projeter la ligne après chaque simulation).

### d) Les classes InterfaceBorder, InterfaceBorderRep et leurs dérivées

A plusieurs reprises, nous avons parlé d'un nouveau type de structure qui concerne les bords d'une interface. De la même manière qu'une interface peut être représentée par différents types d'objets (surface, ligne,...), un bord d'interface peut être représenté de différentes manières (lignes, points,...). A chaque type de représentation de l'interface, correspond un type de représentation pour le bord (sauf pour la représentation "points"):

- \* ligne polygonale --> extrémité de ligne
- \* surface triangulée --> ligne polygonale
- \* objet faille --> génératrice ou groupe de points portant le même indice

**Remarque:** Actuellement, seule la représentation *surface* du bord est réellement utilisée, pour y attacher des *contraintes* à respecter durant l'interpolation.

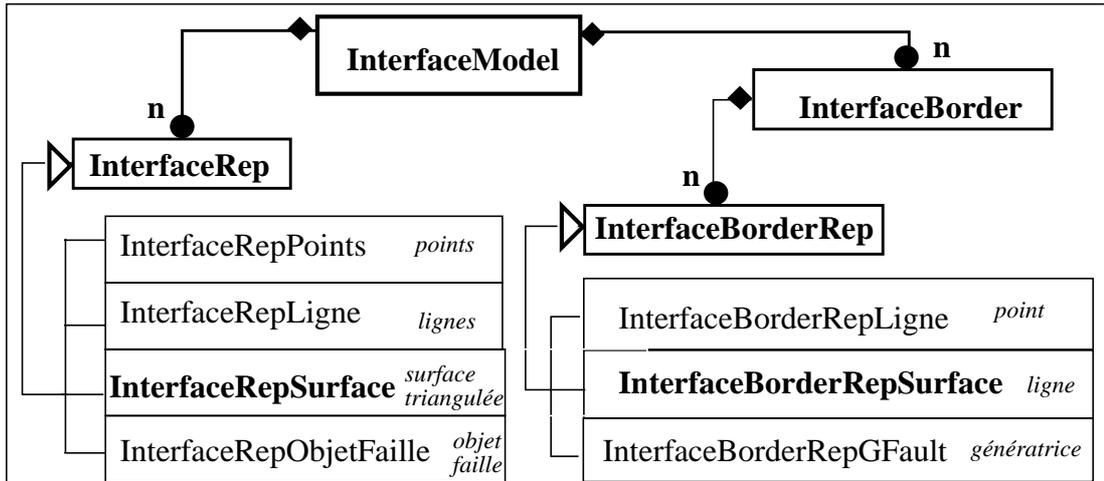


Figure 8-12 : Correspondance entre les InterfaceRep et les InterfaceBorderRep

e) Diagramme récapitulatif pour les contacts à deux dimensions

Maintenant que les différentes classes ont été décrites, il est possible de les rassembler en un diagramme commun [figure 8-13]. Toutes ces classes dérivent de la classe 2d\_Contact, et sont chargées d'installer la contrainte correspondant à leur type.

**Remarque:** dans le cas d'un contact entre un horizon et une faille, nous avons dérivé la classe de base T\_CONTACT en HORIZONFAULTCONTACT.

Cette classe dérivée gère en fait l'équivalent de deux SIMPLE\_T\_CONTACT (un pour la lèvre supérieure de la faille, et un pour la lèvre inférieure). Plutôt que de stocker un INTERFACEBORDER, elle stocke une structure de type BORDERPAIR qui elle-même stocke deux INTERFACEBORDER. Par contre les pointeurs vers les INTERFACEMODEL *main* et *branching* ne sont stockés qu'une seule fois pour les deux bords, afin de limiter un peu l'encombrement mémoire.

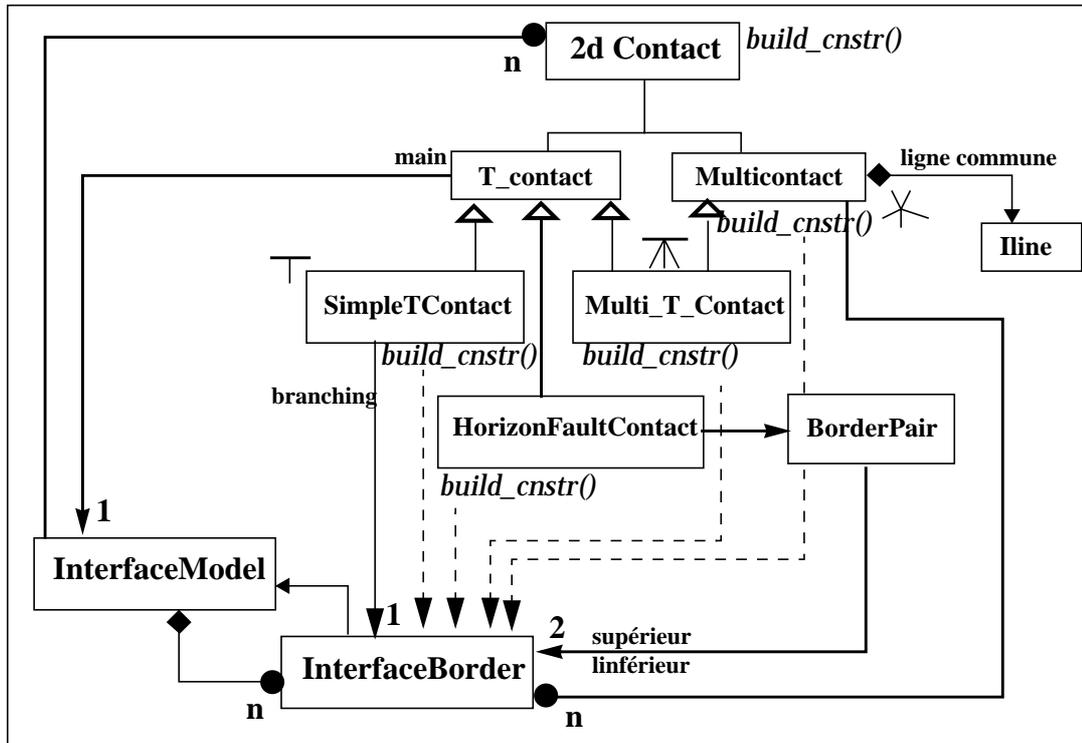


Figure 8-13 : Organisation des contacts «bord sur surface» (Contact 2D)

### 2.1.2 Points triples: contacts 1D

Le deuxième type de contact met en relation un nœud du bord d'une interface avec deux (ou plus) autres bords d'interface. Ce cas se produit lorsqu'un coin d'horizon est en contact avec un contact en T entre 2 failles, ou entre une faille et un bord. Ce coin d'horizon se trouve donc à l'intersection entre trois contacts en T:

- \* le contact *main-branching*
- \* le contact *horizon-main*
- \* le contact *horizon-branching*

Il faut s'assurer que ce coin reste sur la ligne de contact entre l'interface «*main*» et l'interface «*branching*». Ceci est réalisé en installant une contrainte spéciale qui force une extrémité de bord à coulisser sur un autre bord pendant une interpolation géométrique. Comme ce point en coin est partagé par deux bords de l'horizon, il suffit d'installer cette contrainte une seule fois (ceci limite un peu le nombre de variables à stocker).

Afin d'installer cette contrainte, la class 1D\_Contact stocke l'INTERFACEBORDER sur lequel le coin doit coulisser, l'un des INTERFACEBORDER auquel le coin appartient, et le coin lui-même (INTERFACEBORDEREXTREMITY).

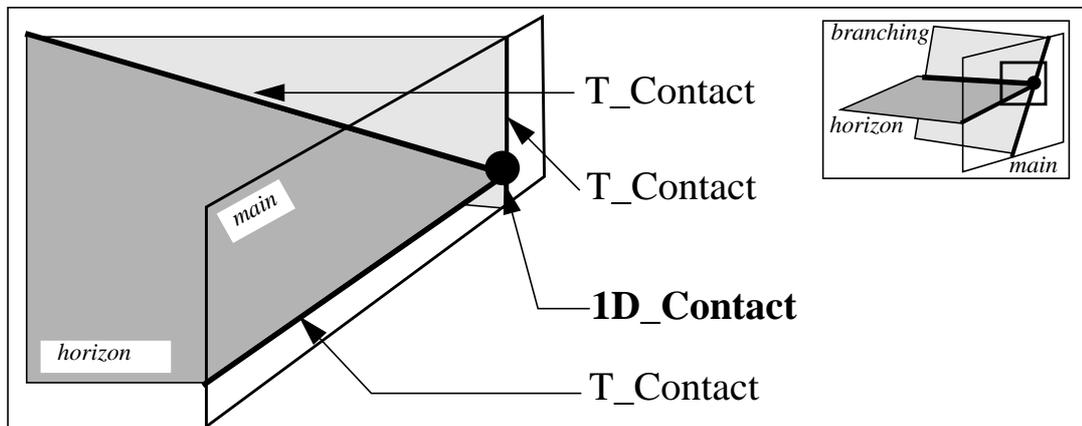


Figure 8-14 : Cas de figure correspondant au point triple de type «contact 1D».

Même si seule la représentation surface est utilisée pour installer les contraintes, la classe INTERFACEBORDEREXTREMITY a été construite sur le même modèle que la classe INTERFACEBORDER. La représentation ligne est, elle, utilisée pour détecter ce type de contact particulier, comme nous le verrons un peu plus loin.

Cette classe INTERFACEBORDEREXTREMITY contient donc une liste de représentations de types INTERFACEBORDEREXTREMITYREP. Cette classe étant elle-même dérivée en INTERFACEBORDEREXTREMITYREPLIGNE (détection), INTERFACEBORDEREXTREMITYREPSURFACE (stockage de la contrainte).

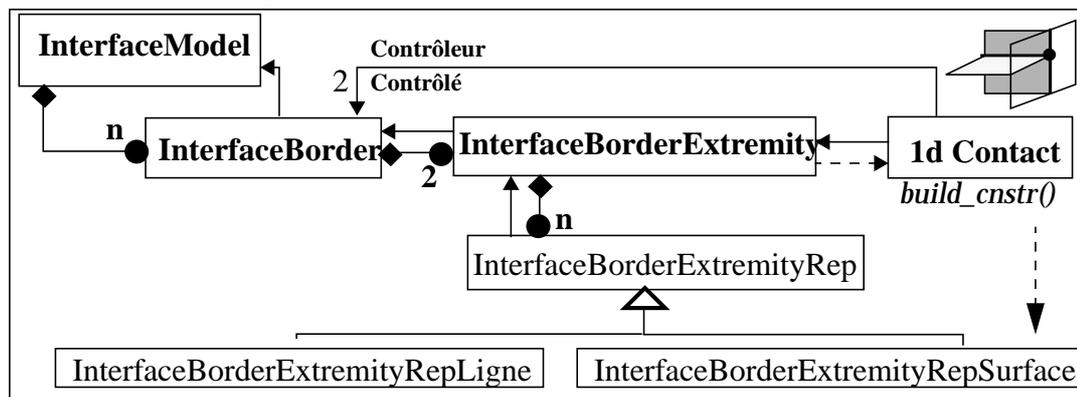


Figure 8-15 : Organisation des contacts en points triples (Contact 1D)

## 2.2 Détection automatique des contacts

Cette sous-section correspond à un travail entièrement réalisé par Isabelle Duvina-ge. Le principe des algorithmes est décrit brièvement ici, car ils constituent une partie essentielle de la préparation des simulations (la détection et le stockage des contacts). Mais pour plus de détails, on pourra se reporter à [103 - Duvina-ge 3].

L'intérêt d'avoir un objet qui stocke tous les composants du modèle, plutôt que de modéliser les objets indépendamment les uns des autres, est de pouvoir utiliser toutes l'information disponible non seulement pour détecter, mais aussi pour gérer ces contacts. Dans la partie actuellement disponible, toute la détection d'information est réalisée à partir des bords des horizons du modèle. Mais les structures sont assez générales pour envisager, plus tard, une détection d'autres contacts à partir d'autres types de données, comme les coupes géologiques par exemple.

### 2.2.1 Découpage des bords d'horizons

Avant de pouvoir appairer les bords d'horizons qui correspondent à une même faille, il faut commencer par découper le bord de l'horizon en sous-parties.

Cet algorithme se déroule en deux grandes étapes:

#### 1. séparation des bords externes et des bords correspondant aux failles

Cette première étape commence par l'extraction de tous les bords de l'horizon. Ensuite, une courbe englobant tous ces bords est calculée. Enfin, une tessellation (soit par triangulation, soit par tétraédrisation) permet de séparer les *bords externes* et les *bords internes* (correspondant à des failles): les premiers sont connectés à la courbe englobante, tandis que les seconds ne sont connectés qu'entre eux. Cette méthode permet de détecter tous les types de failles, qu'elles traversent entièrement l'horizon ou non, qu'elles se rejoignent entre elles ou qu'elles soient isolées [figure 8-16].

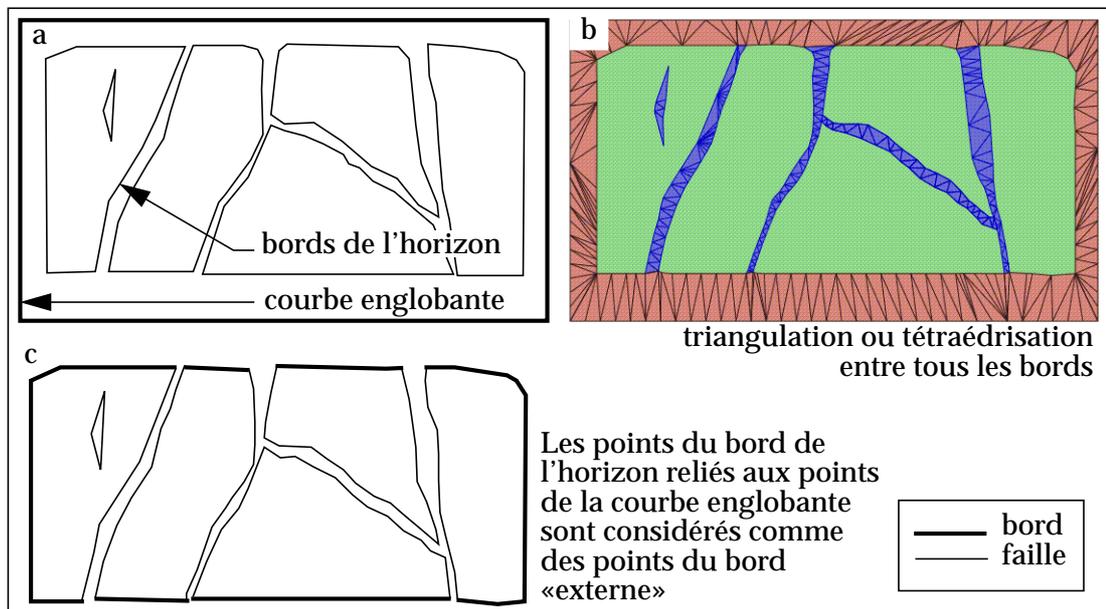


Figure 8-16 : Séparation des bords externes et des bords internes correspondant à des failles

**Remarque:** la méthode de triangulation est beaucoup plus rapide que la méthode de tétraédrisation, mais nécessite que tous les bords puissent être projetés sur un plan (ce qui risque de poser des problèmes pour certaines failles inverses). Durant ce travail de thèse, seule la méthode par tétraédrisation était disponible (un peu plus lente, mais plus générale). Des essais sont actuellement en cours pour tester la méthode par triangulation.

## 2. Découpage des bords internes et appariement des bords correspondant à une même faille

Ces deux étapes délicates (puisqu'elles conditionnent la suite du processus) se font en même temps. Un point P de l'un des bords B1 (lèvre supérieure pour fixer les idées) est sélectionné. En utilisant la tessellation, ainsi qu'un critère de distance, on se déplace alternativement sur le bord B2 (lèvre inférieure correspondant à B1) et sur le bord B1. Ce déplacement se poursuit tant que le point courant p1 et son voisin direct p3 sont situés sur le même bord (B1) et sont reliés tous les deux au même point p2, voisin du point précédent p0. Dès que deux points (p5 et p7 par exemple) sont reliés à un même point (p6), mais ne sont plus voisins, le processus s'arrête.

Les points p5, p6 et p7 sont marqués comme étant des extrémités de bords. Les bords B1 et B2 sont mis en correspondance. De plus, le fait que les bords B4 et B1 soient connectés au niveau du point P5 est stocké. Cette information signale que les deux failles correspondantes (F1 et F2) sont en contact [figure 8-17].

Ceci est recommencé jusqu'à ce que tous les points aient été visités.

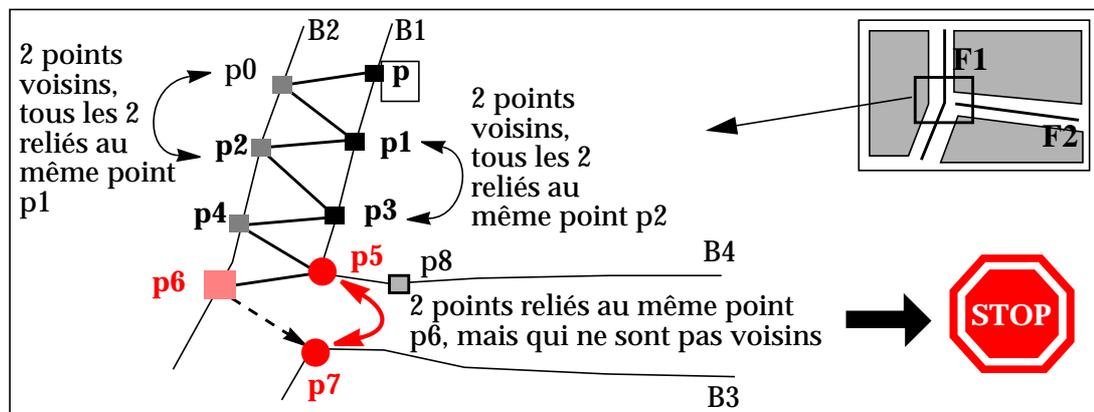


Figure 8-17 : découpage des bords correspondant aux failles, et appariements des bords se correspondant.

En même temps que la représentation ligne est découpée, au fur et à mesure de l'avancée de cet algorithme, la représentation surface de l'horizon est modifiée en conséquence. A chaque fois qu'un triplet type p5-p6-p7 est détecté, des marqueurs d'extrémités de bords sont ajoutés sur le bord de l'horizon.

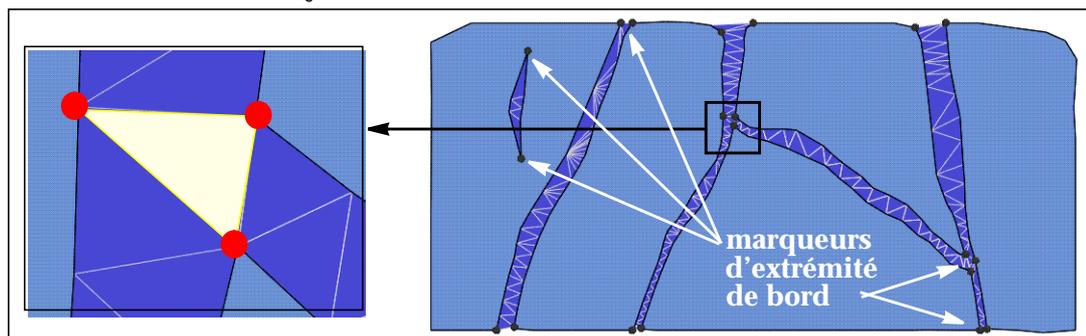


Figure 8-18 : Ajout des marqueurs correspondants sur la représentation surface de l'horizon.

Le même type d'algorithme permet d'apparier les bords d'horizons entre deux horizons successifs. Ensuite, une dernière structure permet d'associer un bord d'horizon (représentation surface du bord de l'interface "horizon 1") à un morceau de ligne (représentation ligne du bord de la même interface "horizon 1", mais aussi partie de la représentation ligne de l'interface "faille 1"). Ainsi, une même courbe polygonale peut appartenir à la fois à un modèle d'horizon et à un modèle de faille. Ce sont les mêmes morceaux de lignes qui peuvent ainsi être regroupés par faille ou par horizon.

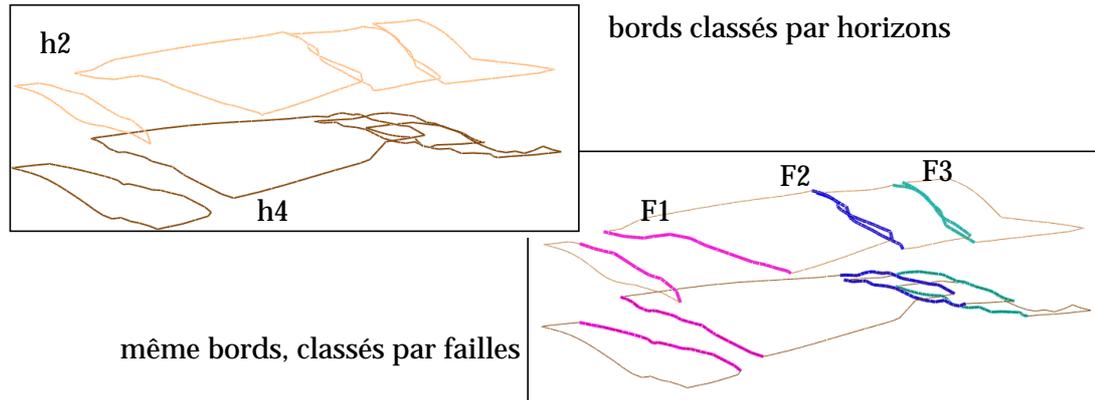


Figure 8-19 : Classification des bords soit par appartenance aux horizons, soit par appartenance aux failles.

### 2.2.2 Installation des liens correspondants

Jusqu'à présent, les contacts ont été détectés, mais seulement la topologie du contact est assurée (on sait que tel bord d'horizon est en contact avec tel bord de faille). Par contre, du fait de l'imprécision des données de départ, le contact n'est pas défini précisément de manière géométrique.

Afin de connecter les surfaces du modèle géométriquement, des contraintes doivent être installées pour être ensuite utilisées lors de la phase d'interpolation. Durant la phase de découpage, des liens sont installés entre les différents morceaux de bords. Ces liens sont ensuite utilisés dans des algorithmes complexes (qui ne seront pas détaillés ici) afin de définir quel est le type du contact (en T, multiple ou multi T) et de créer les structures correspondantes [103 - Duvinage 3]. Ces liens sont de quatre types et ne sont pas tous installés en même temps.

#### a) Pendant la phase de découpage

Durant cette phase, deux types de liens sont installés. Le premier concerne les triplets de type  $p5-p6-p7$  et relie le point  $p3$  au point  $p8$  [figure 8-17]. Ils permettent de savoir que la faille F1 passant entre les bords B1 et B2 et la faille F2 passant entre les bords B3 et B4 sont reliées. Ils contiennent même une information supplémentaire: c'est la faille F1 la faille principale sur laquelle se branche la faille F2. Ces liens servent aussi à détecter le branchement entre une faille et la boîte englobante du modèle.

Le deuxième type de liens sert à créer la paire de bords B1-B2 qui correspondent aux deux lèvres de la faille F1 sur l'horizon h2. Ils permettent de construire la ligne médiane entre B1 et B2.

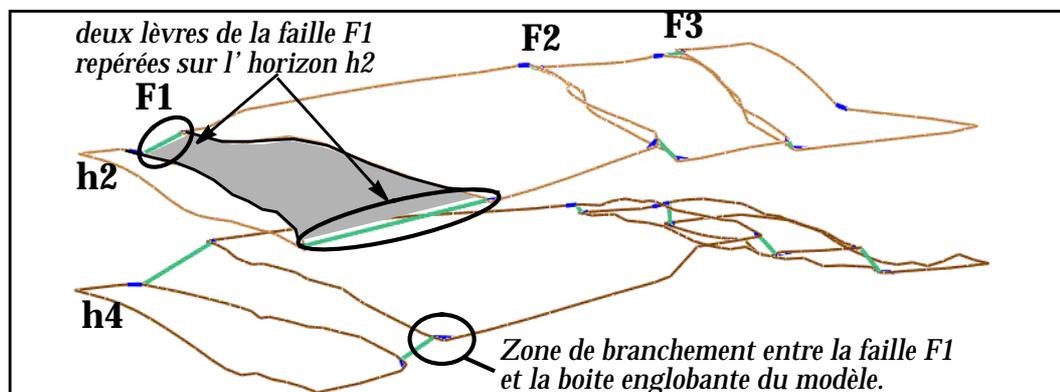


Figure 8-20 : Installation automatique des liens «branchement entre failles» et «association des deux lèvres d'une faille».

### b) Après le découpage

En utilisant un algorithme similaire (triangulation entre les bords de deux horizons consécutifs), il est possible d'installer un autre type de liens. Ce sont ces derniers qui permettent de regrouper tous les bords d'horizons correspondant à une même faille dans le même INTERFACEBORDERREPLIGNE du modèle de faille correspondant.

**Remarque:** cette partie de l'algorithme étant encore *instable*, lorsque les surfaces de failles sont disponibles en plus des données sur les horizons, elles sont utilisées pour guider l'installation de ces liens. Ceci sera précisé un peu plus loin (voir Création de toutes les surfaces triangulées "A partir des deux types de données" page 199).

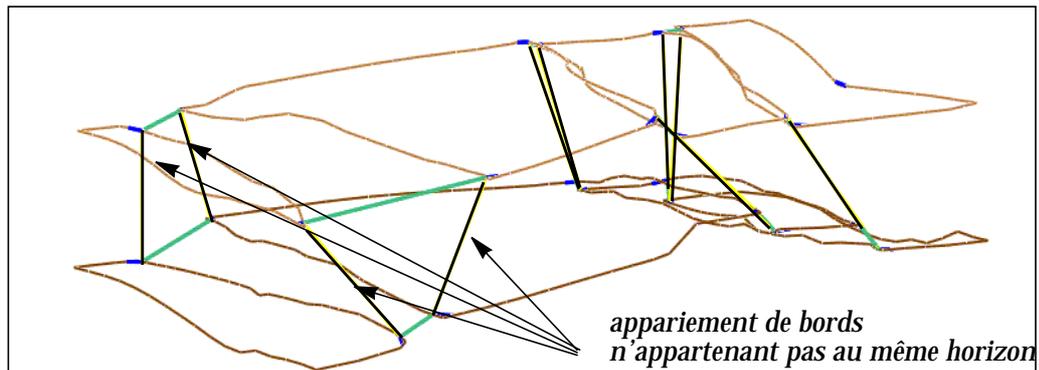


Figure 8-21 : Installation des liens de type «associations de lèvres de failles appartenant à deux horizons consécutifs».

### c) Intervention de l'utilisateur

Une fois la solution automatique proposée, l'utilisateur peut détruire et/ou ajouter certains liens.

Il est aussi possible d'ajouter des liens de la quatrième famille: les liens permettant de rassembler plusieurs bords latéralement. En effet, lorsqu'une faille secondaire se branche sur une faille principale, le bord de l'horizon est séparé en deux parties au niveau de la jonction.

En ajoutant des *liens de fusionnement*, il est possible de rétablir la continuité entre les bords reliés à la faille principale, de part et d'autre de la faille secondaire (tout en conservant l'information des liens de branchement).

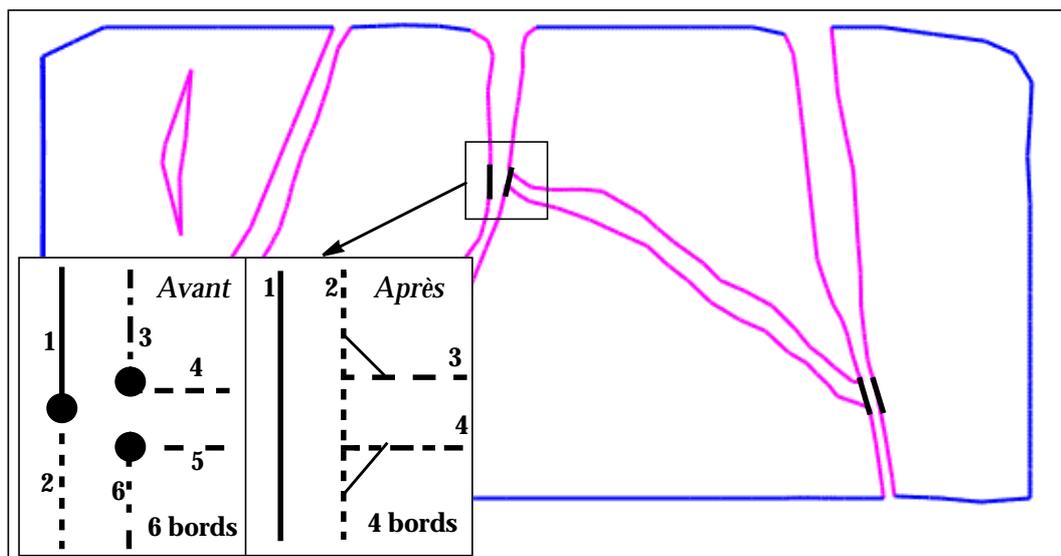


Figure 8-22 : Rassemblement de bords pour obtenir un résultat plus géologique.

#### d) Contacts qui ne sont pas encore pris en compte automatiquement

De la même manière que l'étude des coupes géologiques ne permet pas de détecter tous les contacts, l'étude des bords d'horizons seule ne peut permettre de tout détecter. Tous les contacts en T avec une ligne de contact plus ou moins parallèle aux bords d'horizons ne sont pas détectables de cette manière. Parmi eux, on peut souligner le contact Faille-Faille horizontal, et le contact entre deux horizons (contact dû à une transgression ou une régression marine par exemple)

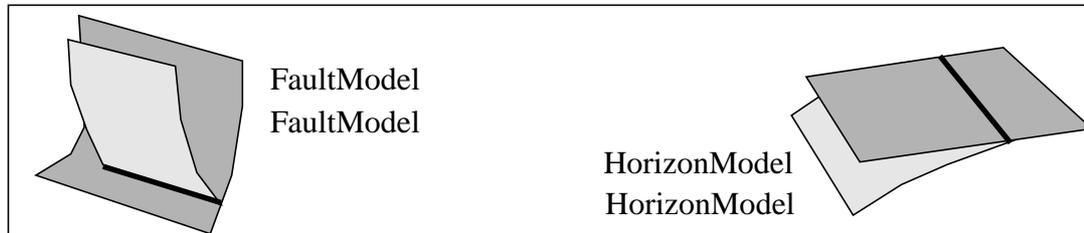


Figure 8-23 : Contacts qui ne sont pas détectés actuellement.

Cependant, comme cela a été souligné plus haut, les structures sont prévues pour que ces contacts soient définis soit par intervention de l'utilisateur, soit par un autre procédé automatique (analyse de coupes géologiques ou opération classique de découpage des surfaces triangulées entre elles).

### 3. Construction de modèles volumiques à l'aide de surfaces triangulées «pré-découpés»

Maintenant que tous les éléments sont définis, cette partie décrit la manière dont ils sont calculés à partir de tous les types de données possibles, afin de construire un modèle volumique sans fuite. Cette construction se fait en 2 parties [figure 8-24]:

1. une première série d'algorithmes permet d'arriver à un point commun quelles que soient les données: la construction de **surfaces triangulées** représentant les horizons et les failles; et la **détection des contacts** entre ces différentes interfaces
2. la deuxième série d'algorithmes est commune, quelles que soient les données de départ. Elle se décompose en la création de la **boîte englobante**, puis l'**installation automatique des contraintes**, l'**interpolation** des surfaces de faille puis des surfaces d'horizon, la **reconstruction de surfaces de faille** incluant tous les contacts les concernant (**pré-découpage**), pour finir par la construction du **modèle volumique**.

Données	Failles: points, lignes, surfaces objets faille	Horizons: points, lignes, surfaces
Phase 1	1. Creation des <b>surfaces</b> de failles et d'horizons	2. Definition des <b>contacts</b>
Phase 2	3. installation de <b>contraintes</b> sur les bords (failles et horizons)	4. <b>Interpolation</b> des failles puis des horizons (fermeture géométrique)
	5. <b>Retriangulation des failles</b> afin d'intégrer les lignes d'intersection dans leur maillage	6. Creation du <b>modèle</b> 7. <i>Modification de la géométrie des failles (et des horizons ?)</i>

Figure 8-24 : Schéma général de construction du modèle

### 3.1 Phase 1: création de toutes les surfaces triangulées et détection des contacts

#### a) A partir des horizons seuls

Si les données initiales ne sont composées que par des horizons, la première phase consiste à créer d'abord les représentations ligne et surface des horizons. La détection des contacts est réalisée selon la méthode détaillée précédemment [102 - Duvinage 2].

Cette détection des contacts permet de rassembler tous les bords d'horizons correspondant à une même faille, de construire les lignes médianes entre toutes les paires de bords d'horizons, puis de construire les objets faille correspondants. Il ne reste plus ensuite qu'à construire les surfaces de faille, après avoir éventuellement étendu les objets faille à leurs extrémités [figure 8-25].

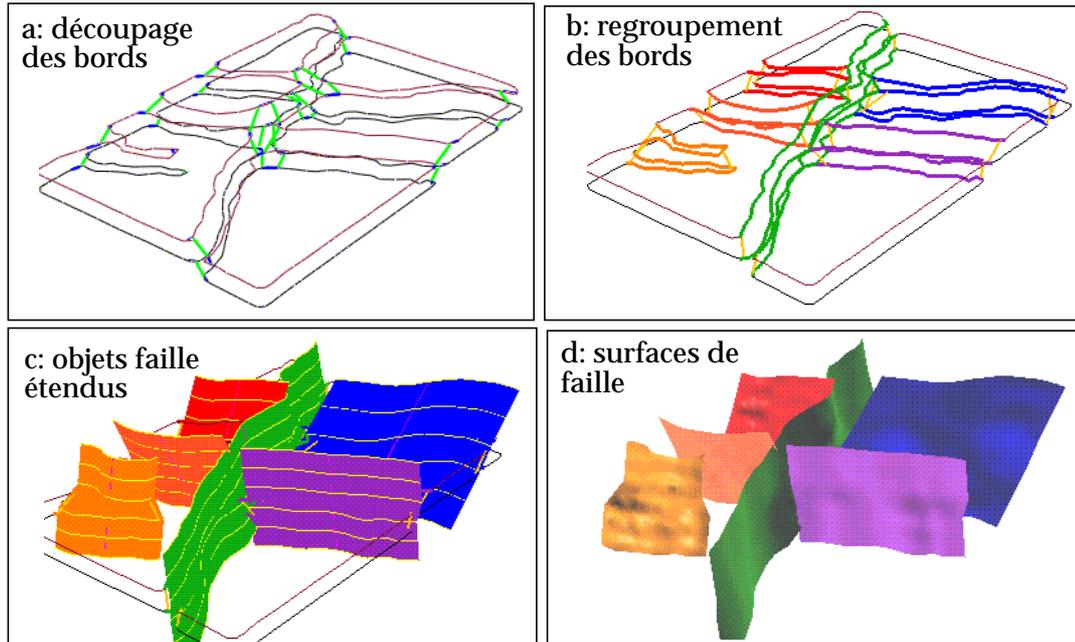


Figure 8-25 : phase 1 réalisée en partant d'horizons uniquement: détection des liens [a]; regroupement des bords de faille [b]; construction des objets faille étendus [c]; construction des surfaces de failles [d] (données fournies par la compagnie EXXON et traitées par Isabelle Duvinage).

#### b) A partir des failles seules

Lorsque seules des données sur les failles sont fournies, les contacts sont calculés en utilisant des critères de distance [104 - Dulac]. Ce critère ne permet de détecter que les contacts en T, et nécessite de donner une distance au delà de laquelle deux failles ne seront pas considérées comme connectées [figure 8-26].

Il est rare d'avoir de telles données dans l'industrie. Par contre, c'est un moyen de construire d'abord un réseau de failles bien fermé, puis de passer à la construction du modèle en utilisant une méthode plus classique (par découpage des surfaces d'horizons à l'aide de ce réseau de failles connectées). Cette méthode est à l'étude.

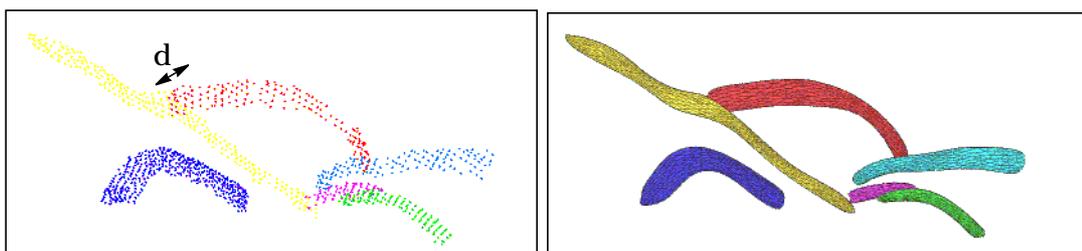


Figure 8-26 : En partant de failles uniquement (critère de distance).

### c) A partir des deux types de données

C'est le cas idéal, et heureusement le plus courant. Dans ce cas, cette phase 1 consiste à construire les surfaces triangulées correspondant aux horizons et aux failles. Les bords des horizons sont découpés et corrélés horizon par horizon, comme cela a été expliqué dans la section précédente.

Par contre, l'installation des liens permettant de rassembler les bords d'horizons entre des horizons différents n'utilise pas de tessellation. Une ligne est tracée à partir du milieu de chaque bord d'horizon en direction de toutes les failles du modèle. Le bord est ensuite ajouté dans la liste de bords correspondant à la faille dont il est le plus proche (à condition de ne pas dépasser la distance seuil).

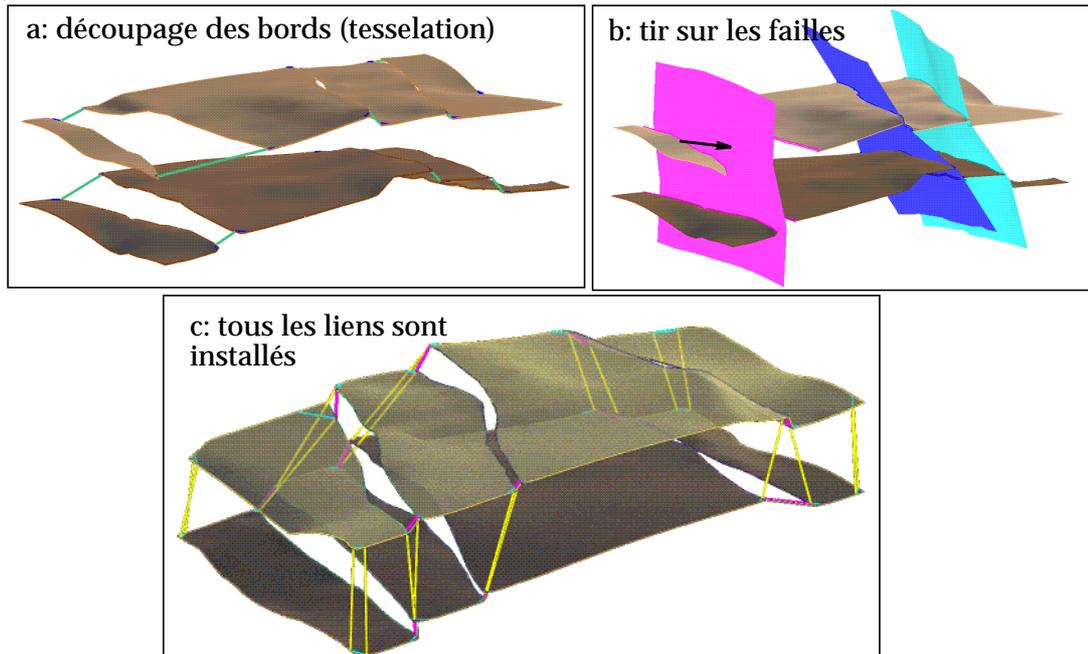


Figure 8-27 : En partant des deux types de données (cas idéal).

## 3.2 Phase 2: construction des surfaces pré-découpées

Quelles que soient les données fournies au départ, le résultat de la phase 1 est donc toujours le même: des surfaces triangulées représentant les horizons et les failles, ainsi que la détection et l'installation des liens matérialisant les contacts du modèle. A partir de là, les algorithmes deviennent communs.

### 3.2.1 Installation automatique des contraintes

A partir de tous les liens qui ont été installés et des structures correspondantes (INTERFACEBORDER, CONTACT2D, INTERFACEBORDEREXTREMITY et CONTACT1D), il est possible d'installer automatiquement des contraintes sur les bords et sur les extrémités de ces bords, en fonction du type de contact auquel ils appartiennent.

Cette étape est complètement automatisée, et se produit globalement sur le modèle, mais une fois de plus, l'utilisateur peut modifier le résultat automatique, en utilisant des commandes classiques d'ajout et de destruction de contraintes.

Même si le résultat n'est pas toujours parfait du premier coup, la solution initiale proposée diminue de beaucoup le temps passé à installer les contraintes qui seront utilisées ensuite par l'interpolateur.

### a) A partir des Contacts 2D

Comme cela a été annoncé dans la description des différents types de contact, les contacts 2D permettent d'installer la contrainte correspondant au type de contact:

- \* *bord de la surface contrôlée sur la surface contrôleur* pour les contacts en T
- \* *attirance par la ligne commune* pour les **contacts multiples**
- \* *attirance par la ligne commune projetée sur la surface principale* pour les **contacts multiples se terminant en T**

### b) A partir des Contacts 1D

Tous les contacts 1D (repérés grâce aux liens de *branchement entre deux failles*) permettent d'installer une contrainte "coin de bord sur bord de surface" sur les extrémités de bord d'horizons concernés.

### c) Autres contraintes

Pour tous les bords qui n'ont pas été contraints (c'est-à-dire qui ne font pas partie d'un contact), une contrainte n'autorisant leurs points à se déplacer que selon un plan localement perpendiculaire au bord est installée. Cette contrainte évite que les bords non contraints ne se rétractent vers le centre de la surface sous l'effet de l'interpolation (qui tend à éliminer les points anguleux).

Enfin, on pourrait ajouter des nœuds de contrôle (qui ne sont pas autorisés à bouger du tout) pour tous les points situés au delà d'une certaine distance du bord. Ainsi, on serait sûr de respecter au mieux les données de type "ensemble de points attirant la surface" lors de l'interpolation.

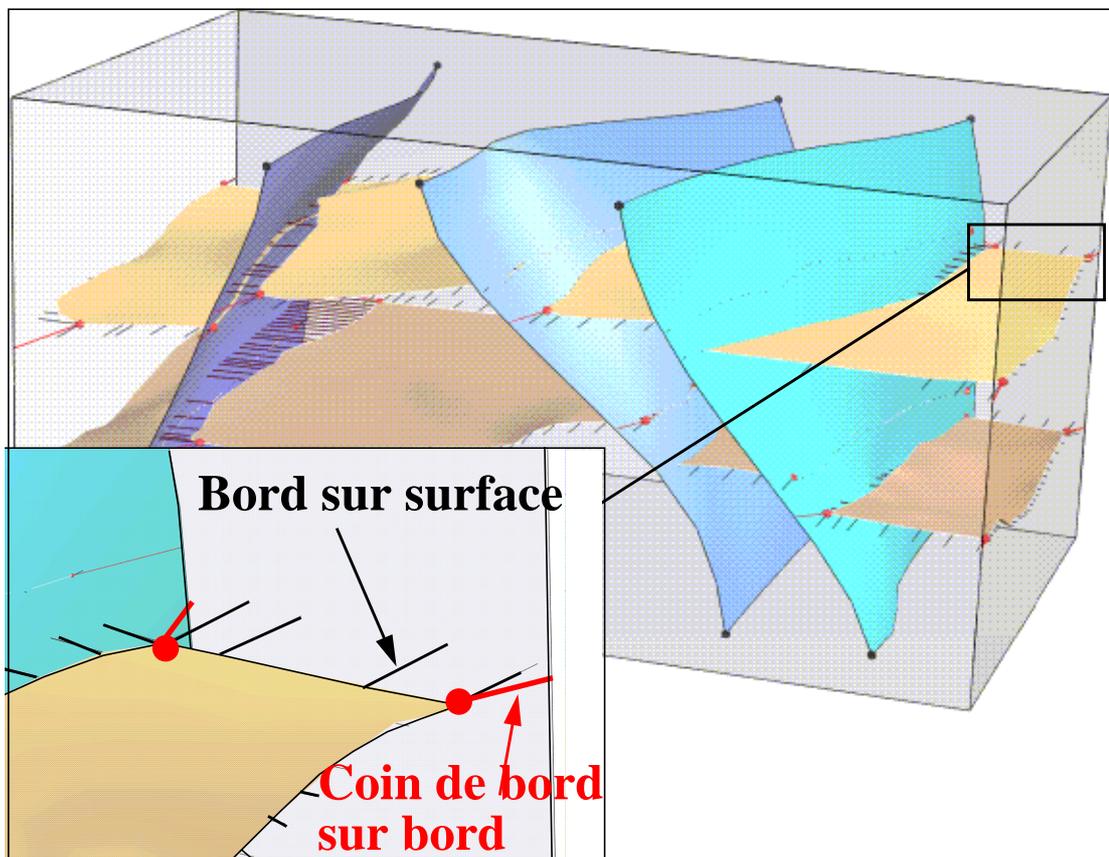


Figure 8-28 : Installation des contraintes sur les bords à l'aide de l'information stockées dans les différents contacts (1D et 2D)

### 3.2.2 Construction de surfaces de failles pré-découpées

Cette construction se fait en utilisant la méthode présentée au chapitre précédent (Voir “Construction de surfaces à partir de courbes frontières” page 169.). Cette construction est basée sur l'utilisation de toutes les lignes qui composent la faille: ses quatre bords, les contacts avec les horizons, les contacts avec les autres failles. Ces lignes sont utilisées pour construire un objet de type “*squelette de surface*” qui détermine les différentes régions à trianguler (voir figure 7-17, page 171).

#### a) Construction des différents groupes de lignes

Dans un premier temps, toutes les failles sont passées en revue. Pour chacune d'entre elles, les bords externes sont copiés sous forme de ligne polygonale dans une liste que nous appellerons “*liste des lignes du squelette de surface*” ou *LISTE*.

Dans un deuxième temps, la liste des failles est à nouveau passée en revue. Pour chaque faille, (F1 pour fixer les idées), la liste des CONTACT2D stockée dans le FAULT-MODEL correspondant est parcourue. Plusieurs cas peuvent se produire:

- le contact est un **contact en T** entre *un horizon et la faille F1* (HORIZONFAULTCONTACT). Dans ce cas, les deux bords de l'horizon en contact avec la faille sont inclus dans la liste *LISTE*
- le contact est un **contact en T** (simple ou multi\_T) *entre plusieurs failles*, et la faille F1 *intervient en tant que “faille principale”*. Dans ce cas, la ligne d'intersection (bord de la faille secondaire ou ligne commune pour un contact multiple) est incluse dans la liste *LISTE*
- le contact est un **contact en T**, et la faille F1 *est une faille secondaire*: rien n'est fait (la ligne de contact est déjà prise en compte dans le bord de la faille F1).
- le contact est un **contact multiple**: la *ligne commune* entre toutes les failles en contact est incluse dans la liste *LISTE*.

A chaque fois qu'une ligne de contact est partagée entre plusieurs objets, la propriété des nœuds “*partage du vertex entre les deux objets*” est utilisée. Cela signifie qu'un seul nœud est créé, et qu'il est utilisé par le maillages des deux objets à la fois (bord d'horizon et squelette de la faille F1, bord d'une faille secondaire et squelette de la faille F1, bords des failles d'un contact multiple et squelette de la faille F1...).

Ainsi, on est sûr que du point de vue géométrique, **le contact sera parfait**, puisque ce sont exactement les mêmes coordonnées (x,y,z) qui sont utilisées pour les deux objets. Puisque la triangulation suit les lignes du squelette, cette propriété est aussi valable pour les surfaces du modèle volumique [figure 8-29].

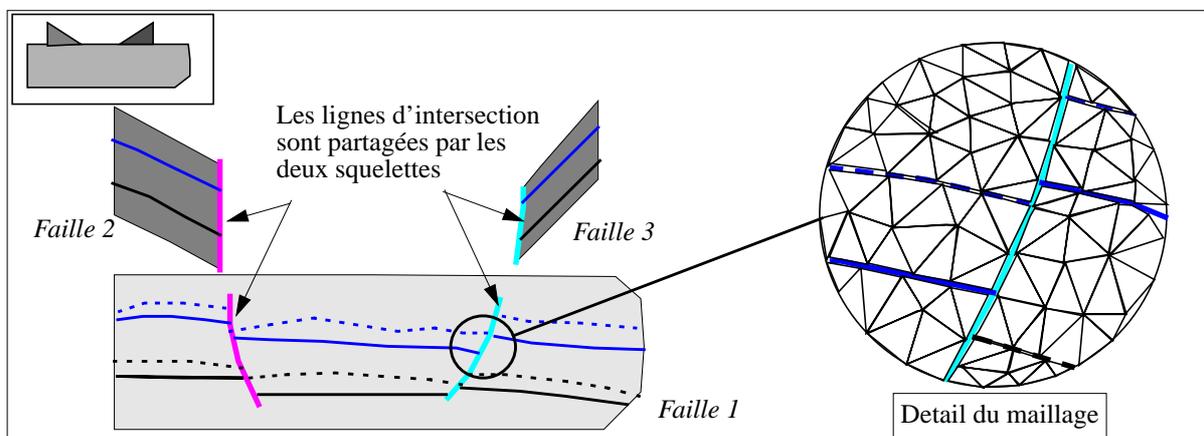


Figure 8-29 : Partage des points du contact entre les différents objets.

### b) Découpage au niveau des intersections

Avant de construire les surfaces triangulées «pré-découpées», il faut encore une étape: découper les lignes du squelette au niveau de leurs intersections (intersection entre une faille et un horizon la plupart du temps). En effet, deux segments peuvent s'intersecter «visuellement», mais, du fait des imprécisions numériques, le contact peut ne pas être parfait du point de vue géométrique.

Une des manières d'assurer ce contact parfait, est de créer un point unique au niveau de l'intersection, de découper les deux segments s'intersectant au niveau de ce point. Ainsi de deux segments, on en crée quatre, ayant chacun une extrémité en commun: le point d'intersection [figure 8-30].

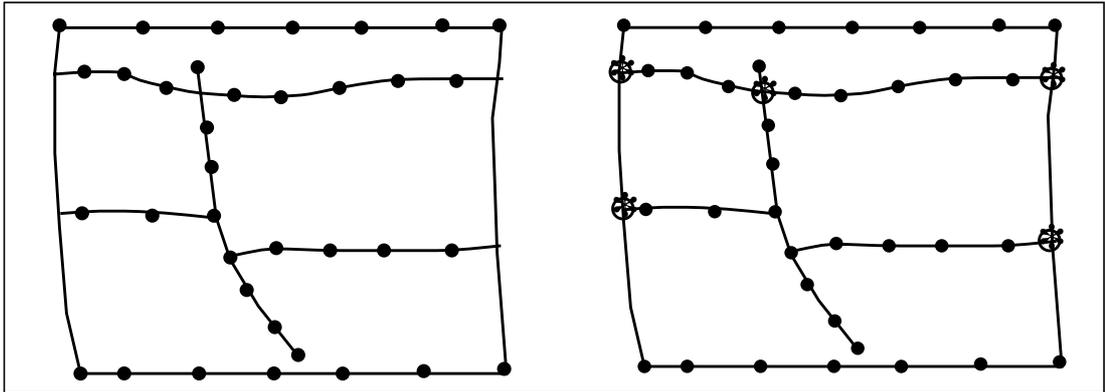


Figure 8-30 : Découpage des lignes constituant le squelette de faille au niveau de leurs intersection.

### c) Construction des surfaces de faille «pré-découpées»

Une fois que les squelettes de faille ont été créés, et que l'on est sûr que tous les points d'intersection sont partagés entre les différents objets, il est possible de créer les surfaces de faille qui vont servir à la construction du modèle volumique [figure 8-31].

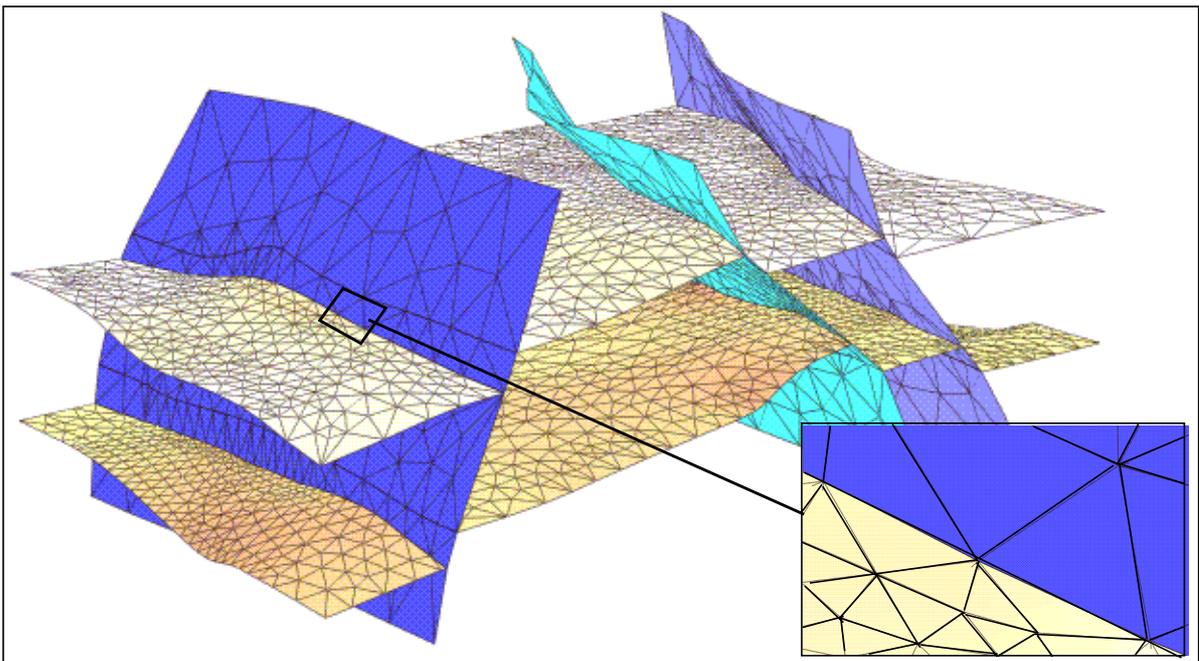


Figure 8-31 : Surfaces pré-découpées, partageant exactement les lignes d'intersection.

### 3.2.3 Construction du modèle volumique

#### a) Construction du modèle à l'aide des surfaces pré-découpées

Une fois que toutes les surfaces sont construites, et que les points correspondant aux lignes de contact sont bien partagés, la construction du modèle est très rapide, et n'est pas sujette aux imprécisions numériques (puisque le contact est assuré géométriquement par construction).

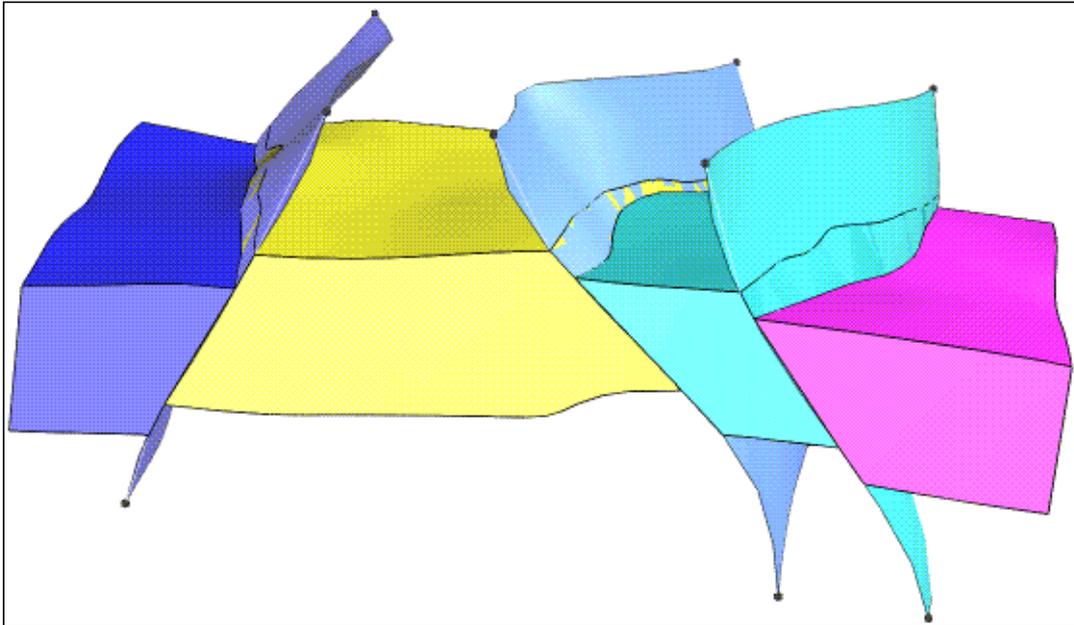


Figure 8-32 : Modèle construit à partir des surfaces pré-découpées.

#### b) Problèmes actuels posés par cette méthode

Si cette méthode marche bien en théorie, il reste encore quelques problèmes à résoudre. La majeure partie de la détermination des contacts est effectuée de manière topologique, mais l'interpolation des surfaces posait encore jusqu'il y a peu des problèmes numériques. Des améliorations ont été apportées concernant certaines contraintes, et les problèmes dus aux contraintes ont quasiment disparu.

Le deuxième problème concerne la triangulation des horizons. Si le bord de l'horizon est un peu irrégulier, et que trois points consécutifs du bord appartiennent au même triangle, l'interpolation donne une surface d'horizon avec un triangle plat (à cause du respect de la contrainte «bord sur surface») [figure 8-33]. Ce cas de figure n'est pas valide lors de la construction du modèle volumique.

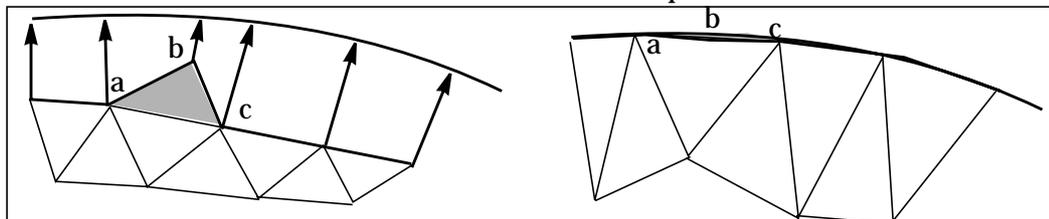


Figure 8-33 : Formation d'un triangle plat après interpolation si trois points d'un triangle du bord sont situés sur le bord.

- Pour résoudre ces problèmes, deux solutions très différentes sont envisagées:
- à court terme, il est prévu de mixer la méthode de fermeture du réseau de failles, et la méthode de découpage des horizons par ce réseau de faille fermé

- à plus long terme., il est prévu de travailler sur le maillage des horizons, afin de revenir à la méthode initiale.

### c) Solution envisagée à court terme

Pour cette solution, le travail se ferait en deux étapes:

1. **assurer d'abord le contact entre les failles.** Cette partie consisterait à détecter les contacts entre failles en utilisant un critère de distance, à installer les contraintes correspondantes, puis à interpoler toutes les failles afin d'assurer le contact géométrique.
2. ces surfaces de faille seraient ensuite utilisées pour **couper les horizons.** On rejoint ici la méthode classique, améliorée par Nicolas Euler, qui permet d'installer automatiquement les contraintes entre les horizons et les failles au moment du découpage [105 - Euler 1].

Le risque, d'une telle méthode est de pouvoir détecter beaucoup moins de cas durant la première partie (puisque cette détection se fait sur des critères plus géométriques que topologiques). De plus, les risques de problèmes numériques ne sont pas complètement éliminés, même si les récents travaux pour améliorer l'installation des contraintes et le résultat de l'interpolation, ainsi que le travail de Nicolas Euler permettent d'obtenir de bien meilleurs résultats qu'au démarrage de ce travail de thèse.

### d) Solution envisagée à plus long terme

Un autre solution, est de travailler sur la détection des anomalies du maillage des horizons, afin d'éliminer les cas non valides pour la construction du modèle.

Par exemple, il est assez facile de détecter le cas de figure où trois points d'un même triangles sont sur le bord. Une solution, consiste ensuite à échanger l'arête entre le triangle anomalous et son voisin contre l'arête qui lui est perpendiculaire [figure 8-34] (gauche). Mais si l'on utilise cet algorithme tant qu'il reste au moins un triangle anomalous, on risque d'entrer dans des boucles infinies [figure 8-34] (droite).

Il faut donc un gros travail, afin de déterminer tous les cas anomalous, et les moyens de les éliminer (par exemple en modifiant le bord lorsque l'on arrive dans un cas «boucle infinie»).

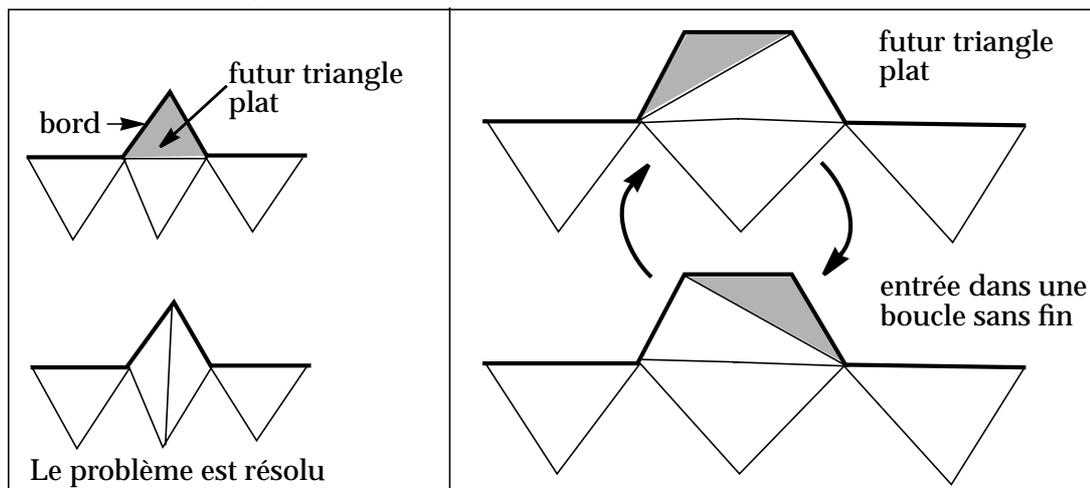


Figure 8-34 : L'élimination des triangles plat n'est pas un problème trivial (problème repéré par Nicolas Euler durant sa thèse).

## 4. Conclusion sur cette structure

La structure décrite dans ce chapitre a comme principal avantage de stocker de manière *ordonnée* les différents objets du modèle, ce qui permet de travailler de façon plus *globale*, et plus «géologique».

Le deuxième avantage de cette structure est de pouvoir *stocker tous les contacts* du modèle (qu'ils aient été détectés de manière automatique à partir des horizons, de manière automatique par la méthode de découpage sous contrainte, ou ajoutés à la main par l'utilisateur).

Ceci laisse prévoir la *modélisation rapide de plusieurs scénarios* de réseau de faille à partir du même modèle initial. Cet autre aspect des incertitudes structurales concernant les failles est développé dans le dernier chapitre de ce mémoire.

Enfin, le stockage des contacts permet d'*installer automatiquement toutes les contraintes* qui permettent de fermer complètement le modèle surfacique, puis de construire le modèle volumique.

C'est pour cela que cette structure est très utile pour la dernière partie de ce travail: la *simulation de la géométrie d'un réseau de failles* (à scénario constant), et le *calcul des volumes* correspondant à chaque simulation.



---

---

## Chapitre 9 Etudes volumétriques au niveau du réservoir: structures mises en jeu

---

---

Dans les chapitres précédents, les différents modules nécessaires à l'étude des variations de volumes d'un modèle structural lorsque la géométrie des failles est modifiée ont été présentés:

- La construction d'une faille de référence et le stockage des incertitudes associées
- les méthodes de simulation de la géométrie d'une faille isolée
- la mise à jour d'un modèle structural dont les failles ont été modifiées

La dernière étape consiste à rassembler tous ces modules indépendants, et à fournir une interface permettant à l'utilisateur de tester différents types de simulations, et les variations de volumes associés.

Cet algorithme global se déroule en cinq étapes:

### **1. Phase de préparation**

Cette phase consiste à construire le *modèle volumique* qui sera utilisé comme **référence**, et à calculer les paramètres qui le caractérisent (volumes des différentes régions de référence par exemple).

Le modèle volumique peut être construit de différentes manières (en utilisant la structure présentée au chapitre précédent ou non). Dans tous les cas, il est nécessaire de stocker les relations entre les différentes failles, ainsi que d'installer les contraintes afin de pouvoir mettre à jour le modèle par interpolation après chaque simulation, et ainsi de conserver le schéma structural du modèle initial.

### **2. Phase de simulation proprement dite (processus itératif)**

Chaque tour de boucle se découpe lui-même en quatre parties:

#### a. Initialisation

Afin de réaliser des simulations équiprobables, les points retrouvent les positions du modèle de référence avant toute nouvelle simulation.

#### b. Simulations

Pour l'instant, la géométrie de chacune des failles est simulée de manière indépendante, en utilisant l'une des fonctions aléatoires présentées au chapitre 6. Il serait plus judicieux de tenir compte de la hiérarchie du réseau de failles lors des simulations, et ce point sera discuté dans le chapitre suivant.

#### c. Mise à jour du modèle

Une fois la géométrie des failles modifiée, l'ensemble du modèle est remis à jour: le réseau de failles d'abord, suivi des horizons. Comme les contraintes ont été stockées dans la phase de préparation, il suffit d'une interpolation pour pouvoir reconstruire le modèle volumique.

#### d. Stockage des résultats

La dernière étape, consiste à stocker le *minimum d'informations nécessaires* pour retrouver ce modèle plus tard. Seuls les paramètres utilisés dans la simulation de la géométrie et les volumes correspondants sont conservés.

## 1. Première phase: préparation du modèle

### 1.1 Création du modèle initial

Cette étape a été décrite en détail dans les deux chapitres précédents (voir «Construction de modèles volumiques», page 155 et «Le Structural Model», page 179). Seuls quelques rappels seront faits ici.

Quelle que soit la méthode utilisée pour construire ce modèle de référence, le but de cette étape est non seulement de construire le modèle initial, mais surtout d'installer tout ce qui est nécessaire à la *mise à jour du modèle* par la suite (contraintes géométriques et informations sur les contacts).

#### 1.1.1 Données

Pour simplifier les explications, nous supposons dans ce qui suit que les objets faille, l'estimation des incertitudes associées, ainsi que les surfaces d'horizon sont fournis.

De plus, lorsque la structure «STRUCTURALMODEL» n'est pas utilisée, il faut être capable de retrouver quelles surfaces sont en contact. Nous supposons donc que les contraintes ont déjà été installées, ou qu'un mécanisme permet de les installer de manière automatique (fichier décrivant le réseau de failles par exemple).

#### 1.1.2 Création du modèle de référence à l'aide du StructuralModel

##### a) Ajout des failles dans le modèle

La création d'un objet de type STRUCTURALMODEL à partir des différents objets faille entraîne la création d'un FAULTMODEL par faille. Au départ, chaque FAULTMODEL ne contient que la *représentation «Objet Faille»* dans sa liste de représentations.

Comme l'installation des contraintes est plus complète sur les surfaces triangulées (plus de contraintes disponibles, structures moins lourdes à gérer), la *représentation surface* est automatiquement ajoutée à la liste de représentations [figure 9-1].

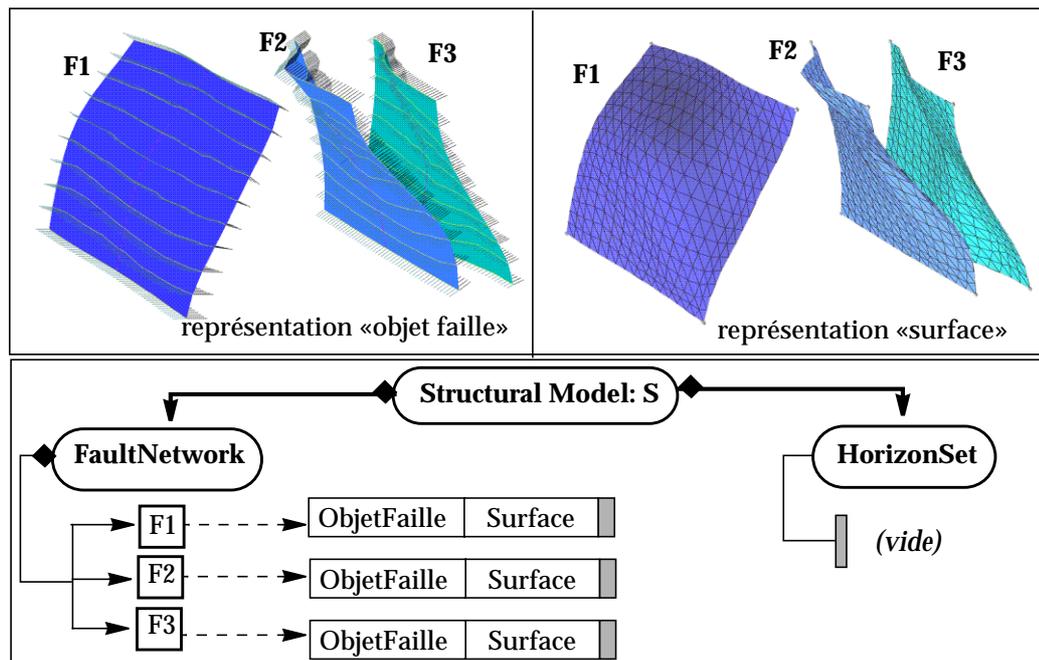


Figure 9-1 : Etat initial du modèle structural (seulement des failles).

## b) Ajout des Horizons

Lorsque les surfaces d'horizon sont ajoutées au modèle, les structures HORIZON-MODEL correspondantes sont créées. Si les horizons sont fournis sous forme de surfaces triangulées, la représentation surface est ajoutée à la liste des représentations, suivie de la représentation ligne (utilisée pour détecter les contacts).

Ensuite, les bords d'horizons sont découpés, classés en bords externes/bords correspondant à des failles, puis appariés horizon par horizon.

Puis, les paires de bords sont corrélées d'un horizon à l'autre, en se servant des surfaces de faille présentes dans le modèle comme guide. Enfin, la représentation ligne des failles est créée (ce sont les mêmes bords que les bords d'horizon, mais classés faille par faille) [figure 9-2].

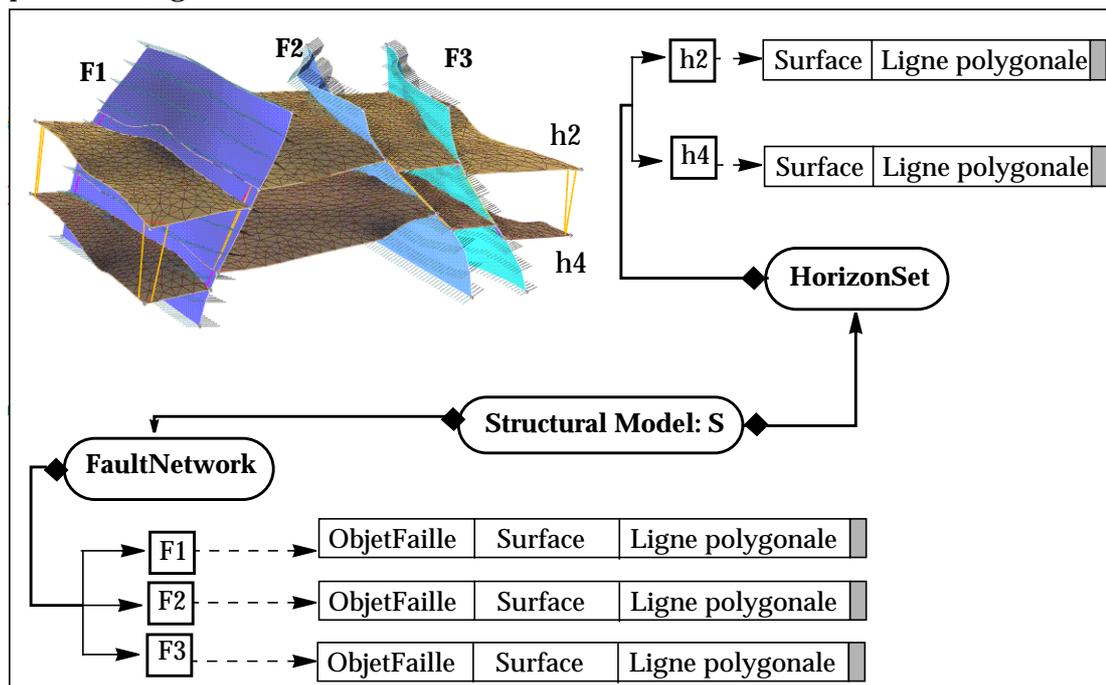


Figure 9-2 : Etat du modèle structural après l'ajout des horizons.

## c) Construction du modèle volumique (ou modèle 3D)

Une fois que les éléments précédents ont été construits, la construction du modèle est réalisée [figure 9-3]:

- Installation des **contraintes** (bord sur surface, coin de bord sur bord, bord pouvant se déplacer le long d'un plan)
- **Interpolation** des différentes surfaces: l'interpolation commence sur les failles, puis les horizons sont interpolés afin de réaliser le contact parfait avec les failles. L'interpolation des failles est réalisée dans l'ordre où les failles ont été ajoutées au modèle. Nous verrons un peu plus loin qu'il serait mieux de tenir compte de la hiérarchie lors de cette phase d'interpolation.
- Construction d'un **squelette de surface** pour chaque faille du modèle. Ce squelette contient toutes les lignes de contact (partagées entre les différents objets en contact).
- Construction des **surfaces de failles pré-découpées** au niveau des lignes de contact
- Construction du **modèle volumique** sans avoir à découper les surfaces (plus stable numériquement)

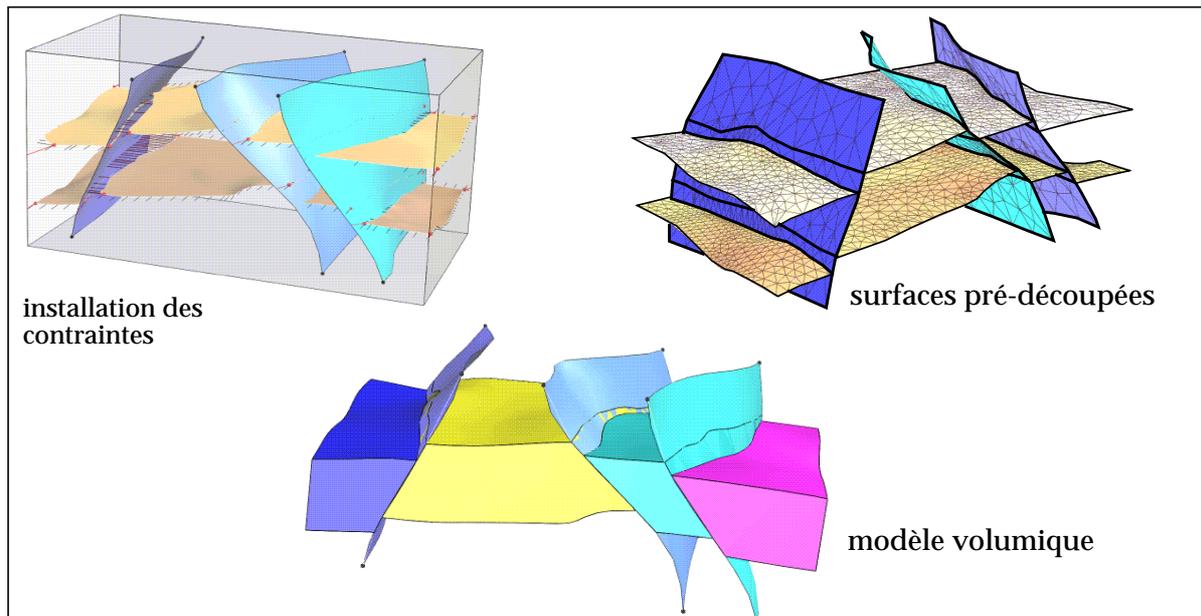


Figure 9-3 : Construction du modèle à partir des surfaces pré-découpées

### 1.1.3 Autres manières de construire le modèle volumique

Avec les améliorations réalisées sur la construction des modèles par un découpage des surfaces guidé par les contraintes [106 - Euler 2], on peut imaginer une détection des contacts à partir des contraintes, et la construction du modèle sans pré-découper les surfaces. Ce test est en cours, et les résultats ne sont pas encore disponibles.

La méthode, plus classique, qui a été utilisée au début de la thèse peut aussi être utilisée pour construire le modèle volumique. Elle donne des résultats corrects, mais nécessite de travailler surface par surface, ce qui est loin d'être convivial. De plus les imprécisions numériques restent un problème important avec cette méthode.

L'approche qui consiste à découper les surfaces à partir de critères topologiques nous semble plus appropriée, notamment dans des cas extrêmes où le découpage sous contraintes risque de poser des problèmes (contact multiple notamment).

## 1.2 Caractérisation des régions du modèle de référence

### 1.2.1 Les problèmes à résoudre

#### a) Connaître les volumes extrêmes de chaque région (minimum et maximum)

Dans la deuxième phase, afin de visualiser les résultats, un histogramme par région du modèle volumique est construit. Ces histogrammes permettent de visualiser l'évolution des volumes des régions, au cours des différentes simulations [figure 9-4].

Si le modèle comporte des régions dont le volume est très différent (différence de quelques puissance de 10 entre les régions les plus grandes et les régions les plus petites), il peut être utile de connaître une estimation de la valeur minimum et de la valeur maximum du volume pour chacun des histogrammes, afin d'adapter l'échelle de l'axe horizontal lors de la construction de ces derniers.

En outre, cette connaissance des volumes minimum et maximum peut suffire, pour réaliser une étude préliminaire par exemple. Elle a donc été déconnectée du reste de l'algorithme de simulation.

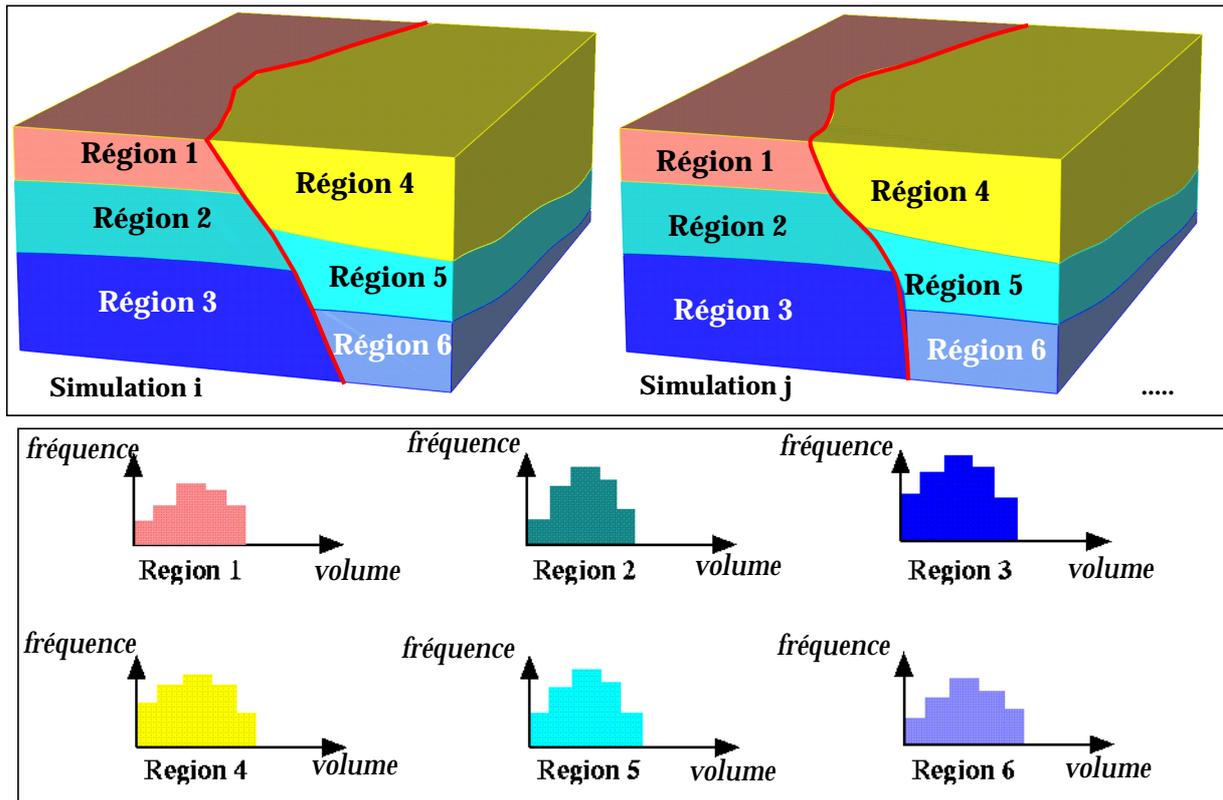


Figure 9-4 : Suivi des volumes des régions au cours des simulations

**b) Etre sûr d'associer toujours la même région avec un histogramme donné**

A chaque fois que le modèle est reconstruit, l'ordre des régions change, même si les surfaces qui servent à le construire restent les mêmes. La numérotation des régions ne se correspond donc pas forcément d'une simulation à l'autre, ce qui pose un gros problème pour relier le calcul du volume d'une région donnée à l'historgramme qui lui correspond.

La solution que nous avons adoptée consiste à calculer pour chacune des régions du modèle de référence un marqueur. Ce marqueur (REGIONMARKER) est calculé de manière à ce qu'il soit toujours situé dans le volume minimum de la région. Une table permet d'associer chaque marqueur à la région du modèle de référence qui lui correspond. Après chaque simulation, ces marqueurs sont utilisés pour ordonner les régions du modèle courant de la même manière que le modèle de référence.

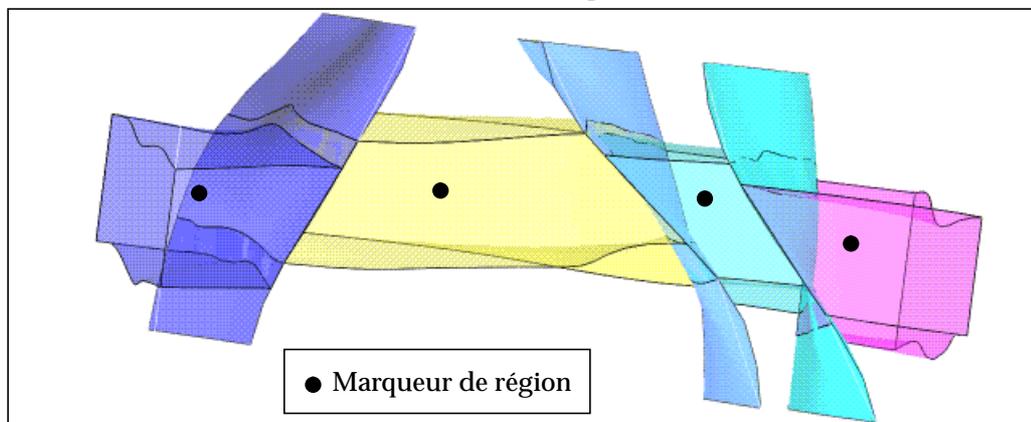


Figure 9-5 : Calcul d'un marqueur par région, restant dans la région quelle que soit le résultat de la simulation.

## 1.2.2 Calcul des marqueurs de région

### a) Isobarycentre

La première méthode utilisée pour calculer le marqueur d'une région donnée est assez rapide mais peu précise. Elle consiste à calculer l'isobarycentre de tous les points composant les faces entourant la région donnée.

Cette méthode est très rapide, mais dépend énormément du maillage. Comme il n'y a aucune pondération sur les points, si l'une des surface a un maillage beaucoup plus dense que les autres, le marqueur en sera plus proche. Cependant, si les faces sont homogènes (environ la même densification pour toutes les faces de la région) et que les maillages sont comparables, cette méthode est valable.

$$\text{marqueur} = \frac{1}{n} \cdot \sum_{\text{point} = 1}^n \text{point}$$

### b) Barycentre pondéré à l'aide d'une tessellation

Si on ne se trouve pas dans le cas précédent, il est possible de faire un vrai calcul de barycentre. Il faut d'abord tesseler la région à l'aide de tétraèdres. Ensuite, le barycentre de chaque tétraèdre est calculé et pondéré par le volume du tétraèdre dont il est le barycentre. Cette méthode permet de faire un calcul plus précis, mais prend plus de temps.

$$\text{marqueur} = \frac{1}{n} \cdot \sum_{\text{tetra} = 1}^n \text{volume}(\text{tetra}) \cdot \text{barycentre}(\text{tetra})$$

### c) Comparaison des deux méthodes

La première comparaison que l'on peut faire concerne les temps de calcul du marqueur, en fonction du nombre de points dans la région. Ces temps ont été calculés sur une silicon graphics (O2, processeur R5000).

Table 9-1 : Temps de calcul en fonction du nombre de points

nombre de triangles dans le modèle	Isobarycentre	Tessellation
cas 1: 6 427 triangles	moins d'1 seconde	6 secondes
cas 2: 25 836 triangles	moins d'1 seconde	26 secondes

Le deuxième type de comparaison que l'on peut faire, concerne la précision du point calculé avec les deux méthodes. La figure ci-dessous montre une comparaison des deux marqueurs, en coupe verticale et en vue de dessus [figure 9-6].

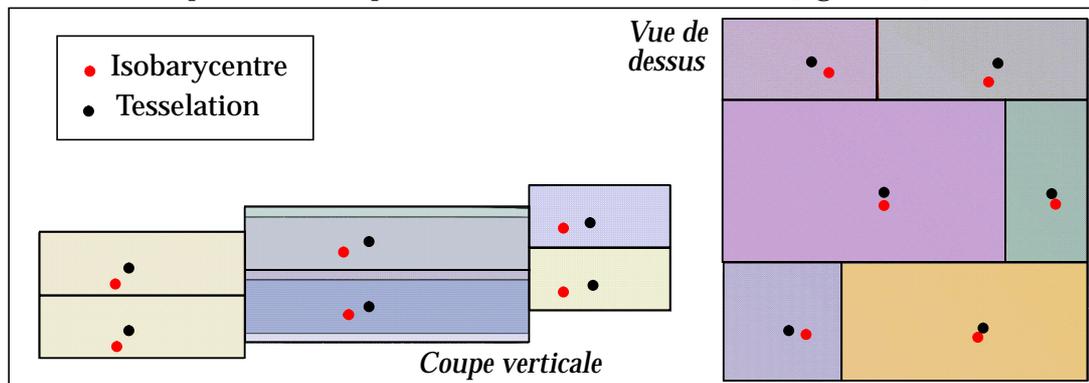


Figure 9-6 : Calcul des marqueurs sur un modèle synthétique cubique.

#### d) Autre méthode (non implantée)

Ces méthodes ne sont valables que si le marqueur ainsi calculé se trouve toujours à l'intérieur de la région à laquelle il est associé, quel que soit le résultat de la simulation. Si la région est petite et présente une concavité, ce critère peut ne pas être respecté.

Une solution pourrait être de trouver un critère qui permet de classer les régions du modèle toujours dans le même ordre. Malheureusement, un tel critère est difficile à déterminer:

- les **critères géométriques** (classer les régions suivant l'axe z, puis suivant l'axe x, puis suivant l'axe y) ne sont pas satisfaisants (trop de cas particuliers apparaissent pour que ces critères soient fiables)
- les **critères topologiques** (utilisant les relations entre les faces entourant les régions) ne sont pas utilisables, puisque d'un modèle à l'autre, les faces ne sont pas conservées. Chaque modèle est composé d'un ensemble de surfaces différent du précédent, et aucun point de repère ne permet de classer les régions de cette manière.

C'est pourquoi, en attendant de trouver mieux, nous nous sommes limités au calcul du barycentre, qui reste valable dans la majorité des cas (et de tous ceux que nous avons eu à traiter jusqu'à présent).

### 1.3 Analyse volumétrique rapide des régions

Pour faire une analyse rapide des régions du modèle volumique de référence (accompagné des estimations d'incertitude), nous proposons le calcul de trois volumes caractéristiques: le *volume de référence*, le *volume minimum*, et le *volume maximum*.

#### a) Calcul du volume de référence

Dans la suite, on pose  $V$  le volume d'une région,  $S(V)$  l'ensemble des surfaces limitant cette région, et  $T$  un triangle quelconque de l'une des surfaces de  $S(V)$ , orienté de l'intérieur vers l'extérieur de  $V$ .

Pour calculer le volume de la région de manière exacte, on peut utiliser la formule de Gauss-Ostrogradsky:

$$\iiint_V \left( \frac{\partial P}{\partial x} + \frac{\partial Q}{\partial y} + \frac{\partial R}{\partial z} \right) dx dy dz = \iint_{S(V)} (P dy dz + Q dx dz + R dx dy)$$

En effet, on peut remarquer la relation suivante:

$$\frac{\partial P}{\partial x} + \frac{\partial Q}{\partial y} + \frac{\partial R}{\partial z} = 1 \Rightarrow \iiint_V \left( \frac{\partial P}{\partial x} + \frac{\partial Q}{\partial y} + \frac{\partial R}{\partial z} \right) dx dy dz = \iiint_V dx dy dz = |V|$$

Ainsi, si l'on prend les fonctions particulières:  $P = Q = 0$  et  $R = z$ , on obtient:

$$|V| = \iint_{S(V)} z dx dy = \sum_{T \in S(V)} \iint_T z dx dy$$

Il ne reste plus qu'à calculer  $\iint_T z dx dy$ , afin d'obtenir le volume de la région courante.

Ce calcul est largement détaillé dans la thèse de Yung-Guao Huang [114 - Huang], et seul le résultat du calcul est donné ici. On pourra se reporter à la thèse pour plus de précisions.

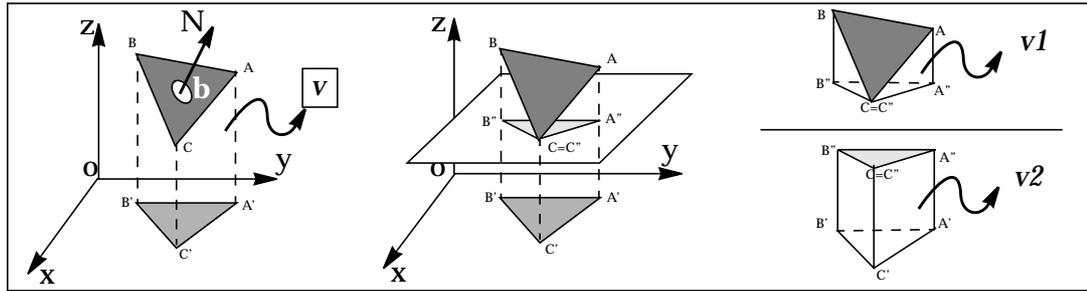


Figure 9-7 : Calcul de l'intégrale associée à un triangle quelconque de la région: décomposition du volume correspondant en deux sous-volumes.

La valeur de l'intégrale pour un triangle quelconque de la surface englobante de la région (ABC pour fixer les idées) correspond au volume (noté  $v$ ) situé entre le triangle ABC et sa projection verticale  $A'B'C'$  sur le plan Oxy [figure 9-7].

Si l'on considère le plan P horizontal, passant par le point du triangle ABC d'altitude minimum (C pour fixer les idées), le volume  $v$  peut se décomposer en deux parties: une pyramide de base  $ABB''A''$  (de volume  $v1$ ), et un prisme droit de base triangulaire  $A''B''C''$ .

Si l'on utilise les formules donnant le volume de ces deux figures géométriques, on obtient:

$$v1 = \frac{\text{Aire}(ABB''A'') \cdot h}{3} \text{ et } v2 = |CC'| \cdot \text{Aire}(A''B''C'')$$

Tous calculs faits (voir [114 - Huang ]), le volume est donc obtenu par:

$$v = v1 + v2 = \frac{1}{3}(|BB''| + |AA''| + 3|CC'|) \cdot \text{Aire}(A''B''C'')$$

Or, si l'on considère la normale  $\vec{N}$  du triangle ABC,  $\vec{N}_z$  sa coordonnée selon  $z$  et  $b$  son isobarycentre, on a aussi les relations:

$$\text{Aire}(A''B''C'') = \frac{|(xA - xC)(yB - yC) - (xB - xC)(yA - yC)|}{2} = \frac{((\vec{CA} \wedge \vec{CB})_z)}{2} = \vec{N}_z$$

et (en utilisant le fait que  $A''B''C''$  et  $A'B'C'$  sont parallèles:  $|A''A'| = |B''B'| = |C''C'|$ ):

$$\begin{aligned} (|BB''| + |AA''| + 3|CC'|) &= |BB''| + |AA''| + |C''C'| + |B''B'| + |A''A'| \\ &= |BB'| + |AA'| + |CC'| = zA + zB + zC \end{aligned}$$

Finalement, on obtient le calcul de la double intégrale associée au triangle ABC:

$$\iint_T z dx dy = \varepsilon \left| \frac{1}{3} (zA + zB + zC) \cdot \frac{((xA - xC)(yB - yC) - (xB - xC)(yA - yC))}{2} \right|$$

avec

$$\varepsilon = 1 \text{ si } \vec{N} \cdot \vec{oz} > 0$$

$$\varepsilon = -1 \text{ si } \vec{N} \cdot \vec{oz} < 0$$

On peut aussi l'écrire:

$$\boxed{\iint_T z dx dy = zb \cdot \vec{N}_z}$$

Le calcul du volume d'une région se traduit donc par un algorithme très simple, qui donne un résultat précis (à la précision du maillage près):

*Exemple de code 9..1. Calcul du volume d'une région (R)*

```

1  Si nf est le nombre de faces entourant la région R
2  Si nt est le nombre de triangles contenu dans la face courante
3  Si p0, p1, p2 sont les trois points du triangle courant
4  et z0, z1, z2 sont les coordonnées z de ces points
5
6  volume = 0
7  Pour toutes les faces de n=1 à nf
8    Pour tous les triangles de t=1 à nt
9      u= côté(p0, p1)
10     v= côté(p0, p2)
11     aire_selon_xy = 0.5 * (coordonnée z de (u^v))
12     z_barycentre = (z0+z1+z2) / 3
13     volume -= aire_selon_xy * z_barycentre
14

```

### b) Calcul rapide des volumes extrêmes

Ce calcul permet une estimation rapide des volumes minimum et maximum, en connaissant le volume de référence, et les incertitudes associées aux failles entourant la région. Par contre, il est *moins précis* que le calcul précédent.

Cependant, comme ce calcul ne sert qu'à avoir un ordre de grandeur de l'étendue des variations de volume pour la région courante, une approximation suffit.

Pour calculer ces volumes, il faut commencer par sélectionner toutes les faces qui entourent la région. Dans cet ensemble de faces, seules les failles sont sélectionnées (d'où l'intérêt de travailler avec le STRUCTURALMODEL qui sépare dans la structure les failles, les horizons et la boîte englobante).

Pour toutes les surfaces de faille, il faut retrouver l'objet faille correspondant. Une fois encore, le fait d'avoir les deux représentations stockées dans le même FAULTMODEL accélère cette recherche lorsque l'on travaille avec le STRUCTURALMODEL.

Ensuite, seuls les vecteurs d'incertitude correspondant aux faces entourant la région sont sélectionnés [figure 9-8].

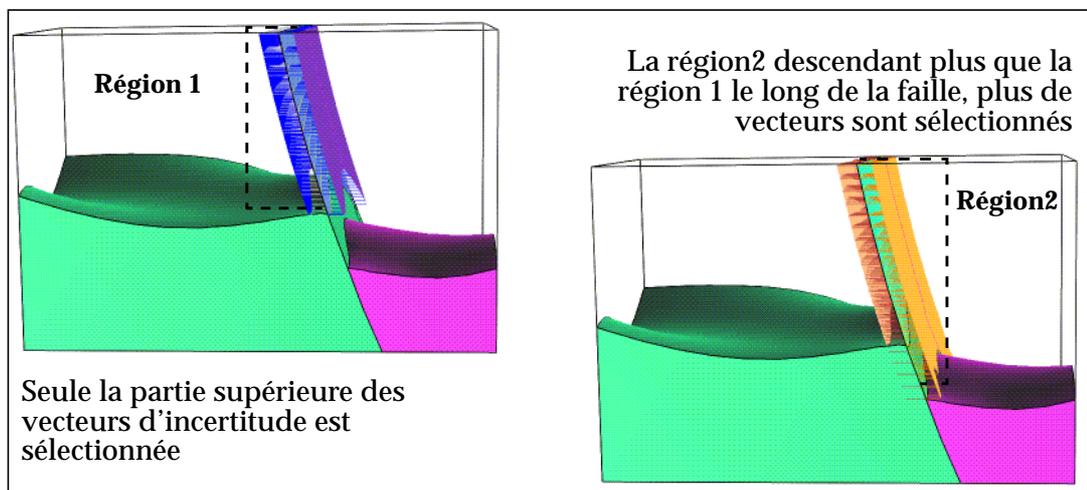


Figure 9-8 : Sélection des vecteurs d'incertitude correspondant à la région.

Dans un deuxième temps, la moyenne des vecteurs d'incertitude à gauche ( $\vec{u}_g$ ) et la moyenne des vecteurs d'incertitude sélectionnés à droite ( $\vec{u}_d$ ) sont calculées.

Si  $n$  vecteurs «droit» et  $m$  vecteurs «gauche» ont été sélectionnés, on obtient:

$$\vec{u}_d = \frac{1}{n} \sum_{i=0}^n \text{vecteur}_{di} \quad \text{et} \quad \vec{u}_g = \frac{1}{m} \sum_{j=1}^m \text{vecteur}_{gj}$$

Les deux volumes gauche ( $v_g$ ) et droite ( $v_d$ ) sont ensuite calculés en multipliant les moyennes  $\vec{u}_g$  et  $\vec{u}_d$  par l'aire de la face qui a servi à les calculer:

$$v_g = \vec{u}_g \cdot \text{Aire}(\text{face}) \quad \text{et} \quad v_d = \vec{u}_d \cdot \text{Aire}(\text{face})$$

Le signe affecté à chaque volume élémentaire est déterminé. Pour ce faire, le vecteur  $\vec{f}_{c,m_r}$  entre le centre de la face ( $f_c$ ) et le marqueur de la région ( $m_r$ ) est calculé. Puis les produits scalaires  $\vec{f}_{c,m_r} \cdot \vec{u}_g$  et  $\vec{f}_{c,m_r} \cdot \vec{u}_d$  déterminent respectivement le signe des volumes  $v_d$  et  $v_g$  (opposé du signe du produit scalaire).

Enfin, les deux volumes extrêmes de la région peuvent être calculés [figure 9-9]:

$$V_{min} = V_{référence} + \sum V_{négatifs}$$

$$V_{max} = V_{référence} + \sum V_{positifs}$$

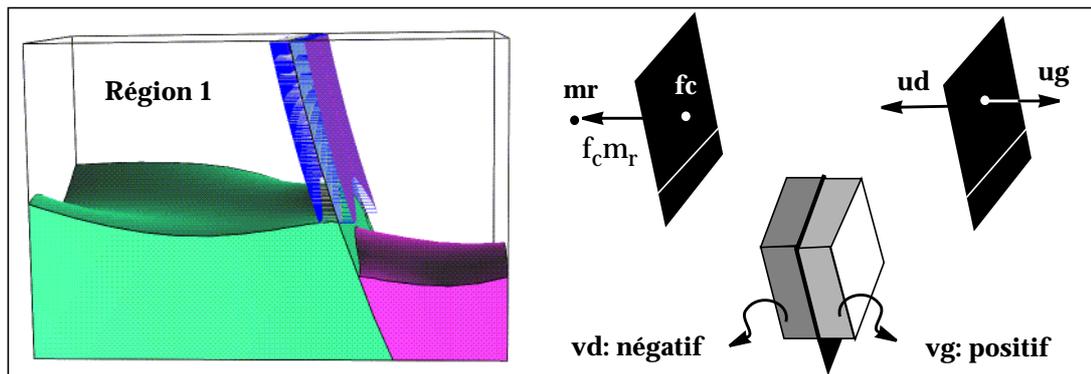


Figure 9-9 : Calcul du volume minimum et du volume maximum de la région. Dans ce cas de figure simple, les variations de volumes ne sont dus qu'à une seule faille.

#### 1.4 Stockage et visualisation des résultats

La plupart des résultats calculés durant cette première phase vont être utilisés par la suite (les marqueurs de région notamment). Une classe est chargée de calculer et de stocker toutes ces informations: le MODEL3DVOLUMEANALYSER. Il associe à chaque région une classe chargée de l'analyser: le REGIONDESCRIPTOR.

Un REGIONDESCRIPTOR associe un *nom* de région, ses *trois volumes caractéristiques* (minimum (approximation), référence (exact) et maximum (approximation)), ainsi que le *marqueur* de région correspondant (celui-ci pouvant être calculé de deux manières: par un isobarycentre ou à l'aide d'une tessellation de la région par des tétraèdres).

Les résultats calculés par ces REGIONDESCRIPTOR sont rassemblés dans une même fenêtre, et peuvent suffire pour une analyse rapide de la région.

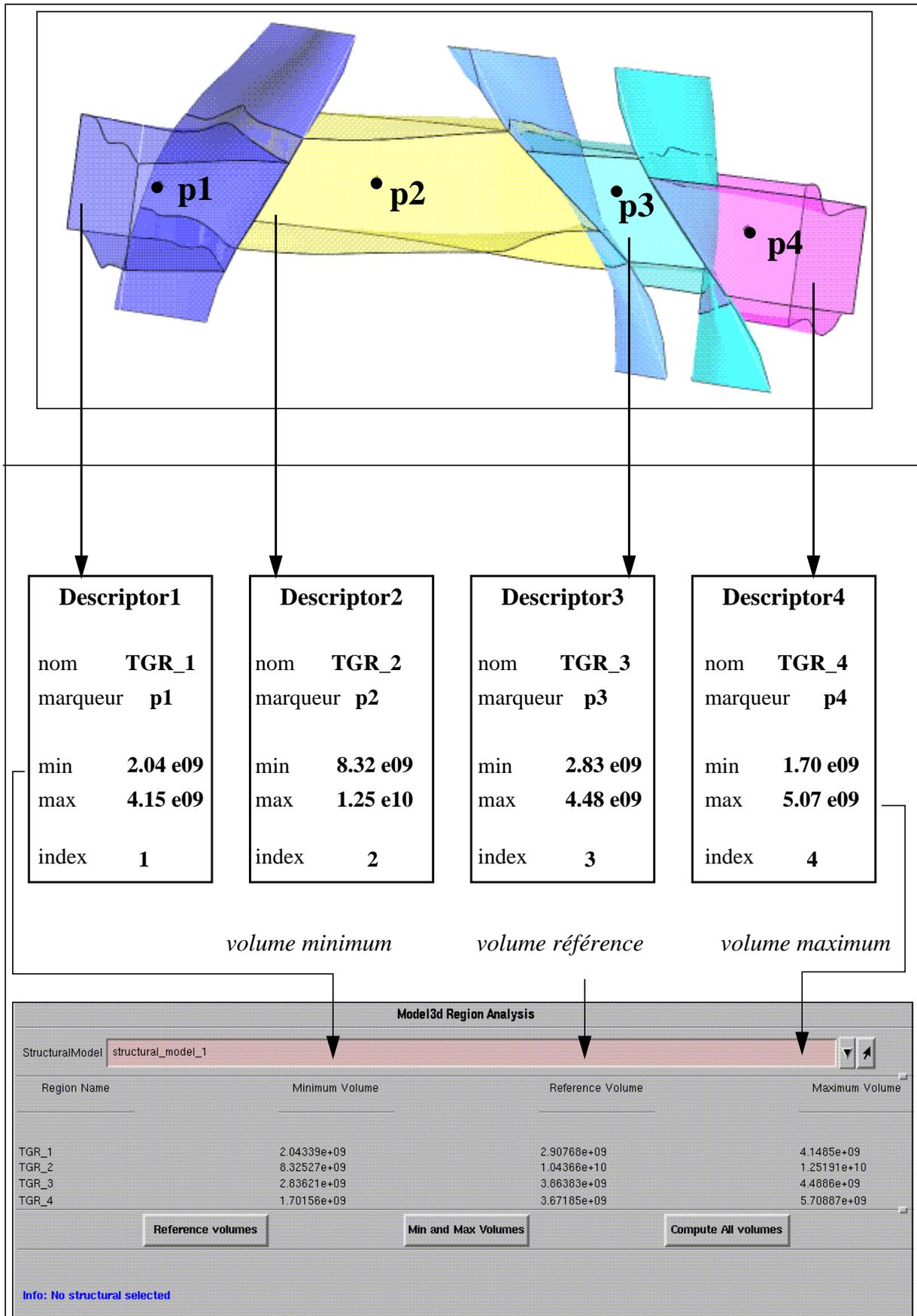


Figure 9-10 : Rassemblement des informations calculées lors de la phase 1.

## 2. Deuxième phase: boucle de simulations

### 2.1 Déroulement d'une simulation

La figure suivante montre de manière schématique le déroulement d'une simulation, qui après la phase de préparation se déroule en quatre étapes [115 - Lecour 4]:

- Le retour au **modèle initial** pour assurer l'*indépendance* des simulations
- la **simulation de la géométrie** des failles (de manière indépendante)
- la **mise à jour du modèle**
- le **stockage des informations** nécessaires pour retrouver le même modèle.

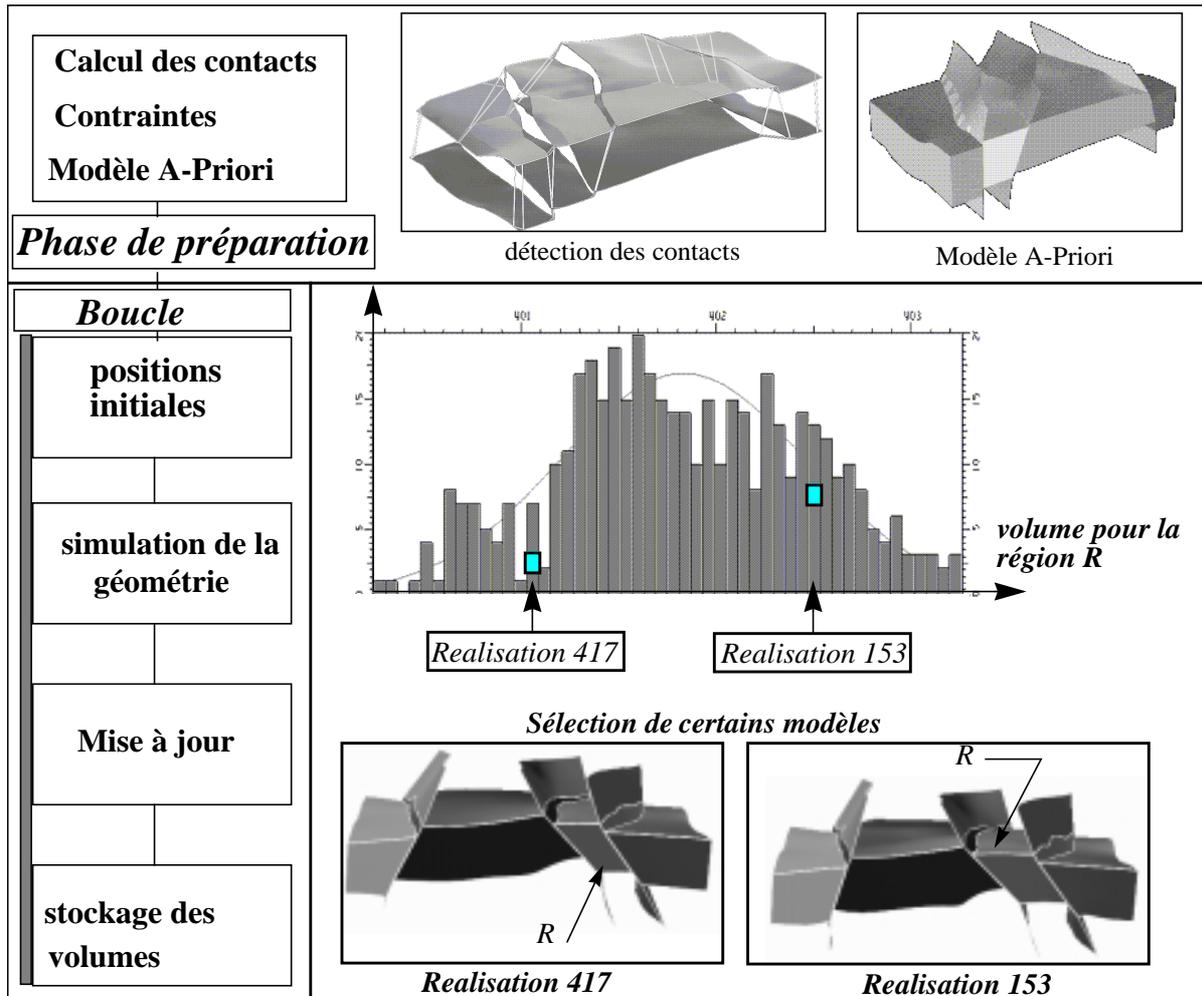


Figure 9-11 : Schéma général de l'algorithme global de simulation

Pour réaliser ces simulations, on associe un «*Simulateur de Faille*» (FAULTSIMULATOR) à chacune des failles, qui est chargé de réaliser toutes les étapes d'une simulation).

A chaque tour de boucle, ce simulateur appelle la bonne méthode de simulation pour modifier la géométrie de la faille (en fonction de ce qui a été demandé par l'utilisateur). Le calcul des translations correspondant aux nouvelles positions par rapport à la faille de référence peut être fait à partir de l'information stockée dans l'objet faille qu'il contient (une position entre [-1;1] et deux vecteurs d'incertitude pour chaque point). Nous rappelons que chaque faille (GFAULT) contenue dans le simulateur est simulée de manière indépendante actuellement.

Ensuite, il faut que la surface qui sert à construire le modèle volumique se mette à jour en fonction de ces nouvelles positions. Chaque face triangulée contenue dans le simulateur se met à jour en fonction de la faille à partir de laquelle elle a été créée.

Une fois que toutes les surfaces de faille ont été mises à jour, la dernière étape consiste à interpoler le réseau de faille, puis tous les horizons, afin de fermer le modèle surfacique et de pouvoir construire le modèle volumique (exactement comme pour construire le modèle de référence).

La fonction correspondant à la première étape (*simulation*) ne fait que calculer une nouvelle position pour chacun des points de l'objet faille, et stocker les paramètres correspondants. La fonction correspondant à la deuxième étape (*mise à jour du modèle*) peut être appelée soit directement après le calcul des paramètres de la simulation (la première fois, cela permet de calculer les volumes correspondants), soit plus tard, à partir des paramètres stockés (pour retrouver un modèle particulier).

### 2.1.1 Simulation d'une nouvelle position pour chaque point de la faille

Ces méthodes ont été décrites précisément dans le chapitre 6 (Voir «Estimation des incertitudes et méthodes de simulation», page 117) et nous ne reviendrons pas sur la manière dont elles se déroulent. Une classe (FAULTSIMULMNGR) est chargée de calculer les paramètres en fonction de la méthode sélectionnée par l'utilisateur, puis de calculer les nouvelles positions pour les points de la faille en fonction de ces paramètres.

Les choix de l'utilisateur sont stockés sous forme de booléens (qui valent 1 si ce type de simulation est sélectionné et 0 sinon). Pour chaque nouvelle simulation, une variable «code» permet de déterminer le type de simulation:

$$code = \text{décalage} + 2 \cdot \text{pendage} + 4 \cdot \text{carte}$$

En fonction de la valeur de code, la méthode correspondante est appelée (certaines méthodes semblables du point de vue algorithmique ont été rassemblées en une même fonction):

code=1: décalage latéral ( <i>simul_shift</i> )	p
code=2: changement de pendage ( <i>simul_dip</i> )	p1, p2
code=3: changement de pendage + décalage latéral ( <i>simul_dip</i> )	p1, p2
code=4: géométrie horizontale ( <i>simul_map_view</i> )	tableau p[]
code=5: horizontale + décalage ( <i>simul_map_view</i> )	bornes, tableau p[]
code=6: horizontale + pendage ( <i>simul_map_view</i> )	bornes, tableau p[]
code=7: les 3 méthodes ( <i>simul_map_view</i> )	bornes, tableau p[]

Le seul ajout pour qu'une simulation donnée puisse être retrouvée facilement, sans encombrer trop la mémoire, est qu'une classe (SIMULATIONPARAMETERS) stocke tous les paramètres qui ont servi à la réalisation courante. La classe FAULTSIMULATOR stocke donc une liste de SIMULATIONPARAMETERS (un par tour de boucle).

Si, lors de l'appel de la fonction de simulation, ces paramètres n'ont pas encore été calculés (première simulation), la fonction «*compute\_parameters()*» est appelée. En plus du numéro de la simulation courante, les valeurs caractéristiques sont ensuite stockées: p pour le décalage latéral; p1 et p2 pour la modification de pendage; graine du générateur de nombres aléatoires et bornes des intervalles restreints pour les autres méthodes.

Plus tard, lorsque l'utilisateur désire retrouver une simulation particulière, les bons paramètres sont retrouvés à l'aide du *numéro* de la simulation demandée, puis la fonction «*simulate()*» calcule les bonnes positions.

Ce stockage des paramètres est détaillé un peu plus loin dans le chapitre (Voir «Stockage des résultats» page 228.).

## 2.1.2 Mise à jour des surfaces

### a) Principe général

Une fois qu'une nouvelle position a été calculée pour chaque point de la faille, une fonction permet de mettre à jour la surface triangulée associée à l'objet faille dans le simulateur.

En effet, la gestion des contraintes étant plus facile avec les surfaces triangulées, *les deux aspects ont été séparés*: l'objet faille ne sert qu'à calculer les nouvelles positions, et la surface associée sur laquelle les contraintes sont installées modifie sa géométrie en fonction des nouvelles positions, ce qui permet ensuite de mettre à jour le modèle.

Le simulateur de faille contient donc, non seulement un objet faille, mais aussi une surface triangulée. Dans les deux paragraphes suivants, nous verrons que ces surfaces sont différentes suivant que ces opérations sont réalisées en partant d'un objet de type STRUCTURALMODEL ou d'un ensemble de surfaces (failles et horizons) reliées par des contraintes [figure 9-12].

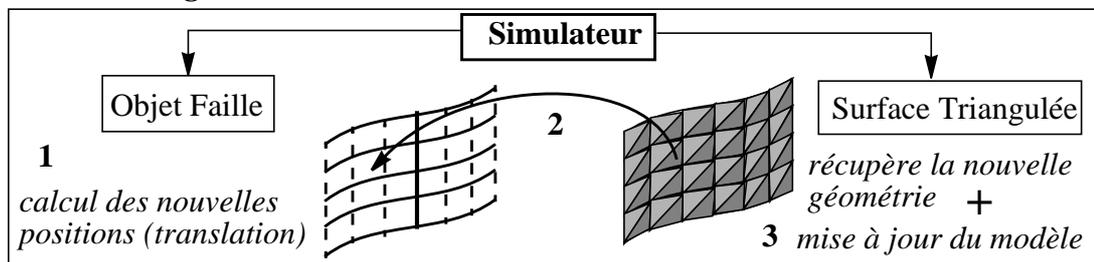


Figure 9-12 : Séparation de la simulation et de la mise à jour du modèle

De la même manière que l'objet faille (GFAULTBUNDLE) peut contenir plusieurs morceaux de faille (GFAULT), le simulateur global (BUNDLESIMULATOR) contient un simulateur par faille isolée (FAULTSIMULATOR) [figure 9-13].

La même classe de base (FAULTSIMULATOR) sert aux deux méthodes (avec ou sans objet STRUCTURALMODEL) et définit les fonctions communes. Ces fonctions concernent surtout la mise à jour de la surface associée à la faille dans le simulateur:

- *simulate()*: cette fonction utilise le gestionnaire de simulation (BUNDLESIMULMNGR) pour calculer les paramètres en fonction de ce que l'utilisateur a demandé, réalise la simulation (via le FAULTSIMULMNGR) puis stocke ces paramètres.
- *update\_fault\_points()*: cette fonction permet de calculer pour chaque point de la faille, le vecteur de translation à appliquer par rapport à la position de référence.

En fonction de la valeur «position» stockée dans l'objet faille, on calcule:

$$\text{translation} = \text{position} \cdot \text{vecteur}_d \quad \text{si} \quad \text{position} < 0$$

$$\text{translation} = \text{position} \cdot \text{vecteur}_g \quad \text{si} \quad \text{position} > 0$$

Ensuite, la position simulée (ou *position courante*) est calculée et affectée au point correspondant sur la surface triangulée:

$$\text{position\_courante} = \text{position\_référence} + \text{translation}$$

Comme certaines fonctions diffèrent en fonction des données fournies en entrée, alors que d'autres fonctions sont communes, les deux classes de base (BUNDLESIMULATOR et FAULTSIMULATOR) sont chacune dérivées en deux classes.

Lorsque les données sont stockées dans un objet STRUCTURALMODEL, on utilisera le STRUCTURALFAULTSIMULATOR (et le STRUCTURALBUNDLESIMULATOR). Lorsque les données sont constituées d'un ensemble de surfaces isolées, et qu'il faut gérer les contraintes et l'interpolation, on utilisera un CNSTRFAULTSIMULATOR (et un CNSTRFAULTSIMULATOR) [figure 9-13].

Les principales fonctions qui diffèrent (dont certaines sont décrites plus loin) sont:

- *create\_fault\_simulator\*()*: chaque type de BUNDLESIMULATOR est responsable de la création de FAULTSIMULATORS du même type que lui.
- *create\_current\_tsurf()* / *create\_current\_tface()*: comme nous le verrons dans les paragraphes suivants, la surface associée (celle qui porte les contraintes) n'est pas tout à fait la même dans les 2 méthodes. La première fonction crée une surface triangulée (TSURF) par groupe de faille (GFAULTBUNDLE), la seconde crée dans cette surface triangulée, une face (TFACE) par morceau de faille (GFAULT)
- *update\_face()*: comme les surfaces associées sont différentes, le processus de mise à jour n'est pas tout à fait le même (une partie est commune cependant: *update\_fault\_points()* [voir exemple de code 9..2]).

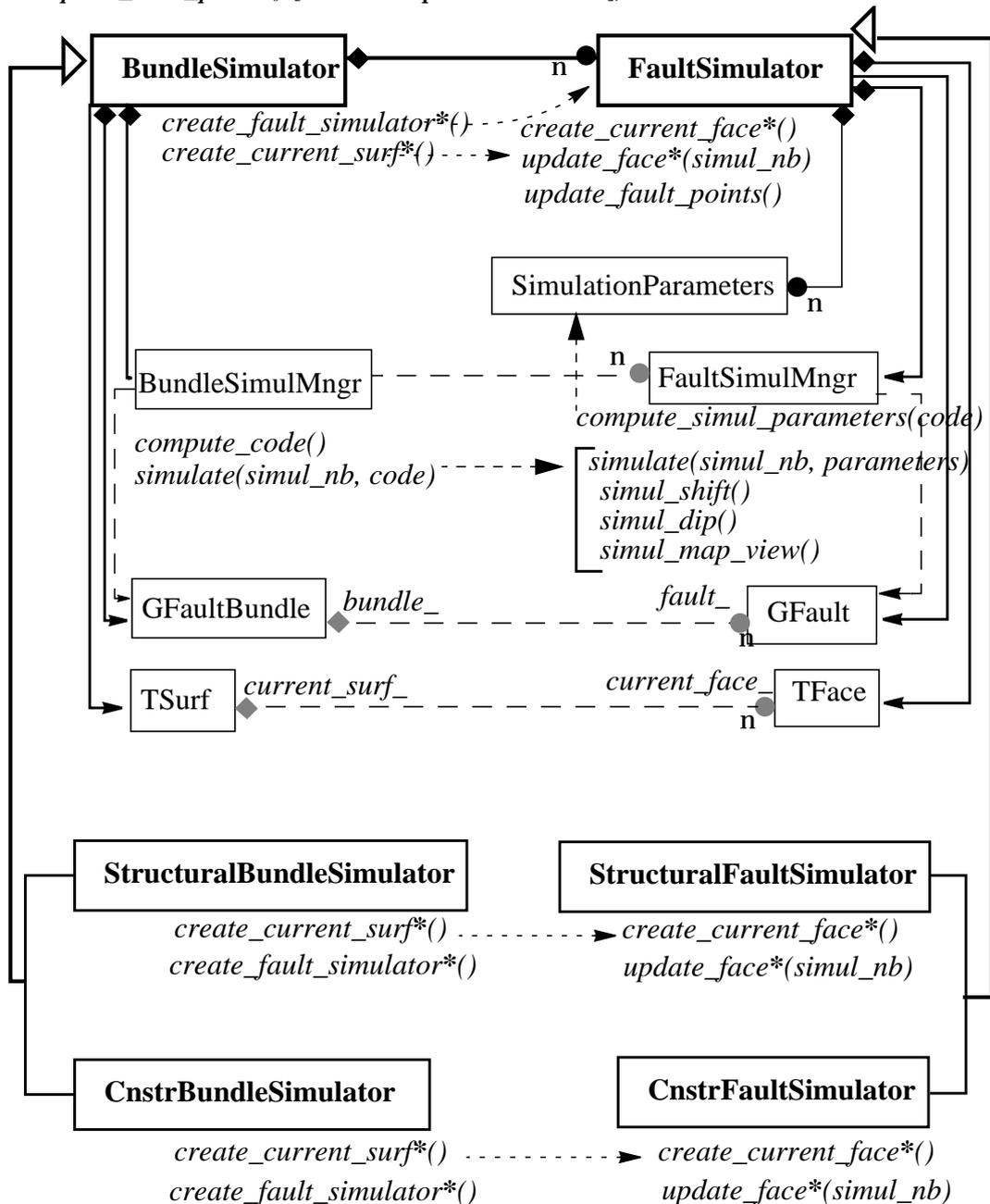


Figure 9-13 : Diagramme montrant les relations entre les simulateurs de faille et les gestionnaires de simulations. Les fonctions de base qui diffèrent entre les deux classes dérivées sont signalées par une étoile.

## b) Mise à jour sans utiliser le Structural Model

Dans ce cas, la surface qui sert à mettre le modèle à jour est une simple copie de la surface cachée dans l'objet faille. Une propriété «*indice*» permet de retrouver très facilement la correspondance entre un point de la surface triangulée, et la position courante calculée pour le point correspondant sur l'objet faille.

La fonction qui met à jour la surface triangulée dans ce cas (*update\_face\*()*) est donc très simple:

### Exemple de code 9.2. *Mise à jour de la surface*

```

1 void create_current_face*() {
2     update_fault_points() ;
3 }
4
5 void update_fault_points() {
6     // Si nb est le nombre de points de la surface triangulée
7     // et indices le nom de la propriété stockant les indices
8
9     Vector3d* translations = compute_translations() ;
10    // tableau de vecteurs
11    Point3d* references = get_reference_points() ;
12    // tableau de points (x,y,z)
13
14    Pour tous les points de la surface triangulée de i=0 à nb
15    int index = indices(i) ;
16    translation = translations[i] ;
17    reference = references[i] ;
18    new_pos = reference + translation
19    affecte_nouvelle_position(i, new_pos) ;
20 }
```

**Remarque:** plutôt que de réaliser une copie de cette surface, il aurait été plus simple (en apparence) d'utiliser directement la surface cachée. Cependant, il ne faut pas oublier que pour avoir des simulations indépendantes, il faut stocker les positions de référence afin de ne calculer que les vecteurs de translation par rapport à ces positions initiales. Comme la surface triangulée associée à l'objet faille dans le simulateur sert à construire le modèle volumique et subit des interpolations, nous avons préféré modifier une copie de la surface cachée, et nous servir de cette dernière que pour redonner les positions de référence à la surface «courante».

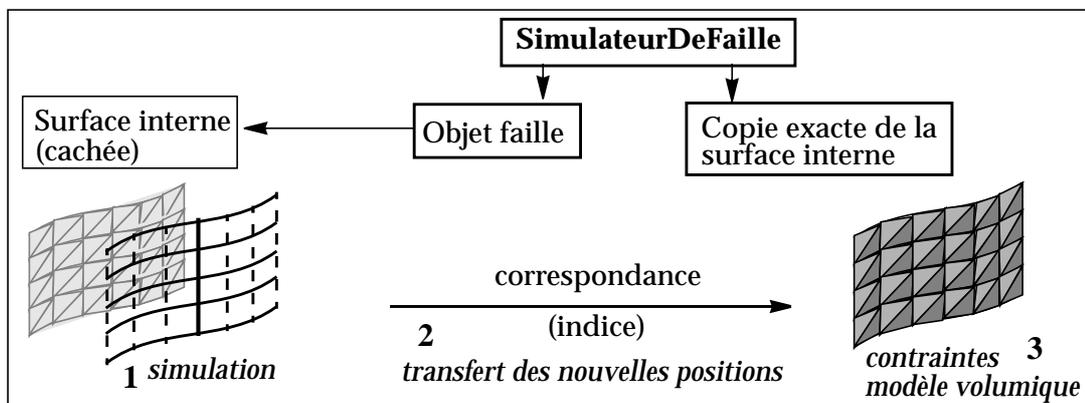


Figure 9-14 : Sans utiliser de STRUCTURALMODEL, la surface courante est une copie de la surface «cachée» dans l'objet faille.

### c) Mise à jour en utilisant le Structural Model

Dans ce cas, la surface qui sert à mettre à jour le modèle (et donc qui porte toutes les contraintes) est celle contenue dans le même FAULTMODEL que l'objet faille stocké dans le simulateur.

Contrairement à la surface cachée dans l'objet faille, elle possède un maillage où les triangles sont beaucoup plus équilatéraux, ce qui est préférable pour réaliser des interpolations de géométrie et de propriétés (rejet par exemple).

Par contre, ce maillage ne correspond pas du tout à la paramétrisation de l'objet faille, et la mise à jour de la géométrie n'est pas aussi triviale.

Afin, de simplifier la mise à jour, le simulateur de l'objet faille stocke une copie de la surface cachée de l'objet faille, étendue au niveau de chacune de ces extrémités.

En effet, lors de la construction du modèle volumique, les nœuds de la surface associée au même FAULTMODEL (B) vont directement chercher leur nouvelle position en prenant les coordonnées du point d'impact le plus proche d'eux sur la surface courante du simulateur de faille (A).

Si les bords sont exactement les mêmes, on risque de ne pas trouver de point d'impact pour les points du bord, du fait des imprécisions numériques. C'est pourquoi, en travaillant avec une surface étendue, on s'affranchit de ce risque [figure 9-15].

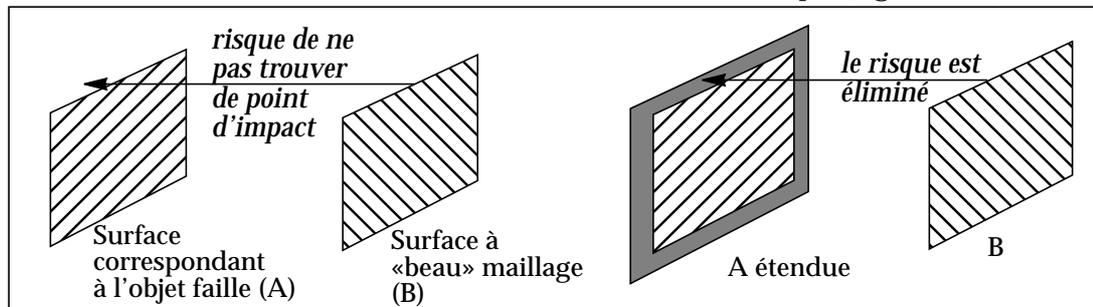


Figure 9-15 : Extension des bords afin d'éviter les problèmes numériques.

Par contre, ces points du bord n'ayant aucune correspondance avec l'objet faille (qui stocke les nouvelles positions), on est obligé de leur appliquer un traitement spécial. Lors de la construction de la surface courante (A), tous les points qui ne sont pas sur les bords sont mis en «points de contrôle». De plus, la propriété «indice» est initialisée à -1 pour tous les points du bord, et à l'indice du point de l'objet faille pour les autres points.

La même fonction que celle détaillée dans l'exemple de code 9..2) est utilisée (`update_fault_points()`). Un test permet de ne pas traiter les points du bord (indice égal à -1) durant cette phase.

Il ne reste plus, ensuite qu'à réaliser une interpolation sur cette surface. Les points mis en contrôle ne bougent pas, et à condition d'avoir mis la contrainte qui évite le rétrécissement des bords, ceux-ci se déplacent en suivant la tendance donnée par les points de contrôle [figure 9-16].

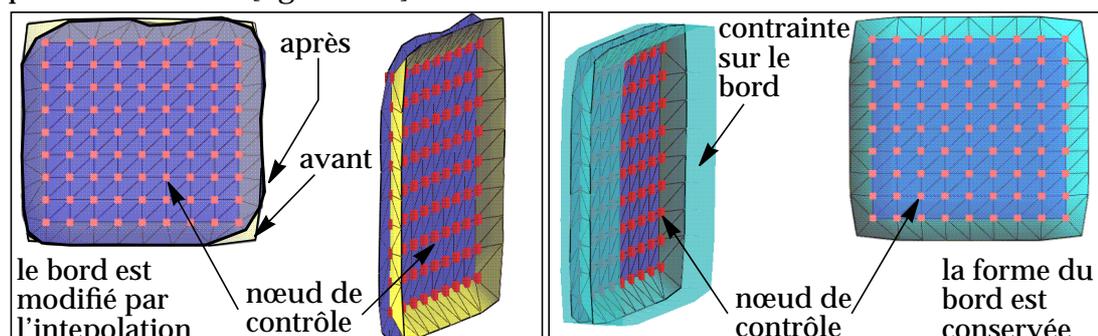


Figure 9-16 : Interpolation afin de mettre à jour les points du bord.

La fonction de mise à jour se sépare donc en deux parties:

- *update\_fault\_points()*: modifie les points de la surface triangulée qui sont en correspondance avec l'objet faille (indice différent de -1)
- *update\_border\_points()*: réalise l'interpolation pour que les points du bord suivent la tendance donnée par les points précédents.

Le code correspondant à la mise à jour des surfaces triangulées par le STRUCTURAL-FAULTSIMULATOR, peut donc sembler plus complexe que celui correspondant à la mise à jour réalisée par le CNSTRFAULTSIMULATOR, notamment en ce qui concerne la construction de la face courante (celle qui est en correspondance avec l'objet faille).

Il en est de même pour la représentation des relations entre les différents objets intervenant dans la simulation [figure 9-17].

L'exemple de code suivant, montre de manière très simplifiée comment se réalise la construction de la surface courante du simulateur de faille. Les lignes en gras correspondent à ce que l'on doit rajouter pour le STRUCTURALFAULTSIMULATOR (et sont beaucoup moins nombreuses que dans le code réel).

*Exemple de code 9..3. Construction de la surface interne*

```

1 void create_current_face*() {
2     // copie de la surface interne cachée dans l'objet faille
3     TFace* face_interne = objet_faille.face_cachée() ;
4     TFace* current_face_ = copie(face_interne) ;
5
6     // extension du bord de la face
7     double extension = calcule_extension_distance() ;
8     étendre_bordure(extension) ;
9
10    // Marquage des noeuds de contrôle
11    Pour tous les noeuds de la face de n=0 à nb_noeuds
12    si le noeud est sur le bord
13    indice_du_noeud = -1 ;
14    sinon
15    noeud = noeud_de_contrôle() ;
16    fin_si
17
18    // ajout de la contrainte pour maintenir le bord
19    Pour tous les bords de b=0 à nb_bords
20    installation_de_la_contrainte() ;
21
22    // affectation des bons indices
23    Pour tous les atomes du backbone de u=0 à nb_atom
24    Pour tous les points de la génératrice de v=0 à nb_points
25    Pour tous les points de la face de p=0 à nb_p
26    Si point_face = point_faille
27    indice_point_face = u*nb_points + v ;
28    fin_si
29 }
```

Cependant, autant cette phase de mise à jour des surfaces triangulées semble en défaveur de l'utilisation d'un objet STRUCTURALMODEL, autant la reconstruction du modèle justifie son utilisation, et les quelques efforts supplémentaires lors de la construction de la surface courante. C'est pourquoi, faute de temps, pour cette dernière partie, tout n'a pas été entièrement réalisé pour le CNSTRFAULTSIMULATOR.

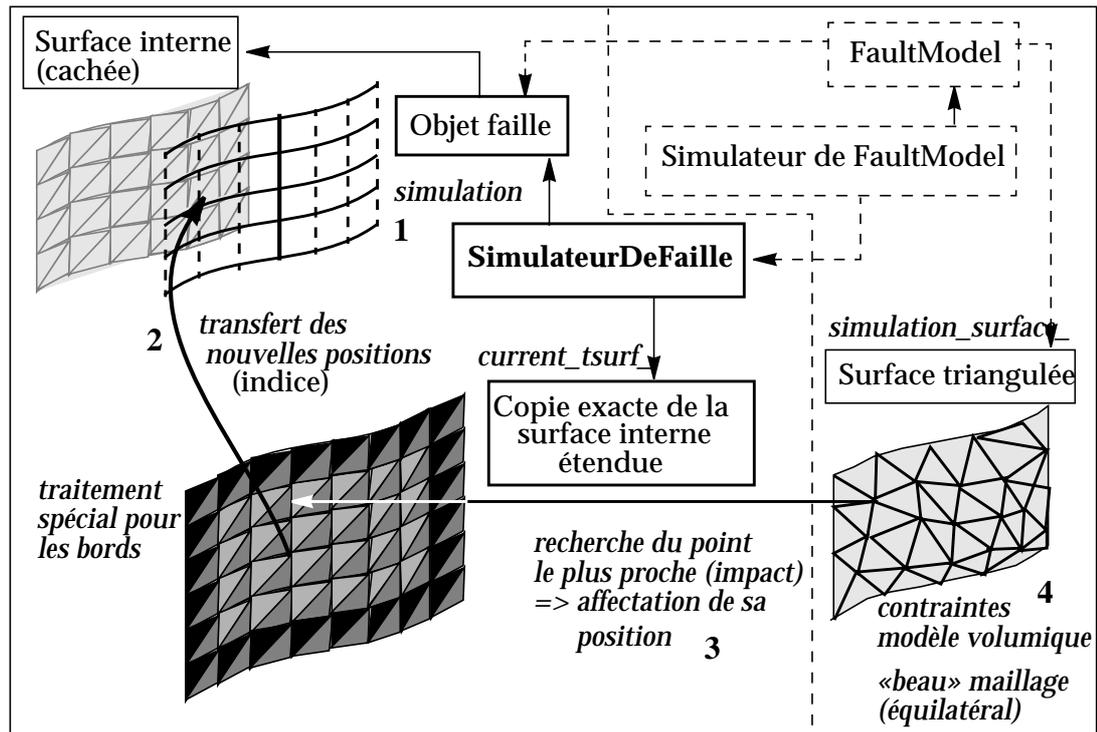


Figure 9-17 : Avec l'utilisation d'un STRUCTURALMODEL, la surface courante est une copie de la surface «cachée» dans l'objet faille étendue, mais elle ne sert que d'intermédiaire.

### 2.1.3 Reconstruction du modèle volumique

De même que la mise à jour des surfaces, la reconstruction du modèle volumique est très différente en fonction des données utilisées au départ. Si beaucoup de choses sont réalisées globalement et de manière automatique à l'aide du STRUCTURALMODEL, elles doivent être faites par le simulateur de faille lorsque l'on part de surfaces isolées.

Le principe global reste cependant très semblable à la construction du modèle de référence:

- **interpolation de toutes les surfaces de faille**: actuellement, ces interpolations se réalisent dans l'ordre où les failles sont stockées, ce qui n'est pas géologique. Le stockage structuré des contacts dans le StructuralModel permet d'imaginer une meilleure gestion de ces interpolations plus tard (voir dernier chapitre).
- **interpolation des surfaces d'horizon**: pour cette étape, il n'y a pas vraiment de différence entre les deux méthodes.
- **reconstruction du modèle volumique**: avec découpage si l'on n'utilise pas de STRUCTURALMODEL et en passant par des surfaces de failles pré-découpées dans le cas contraire.

Même si le début de ce travail a été réalisé sans utiliser le STRUCTURALMODEL, la suite de la thèse a été beaucoup orientée pour correspondre à cette structure de donnée, qui stocke *proprement* tous les contacts. Ce choix correspond aussi aux difficultés rencontrées il y a trois ans, lors de la construction d'un modèle volumique sans pré-découper les surfaces (ces difficultés semblent maintenant éliminées).

La suite de ce paragraphe concerne donc principalement le STRUCTURALFAULTSIMULATOR.

### a) Avec l'objet StructuralModel

Les classes qui calculent les différentes simulations suivent la même organisation que celle du STRUCTURALMODEL [figure 9-18]. Le diagramme correspondant peut sembler complexe, mais dans la plupart des classes «globales» les fonctions ne font qu'appeler la même fonction pour toutes les sous-classes qu'elles contiennent. Le code de ces classes globales est donc très simple.

- Un **FAULTNETWORKSIMULATOR** est chargé des opérations les plus globales:
  - initialisation de toutes les surfaces à la géométrie de référence (*init()*)
  - lancement de la simulation numéro «nb» (*simulate(nb)*)
  - calcul du modèle volumique correspondant (*update\_3D\_model(nb)*)

Il contient une liste de **FAULTMODELSIMULATOR** (un par **FAULTMODEL**) et leur délègue le calcul des positions courantes pour les points de l'objet faille (la fonction «*simulate()*» appelle la fonction «*simulate()*» de chacun des **FAULTMODELS** qu'il contient.

- Chaque **FAULTMODELSIMULATOR** ne s'occupe que d'un **FAULTMODEL** à la fois. Il permet de retrouver très rapidement les correspondances entre la représentation «Objet-Faille» et la représentation «SurfaceTriangulée» correspondant à la même faille. De plus, il crée et stocke le **STRUCTURALBUNDLESIMULATOR** correspondant à la faille dont il s'occupe.

Lorsque sa fonction «*simulate()*» est appelée, elle ne fait qu'appeler la fonction «*simulate()*» su **STRUCTURALBUNDLESIMULATOR** qu'il contient.

- Chaque **STRUCTURALBUNDLESIMULATOR** s'occupe de simuler le **GFAULTBUNDLE** du **FAULTMODEL** qui le contient.

Cette classe a déjà été présentée. Elle contient un **BUNDLESIMULMNGR**, chargé de calculer le type de simulation demandé par l'utilisateur. De plus, elle contient un **STRUCTURALFAULTSIMULATOR**, qui lui calcule les paramètres de simulation correspondants, réalise la simulation demandée, et stocke les bons paramètres.

- Enfin, en bout de chaîne, le **STRUCTURALFAULTSIMULATOR** réalise la simulation. Comme on l'a vu dans le paragraphe précédent, cette simulation est calculée sur l'objet faille, et ensuite la surface associée (*surface courante*) est mise à jour.

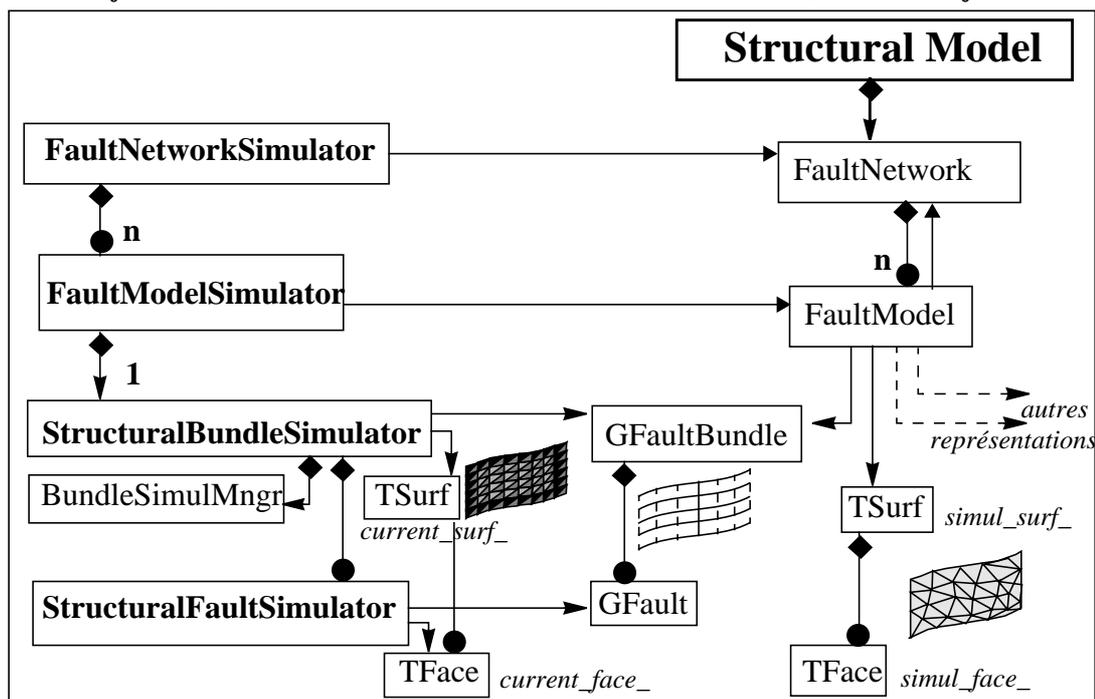


Figure 9-18 : L'architecture des simulateurs suit celle du StructuralModel.

Si l'on revient à l'algorithme général, on peut enfin décrire les trois grandes étapes d'une simulation qui sont réalisées au plus haut niveau (dans le FAULTNETWORKSIMULATOR) [voir exemple de code 9..4]:

1. **Initialisation:** fonction *init()*

Cette fonction remet tous les points des failles et des horizons à leur position initiale. Cette position de référence est stockée sous forme de propriété lors de la construction du modèle de référence.

2. **Simulation des positions:** fonction *simulate()*

Comme cela a été dit plus haut, cette fonction ne fait que provoquer l'appel en cascade des fonctions «*simulate()*» des différents simulateurs, jusqu'à arriver au FAULTSIMULATOR-MNGR, qui réalise vraiment la simulation, puis met à jour la surface triangulée «*courante*». A ce stade, seul l'objet faille stocke les positions simulées.

3. **Calcul du modèle volumique:** fonction *update\_3D\_model()*

Cette fonction se déroule en trois phases. La première consiste à mettre à jour la représentation surface de chaque FAULTMODEL. C'est le FAULTMODELSIMULATOR qui réalise le travail. Cela permet d'associer à chaque point de la représentation «surface triangulée» la position du point d'impact le plus proche sur la surface courante du STRUCTURALBUNDLE-SIMULATOR. Ensuite, il suffit d'interpoler toutes les surfaces de faille, tous les horizons, puis de reconstruire le modèle volumique, exactement de la même manière que pour le modèle de référence (en créant des surfaces de faille pré-découpées).

*Exemple de code 9..4. Simulation d'un modèle entier (simplifié)*

```

1   init() {
2       init_horizons() ;
3       init_fault_surfaces() ;
4   }
5
6   simulate(paramètres de l'utilisateur) {
7       // calcul des positions pour les points de l'objet faille
8       Pour tous les FaultModelSimulator
9           fm_simulator -> simulate(paramètres de l'utilisateur) ;
10          // appelle lui-même la fonction simulate() du
11          // BundleSimulator
12  }
13
14  update_3D_model(int simul_nb) {
15      // calcul des positions pour la surface courante
16      // (appelle des fonctions du FaultSimulator)
17      update_simulation_surfaces(simul_nb) ;
18
19      // mise à jour des surfaces du StructuralModel
20      update_faultnetwork() ;
21      update_horizon_set();
22
23      // creation du nouveau modèle (comme pour modèle_initial)
24      Model3d* model = new Model3d ;
25      structural_-> create_model3d(model) ;
26
27      // Vérification des noms des régions
28      Model3dVolumeAnalyser* analyser = structural_->analyser() ;
29      analyser-> execute(model) ;
30  }

```

## b) Stockage des résultats

A plusieurs reprises, il a été mentionné que pour chaque simulations, un certain nombre de paramètres sont stockés: les paramètres permettant de recalculer la même simulation d'une part, et les volumes des régions correspondant au modèle volumique courant.

### 1. Stockage des paramètres de la simulation

Le premier type de stockage est spécifique à un FAULTMODEL, voir à une GFAULT de ce FAULTMODEL. En effet, actuellement, chaque GFAULTBUNDLE ne contient qu'une seule GFAULT, et ces morceaux de failles sont simulés de manière complètement indépendante.

Comme cela est détaillé dans le dernier chapitre (voir «Objectifs restant en structurale: organisation des réseaux de failles (hiérarchie et corrélation en 3D», page 241), on peut imaginer de regrouper certains morceaux de failles dans un même GFAULTBUNDLE, afin de tenir compte de la hiérarchie.

Ceci permettrait de calculer des paramètres de simulation communs à toutes les failles d'un tel groupement de faille, et de ne faire différer les simulations que d'un  $\epsilon$  d'une faille à l'autre (si cela est souhaité).

C'est pourquoi le calcul des paramètres globaux de la simulation est réalisé par le BUNDLESIMULATOR, et que les paramètres ayant réellement servi à la simulation sont stockés dans le FAULTSIMULATOR.

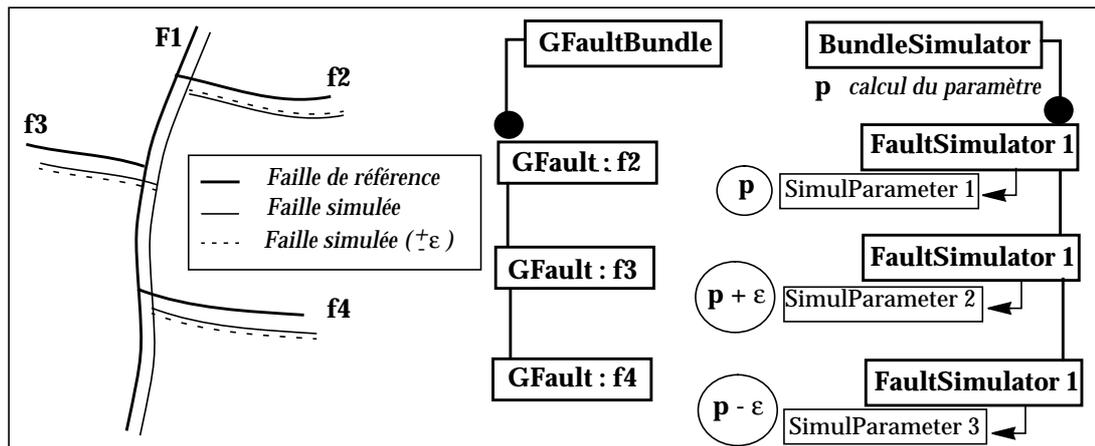


Figure 9-19 : Introduction de la notion de hiérarchie dans les simulations.

La classe qui stocke les paramètres (SimulParameter) stocke le numéro de la simulation, ainsi que le code correspondant au type de simulation. Cette classe est ensuite dérivée en trois classes qui stockent chacune les paramètres correspondant à un type de simulation particulier. [figure 9-20]

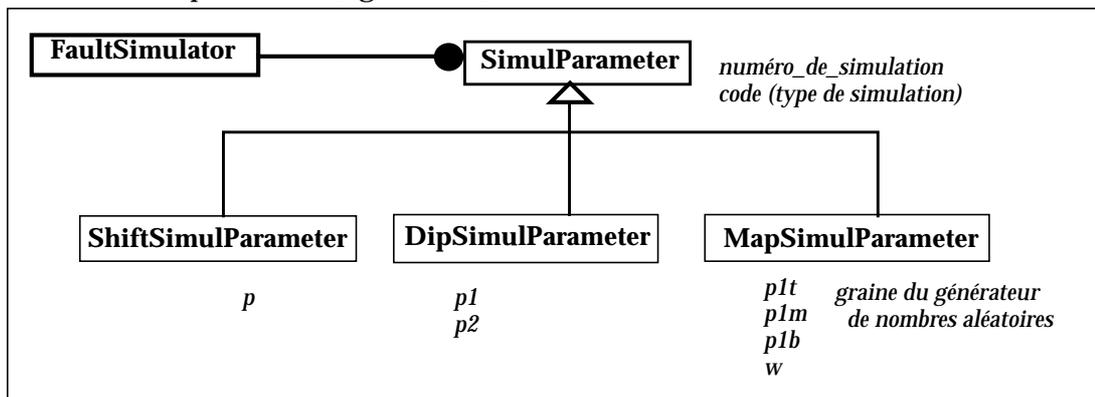


Figure 9-20 : Stockage des paramètres après chaque simulation.

### 2. Stockage des volumes pour le modèle courant

Au contraire des précédents, ces informations concerne tout le modèle structural et ne doivent être stockées qu'une seule fois.

C'est pourquoi, une fois que le modèle est construit, et que le nom des régions correspond bien aux région du modèle de référence, tous les volumes sont calculés puis stockés dans une structure de type SIMULSTORAGE, stockée elle-même dans le FAULTNETWORKSIMULATOR.

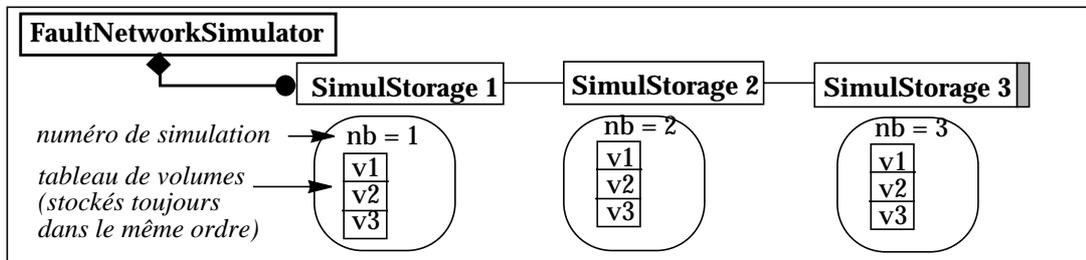


Figure 9-21 : Stockage des volumes du modèle courant (après 3 simulations).

### 3. Stockage des éléments descriptifs de la région

Enfin on peut rappeler qu'il existe une troisième forme de stockage: les objets du type REGIONDESCRIPTOR qui permettent de renommer les régions du modèle courant de la même manière que le modèle de référence.

Le diagramme ci-dessous résume toutes les classes intervenant dans la simulation:

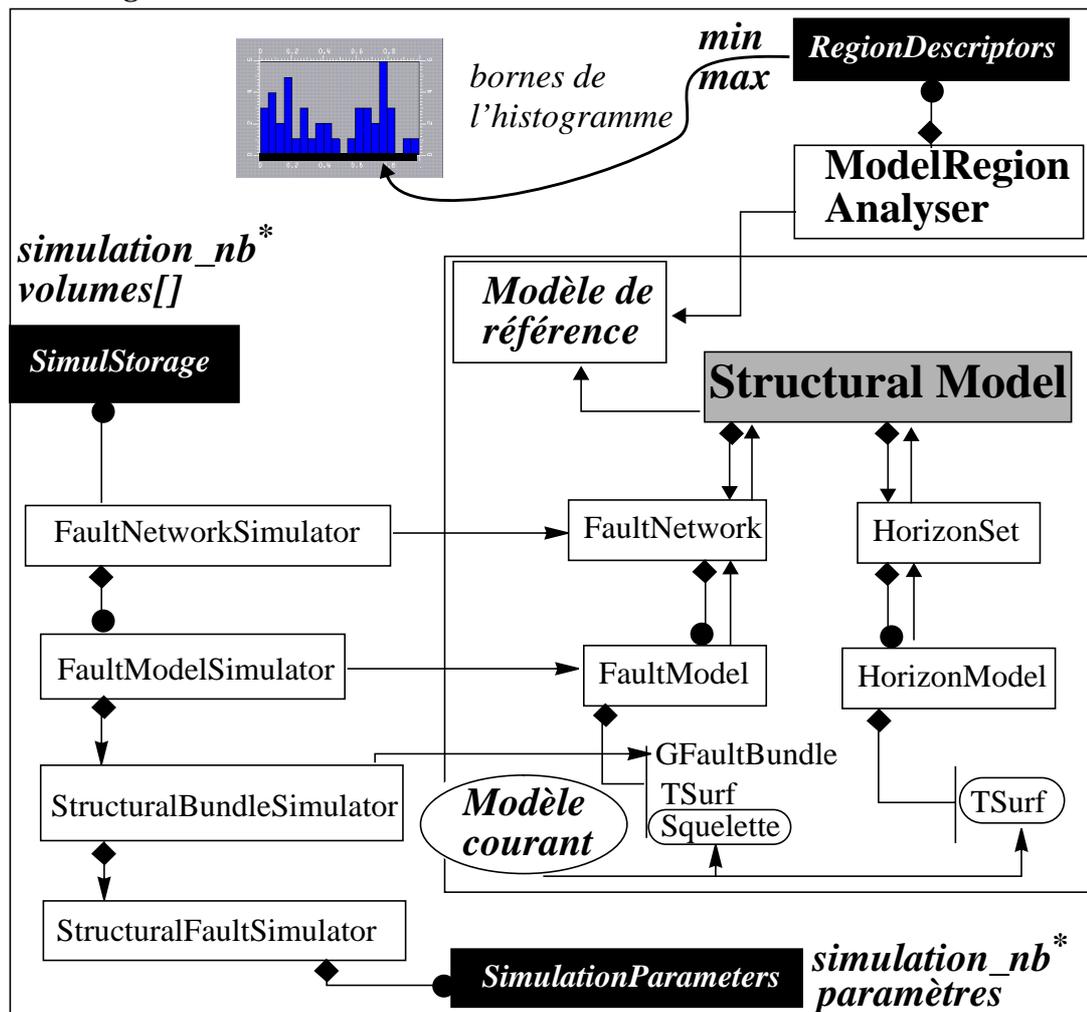


Figure 9-22 : Diagramme récapitulatif (classes de stockage sur fond noir)

### 3. Conclusion sur ces structures

Toutes les classes présentées dans ce chapitre permettent de réaliser de nombreuses simulations sur des modèles qui peuvent, a priori, être complexes.

Ces classes sont nombreuses car elles sont calquées sur la structure de l'objet `STRUCTURALMODEL`, mais chacune de ces classes est spécialisée et ne contient que très peu de fonctions à la fois. La plupart de ces fonctions, dans les classes de haut niveau consistent uniquement à appeler les fonctions correspondantes dans les classes qu'elles contiennent (comme la fonction `simulate()`).

Le principal but de ce travail est atteint: le **processus complet** de simulation est mis en place, des interprétations des différentes surfaces du modèle (failles et horizons) et des estimations d'incertitude autour des failles, à l'estimation des variations de volume sur le modèle structural.

L'algorithme global est écrit de façon *modulaire*, ce qui permettra d'améliorer certaines des étapes lorsque de meilleures solutions seront trouvées, sans provoquer de changement radical (pour les méthodes de simulation notamment). Les solutions simples disponibles actuellement permettent cependant de réaliser les premières études.

Les structures mises en place permettent en outre de stocker les paramètres de chacune des simulations, afin de pouvoir retrouver une simulation donnée par la suite, tout en stockant le minimum de paramètres nécessaires.

Afin que le travail soit complet, il faut aussi permettre à l'utilisateur de sélectionner les méthodes de simulation et de visualiser les résultats, sans avoir à connaître toute l'architecture. Le chapitre suivant donne un exemple d'interface permettant à l'utilisateur de suivre les résultats au fur et à mesure.

---

---

# Chapitre 10 Lancement des simulations et visualisation des résultats

---

---

Le chapitre précédent était axé sur la description des structures et algorithmes mis en place. Ce chapitre se propose, lui, de montrer sur quelques exemples, un aspect plus axé sur l'utilisation de ces structures.

La première partie de ce chapitre décrit l'interface actuelle, et montre sur deux exemples relativement simples comment se déroule un cycle de simulation. En ce qui concerne l'évolution des volumes, on peut présenter les résultats d'une manière synthétique à l'aide d'une fenêtre contenant un histogramme par région, et se mettant à jour après chaque simulation.

Dans une deuxième partie, ce chapitre décrit comment la hiérarchie du réseau de failles pourrait être prise en compte, sans avoir à réaliser de gros changements dans les structures et algorithmes actuels. Cet aspect est fondamental pour générer des réalisations réalistes, mais certaines structures de données manquent encore pour inclure la notion de hiérarchie dans les simulations. Cependant, si cette hiérarchie n'est pas encore utilisée, les structures sont décrites de manière à pouvoir en tenir compte le plus vite possible.

Enfin, la dernière partie de ce chapitre décrit comment le processus actuel peut s'insérer dans le logiciel JACTA, pour lequel ce travail a été réalisé principalement.

## 1. Description de l'interface

Tout le processus de simulation peut être visualisé à l'aide d'une unique fenêtre, de la sélection des paramètres, au suivi des réalisations. Cette fenêtre est moins sophistiquée que celles proposées dans le logiciel JACTA, mais ne sert que temporairement, en attendant l'intégration du travail dans ce logiciel.

Cette fenêtre peut se décomposer en quatre parties [figure 10-1]:

- **La sélection des paramètres:** (1), (2), (3). Cette partie de l'interface est encore provisoire. Elle pourra être modifiée lorsque le travail de Cécile Audinet sera terminé [86 - Audinet].
- **Les fonctions de simulation:** histogrammes, (4), (5), (6). A l'appel de cette commande, cette partie de fenêtre construit un histogramme par région du modèle. Chaque histogramme est vide au départ, mais les bornes correspondent aux volumes minimum (minoré) et maximum (majoré) stockés dans les descripteurs de région. La couleur de l'histogramme correspond à celle de la région.
- **La récupération de certains modèles** pour poursuivre l'étude avec des modèles caractéristiques (ou sélectionnés au hasard) (7), (8). Cette partie n'est pas achevée.
- La zone donnant les **informations** au fur et à mesure (9) (numéro de simulation courant par exemple).

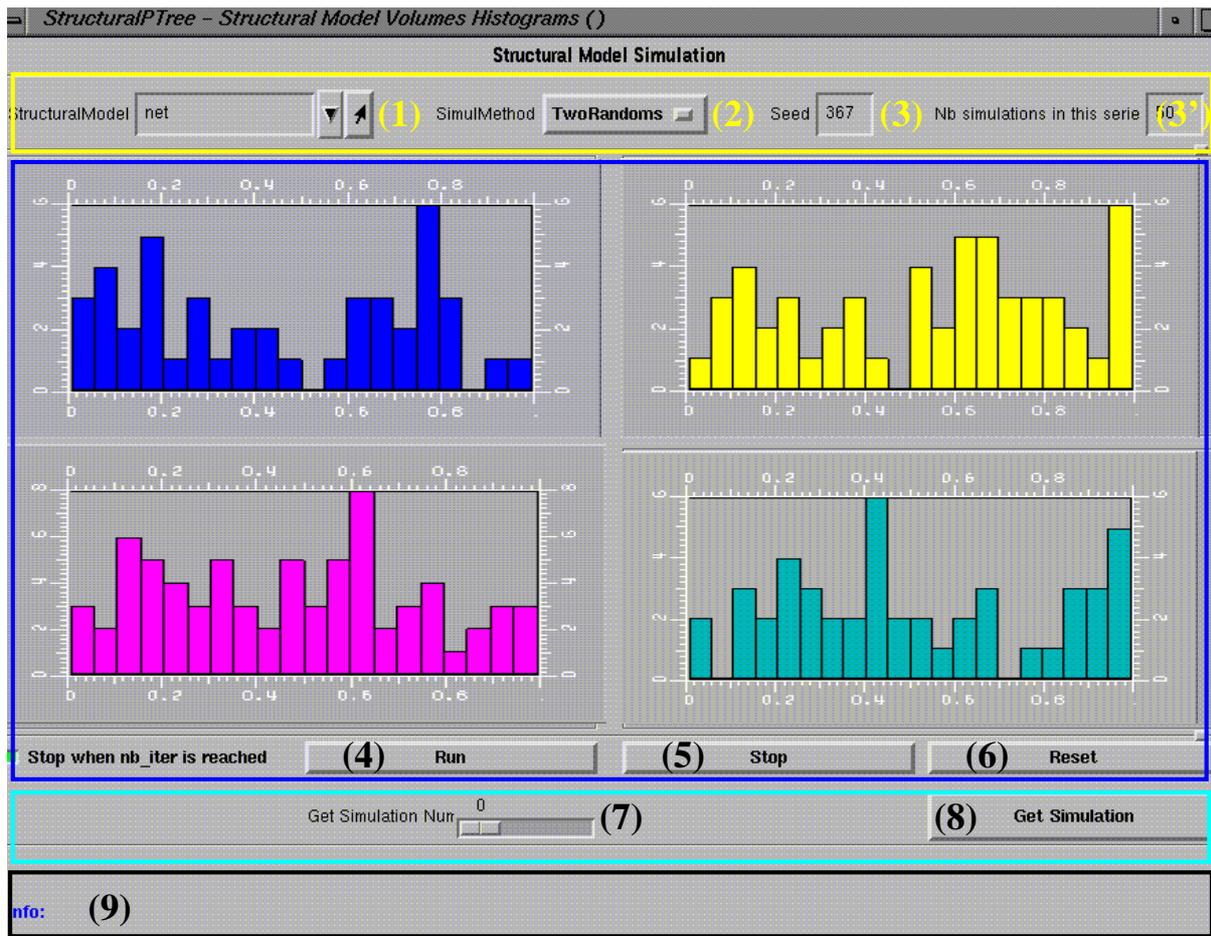


Figure 10-1 : Fenêtre permettant la visualisation des résultats.

## 1.1 Choix des paramètres

### a) Fenêtre actuelle

La fenêtre disponible actuellement ne permet pas de choisir toutes les possibilités. Seules les trois méthodes principales sont disponibles (décalage latéral, modification du pendage et modification en carte).

Elle permet de sélectionner l'objet de type STRUCTURALMODEL sur lequel on veut réaliser des simulations (1). La deuxième partie de cette boîte permet de sélectionner une méthode de simulation parmi les trois méthodes citées ci-dessus (2). Cette méthode sera la même pour toutes les failles du réseau. Enfin, l'utilisateur peut choisir la graine du nombre aléatoire (3) et le nombre de simulations (3').

### b) Amélioration possible

Lorsque le gestionnaire de méthode de simulation sera au point, il sera possible de donner plus de flexibilité à l'utilisateur. Au lieu de sélectionner une méthode globale pour toutes les failles, on peut envisager que le mode de simulation puisse être différent pour les différentes failles du modèle. En effet, le type d'incertitude n'est pas forcément le même sur toutes les failles d'un modèle.

Le bouton de sélection du type de simulation pourrait donc ouvrir une fenêtre où l'utilisateur pourrait choisir le type de simulation pour chacune des failles [figure 10-2]

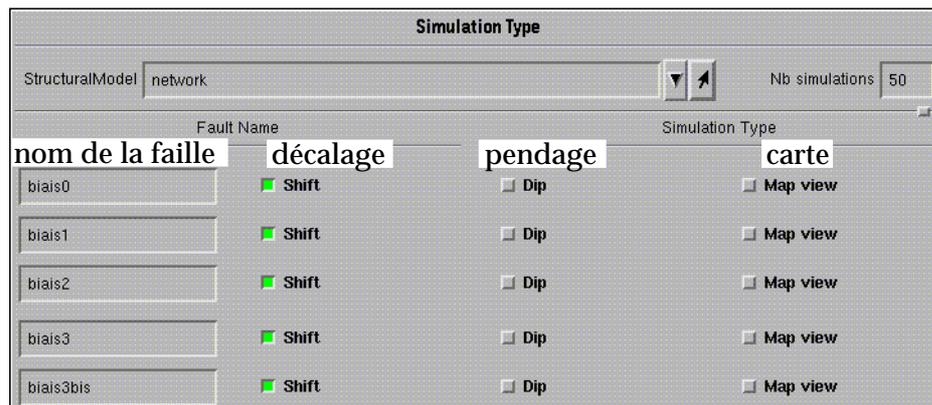


Figure 10-2 : Sélection du type de simulation en fonction de la faille.

## 1.2 Les boutons “Run”, “Stop” et “Reset”

Ces trois boutons concernent la *simulation proprement dite*.

Le premier bouton (*run* 4) lance un cycle de simulations. Après chaque tour de boucle, les volumes de chaque région sont calculés et les histogrammes sont mis à jour. Lorsque le nombre de simulations demandé en (3') est atteint le processus s'arrête. Mais les structures ne sont pas détruites, et l'utilisateur peut relancer un cycle de simulations (par exemple si les premières simulations ne sont pas suffisantes pour prendre une décision).

Le deuxième bouton (*stop* 5) permet d'arrêter un cycle de simulations en cours (afin de changer des paramètres par exemple). Le troisième bouton (*reset* 6), permet de tout effacer, afin de recommencer une simulation avec d'autres paramètres par exemple. Ceci provoque le nettoyage de toutes les structures (listes de paramètres et de volumes correspondants).

**Remarque:** il serait judicieux de rajouter un bouton «*save*» qui permettrait de stocker tous les paramètres calculés, afin de reprendre une étude arrêtée trop tôt par exemple.

## 1.3 Sélection d'un certain nombre de modèles

Le but final de ce travail est de permettre à l'utilisateur de sélectionner une région qui l'intéresse, un intervalle de l'histogramme qui correspond à un volume spécifique de cette région (ou pris au hasard, en fonction de ce que l'utilisateur souhaite), et de lui redonner l'un des modèles dont le volume de la région considérée se situe entre les deux bornes de l'intervalle choisi.

Pour cela, il suffit de parcourir tous la liste des objets stockant les volumes, de garder tous les numéros de simulation qui donnent un volume dans l'intervalle sélectionné pour la région choisie. Dans un deuxième temps, il faut retrouver les paramètres de simulation correspondant au même numéro, et recalculer le modèle volumique.

Cette commande ultime n'a pas été entièrement testée, faute de temps, mais tous les éléments nécessaires pour la réaliser sont stockés (sélection des numéros de simulation «candidats», choix d'un numéro parmi les précédents). L'étape de reconstruction du modèle est exactement la même que lors de la première construction, qui sert à calculer les volumes.

## 2. Résultats obtenus sur des modèles simples

Dans tout ce qui a été présenté jusqu'à présent, le *facteur limitant* est du à l'étape de **construction du modèle volumique**. Au démarrage de ce travail, les instabilités numériques ne permettaient souvent pas de calculer le modèle initial, éliminant toute possibilité de tester la suite du processus. Avec le pré-découpage des surfaces (grâce, entre autres, à l'objet STRUCTURALMODEL), la construction du modèle volumique se déroule beaucoup mieux, à condition de détecter les contacts correctement au départ. Mais cet objet étant très récent, il ne permet pas encore de résoudre tous les cas de figure, et ne fonctionne pas encore de manière parfaite.

C'est pourquoi jusqu'à présent, les tests n'ont été effectués que sur de *petits modèles* et suffisamment *simples* pour s'affranchir des problèmes liés à la construction du modèle (peu de contacts entre les failles, «*contacts en T*» uniquement) [116 - Lecour 5].

### 2.1 Exemple du modèle de Soultz

Ce modèle très simple ne contient que deux horizons et trois failles plus ou moins parallèles (donc sans contact), définissant quatre régions.

La figure suivante montre un exemple de simulation réalisé sur ce modèle [figure 10-4]. Malgré sa simplicité, **une boucle de simulation dure 2 minutes** sur une SGI O2 avec un processeur R5000, dont 30 secondes pour la partie simulation, et 1 minute 30 pour la construction des surfaces pré-découpées, et du modèle volumique.

La méthode est donc encore *un peu lente* (en tous cas, trop pour pouvoir la tester sur de plus gros modèles).

### 2.2 Exemple d'un modèle synthétique plus complexe

Si l'exemple utilisé pour expliquer le principe de la simulation est très simple, d'autres tests ont été effectués sur un modèle synthétique comportant 3 horizons, 5 failles, 12 régions et 55 contacts en T (en comptant ceux avec la boîte englobante).

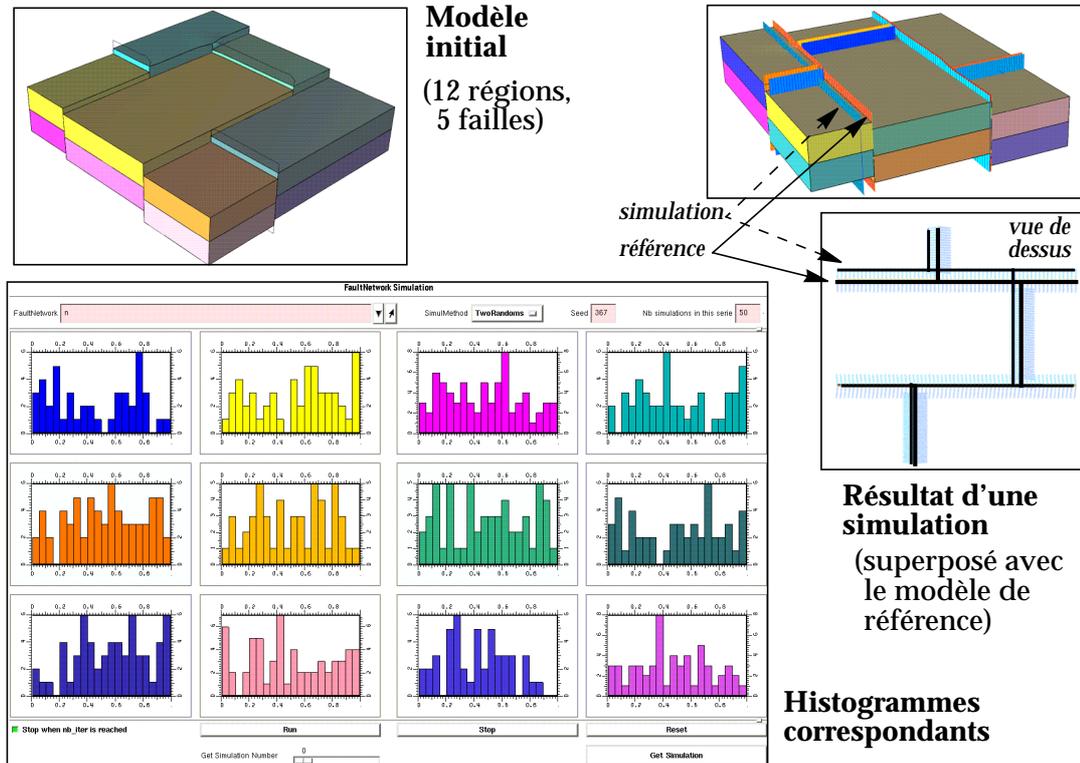


Figure 10-3 : Exemple de simulations sur un modèle plus conséquent.

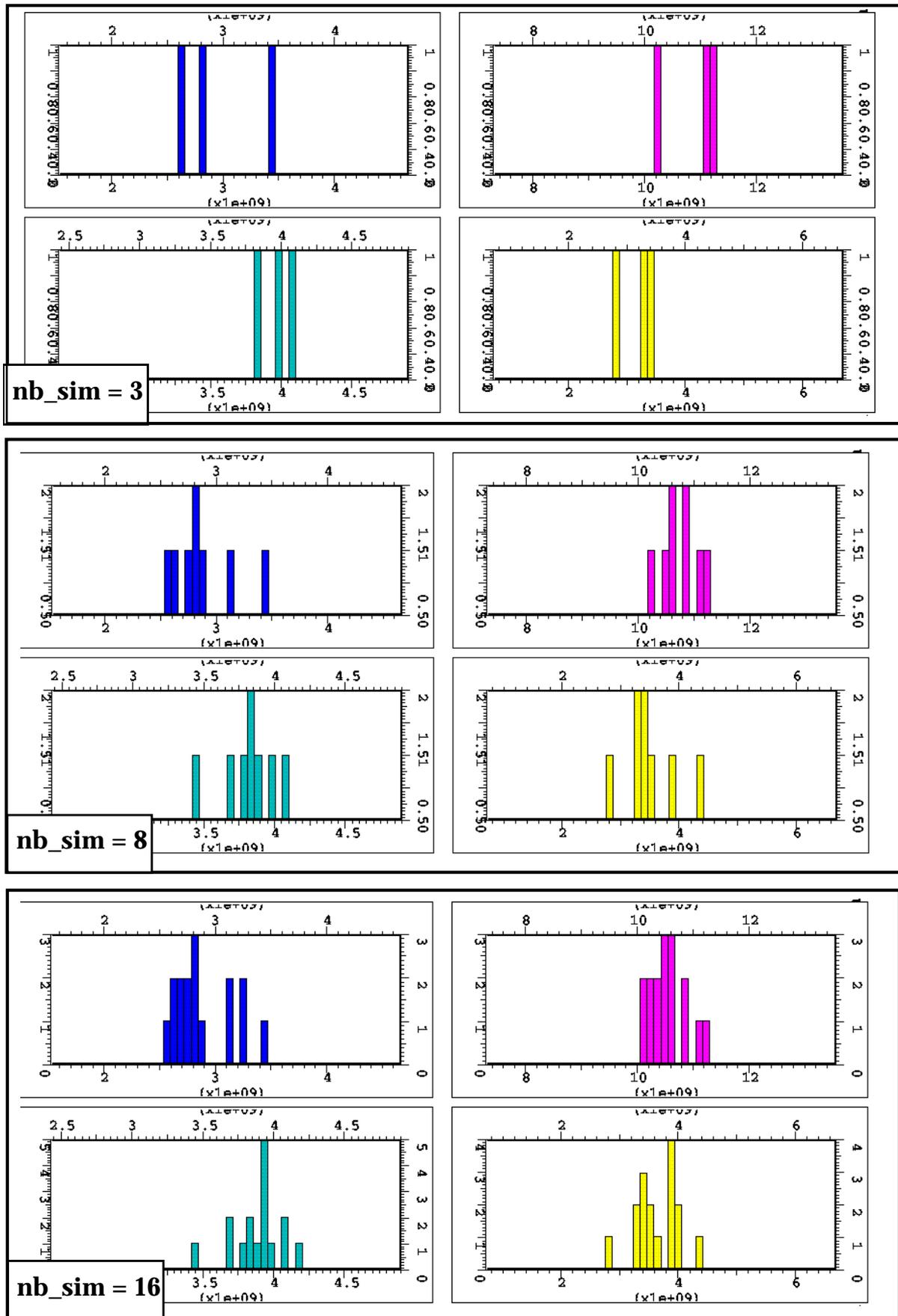


Figure 10-4 : Evolution des histogrammes au cours d'un cycle de simulations

Une fois le *code optimisé*, il sera plus facile de tester des modèles plus conséquents. Mais en théorie, le nombre de failles ou de régions dans le modèle ne joue pas sur le déroulement de l'algorithme, et si l'on est capable de reconstruire le modèle après chaque simulation, même sur des données plus complexes, le reste de l'algorithme doit suivre sans problème. Comme de nombreuses améliorations ont été réalisées sur ce point récemment, on peut être optimiste.

Un autre facteur ne permet pas encore de traiter de modèles plus complexes: **la hiérarchie n'est pas prise en compte.**

### 2.3 Comment tenir compte de la hiérarchie?

Comme cela a été dit plus haut, les failles sont traitées individuellement actuellement, autant lors de la simulation de la géométrie que durant la phase d'interpolation. Cependant, toutes les structures sont prévues pour en tenir au plus vite.

La notion de hiérarchie intervient à deux niveaux:

- **interpolation des failles "dans le bon ordre"**: actuellement, les failles sont interpolées dans l'ordre où elles sont stockées, c'est à dire dans l'ordre où elles ont été ajoutées au modèle. Il serait plus judicieux de les reclasser afin d'interpoler les failles les plus anciennes en premier, puis de remonter jusqu'aux failles les plus récentes (qui se branchent sur les précédentes).
- **simulation des failles de manière "géologique"**: comme cela a été annoncé au niveau du stockage des résultats (voir «Stockage des résultats»), il manque l'introduction de la notion de hiérarchie lors des simulations, qui permettrait de mieux respecter le schéma de faille initial (si deux failles sont éloignées d'une distance  $d$  initialement, il n'est pas acceptable d'obtenir une distance  $2d$  après simulation, parce qu'une faille se déplace vers le nord, tandis que l'autre bouge vers le sud).

#### 2.3.1 A court terme: classement de la liste des failles

Il est possible de classer la liste de failles lors de la phase de préparation. Ce classement, réalisé une fois pour toutes, peut se baser sur deux critères (qui devraient donner des résultats similaires):

- l'âge relatif des failles
- la topologie du réseau de failles

Le principe de ce tri est très simple. Une liste vide (*liste\_classée*) est créée au départ. Puis la liste de failles initiale (*liste\_non\_classée*) est parcourue. Une faille de la *liste\_non\_classée* ne peut être ajoutée dans la *liste\_classée* que si toutes les failles sur lesquelles elle se branche sont déjà dans *liste\_classée* (critère topologique), ou si toutes les failles plus anciennes sont déjà dans la *liste\_classée* (critère d'âge).

Si la faille courante ne remplit pas ce critère, elle est stockée dans une troisième liste (*à\_classer*). Une fois la *liste\_non\_classée* entièrement parcourue, on recommence le même algorithme avec la liste *à\_classer*, et ceci, tant que la liste *à\_classer* n'est pas vide.

Ces critères ne résolvent sûrement pas tous les cas possibles (failles qui se branchent l'une sur l'autre, contact multiple), mais en faisant des choix arbitraires dans ce genre de configuration, la majorité des cas devrait pouvoir se traiter (cet algorithme reste cependant à écrire et à tester).

### 2.3.2 A plus long terme: tenir compte de la hiérarchie dans les simulations

Par contre, tenir compte de cette hiérarchie durant les simulations demande un peu plus de travail. En supposant qu'elle est connue (donnée par l'utilisateur [sous quelle forme? ]), calculée automatiquement, par l'algorithme précédent ou par un autre algorithme), il faut écrire un certain nombre de structures pour en tenir compte.

Quelques pistes concernant ces structures sont données dans le dernier chapitre (voir «Objectifs restant en structurale: organisation des réseaux de failles (hiérarchie et corrélation 3D», page 241), mais elles seront plus longues à mettre en place et à tester.

Cependant, une fois ce stockage de la hiérarchie écrit, l'algorithme global n'aura pas à être beaucoup modifié, puisque les structures ont été prévues pour tenir compte de cette hiérarchie des failles.

## 3. Comment lier ces simulations à la construction de grilles faillées?

Une autre question que l'on peut se poser à propos de ce travail concerne son intégration dans l'étude plus générale des incertitudes au niveau du réservoir.

Actuellement, notre travail fonctionne en utilisant un *modèle volumique continu*, qui suffit amplement pour les études volumétriques (et même correspond à la meilleure structure pour ce type d'étude, puisque les calculs de volumes dépendent peu de la discrétisation choisie et sont précis).

Cependant, *l'étude structurale n'est pas une fin en soi*, et doit pouvoir servir à la suite de la phase d'exploration (étude stratigraphique, étude des faciès puis simulation des écoulements). Ces études se réalisant sur des *modèles discrets* (grilles suivant la stratigraphie le plus souvent), il faut s'assurer que les algorithmes présentés ici peuvent facilement être transposés du modèle volumique continu à la grille stratigraphique faillée.

### 3.1 Principe de la construction des grilles stratigraphiques faillées

La construction des grilles faillées part du même état initial que la construction du modèle volumique continu: un ensemble de surfaces de failles et de surfaces d'horizons. Par contre les éléments nécessaires à sa construction sont plus nombreux, et la construction elle-même est *plus complexe* (et relativement *instable* actuellement).

Les paragraphes suivants définissent les deux stratégies testées actuellement pour construire de telles grilles. Je tiens à remercier particulièrement Bruno Levy, Richard Cognot et Arben Shtuka pour leurs explications claires sur leur sujet de prédilection.

#### a) Mise en correspondance des bords d'horizons

Si elle est réalisée manuellement, cette étape de corrélation des bords d'horizons peut se révéler très fastidieuse lorsque le modèle est complexe.

Elle consiste à ajouter des marqueurs le long des bords des horizons, afin de *séparer les bords externes des bords de failles* et à *marquer les zones de branchement*. Dans un deuxième temps, *des liens doivent être installés entre les deux lèvres de chacune des failles*, et ceci pour chaque horizon. Cette étape est aussi réalisée manuellement.

Enfin, il faut relier les bords qui se correspondent d'un horizon à l'autre. Cette opération peut partir d'une solution obtenue de manière automatique, modifiable localement, ou être réalisée entièrement manuellement.

Cette phase de préparation ressemble donc beaucoup à ce qui est réalisé pour détecter les contacts à l'aide de l'objet `STRUCTURALMODEL`. Cet objet pourrait donc aider à automatiser ces opérations, en plus de son rôle dans la construction de modèles volumiques continus [figure 10-5].

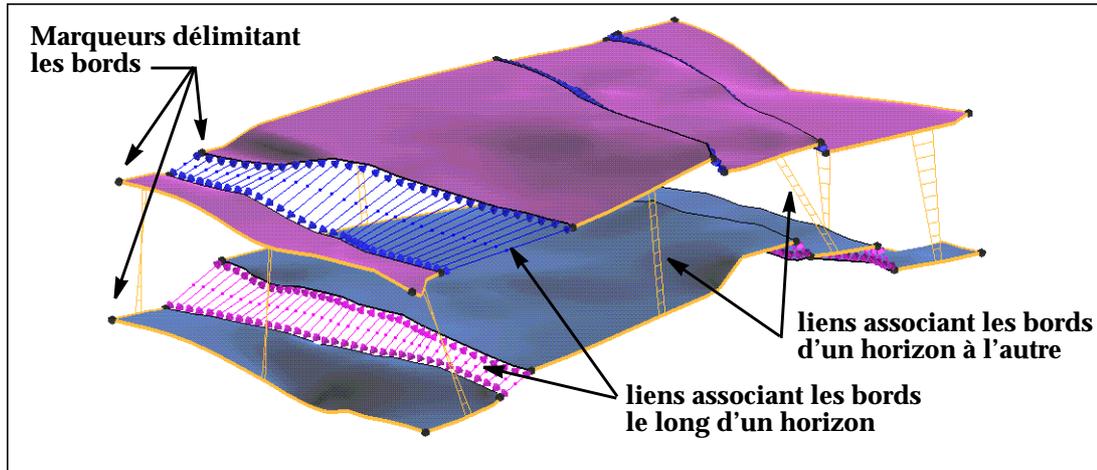


Figure 10-5 : Association des bords d'horizon

### b) Paramétrisation des surfaces d'horizon

Afin de définir la grille selon le plan horizontal, une paramétrisation en deux dimensions est réalisée [117 - Levy 2]. Elle permet de mettre en correspondance le domaine réel  $(x,y,z)$  avec un domaine  $(u,v)$ , où les failles sont éliminées et où la grille est plus facile à définir (domaine *régulier* en  $u$  et en  $v$ ).

Une bijection permet de passer facilement d'un domaine à l'autre. Ainsi, deux points situés de part et d'autre d'une faille ont les mêmes coordonnées  $u$  et  $v$ . Ceci est très important pour pouvoir calculer des propriétés de manière continue à travers les failles lorsque cela est souhaité [figure 10-6].

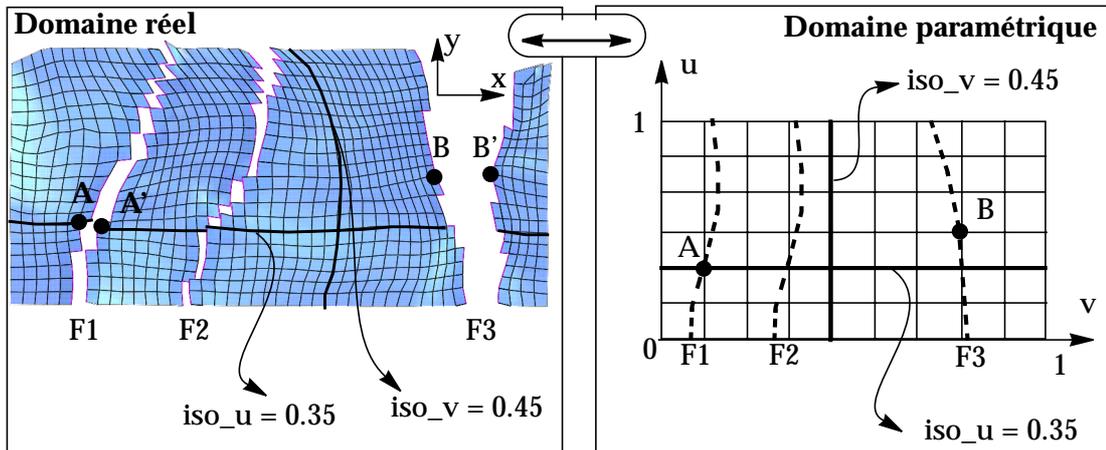


Figure 10-6 : Le passage en domaine paramétrique permet de définir la grille plus facilement.

Il est possible de contraindre cette paramétrisation de différentes manières: avoir des *cellules homogènes* (très important pour appliquer des méthodes géostatistiques ensuite) et/ou avoir des cellules les plus *orthogonales* possible. Récemment, une dernière contrainte a été ajoutée: elle permet de sélectionner certaines lignes  $iso_u$  et/ou certaines lignes  $iso_v$  parallèles à certaines lèvres de failles. Cette contrainte permet d'obtenir des cellules qui suivent les lèvres des failles sélectionnées.

### c) Définition de vecteurs guidant la construction suivant la troisième direction

Une fois que la paramétrisation est réalisée, on peut construire une grille par **extrusion verticale** à partir d'un horizon paramétrisé. Cette méthode consiste à empiler des cellules verticalement en se basant sur la paramétrisation d'un horizon. La définition d'un vecteur vertical permet de définir où l'extrusion doit s'arrêter [figure 10-7].

Il est aussi possible de définir des vecteurs obliques correspondant à des directions privilégiées (par exemple, en suivant le pendage des failles) [figure 10-7] (b).

Une autre technique consiste à déduire ces vecteurs à partir des correspondances dans l'espace paramétrique entre deux horizons de *même topologie* (les liens installés entre les deux lèvres des failles sur un même horizon indiquent quelle est l'isovaleur au niveau du bord). Cette méthode est plus complète, puisqu'elle autorise les deux horizons à ne pas avoir exactement la géométrie. Par contre la restriction concernant les topologies identiques entre les deux horizons ne permet pas de modéliser des failles qui disparaissent entre deux horizons [figure 10-7] (c).

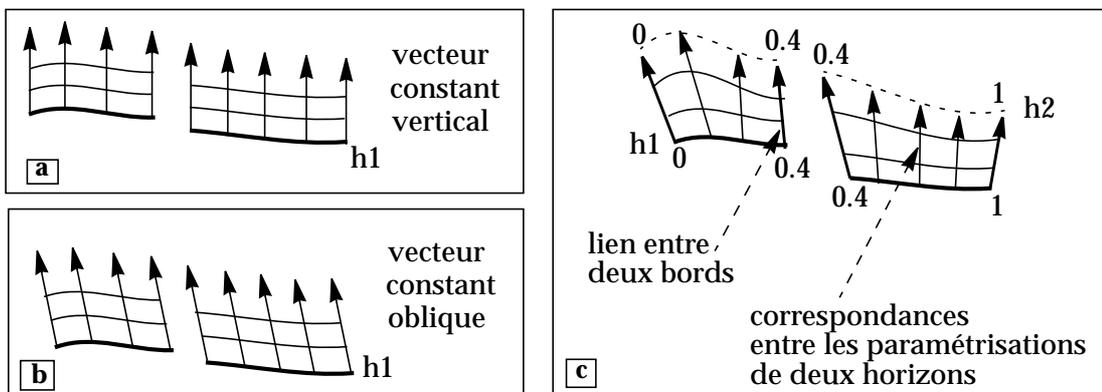


Figure 10-7 : Différentes manières de définir les vecteurs dans la troisième direction: constant vertical [a] ou oblique [b], correspondances entre horizons [c].

Une fois que certains vecteurs ont été définis au niveau des bords notamment (manuellement ou de manière automatique), une interpolation permet d'associer un vecteur guide à chacun des points de la grille issue de la paramétrisation de l'un des horizons [figure 10-8].

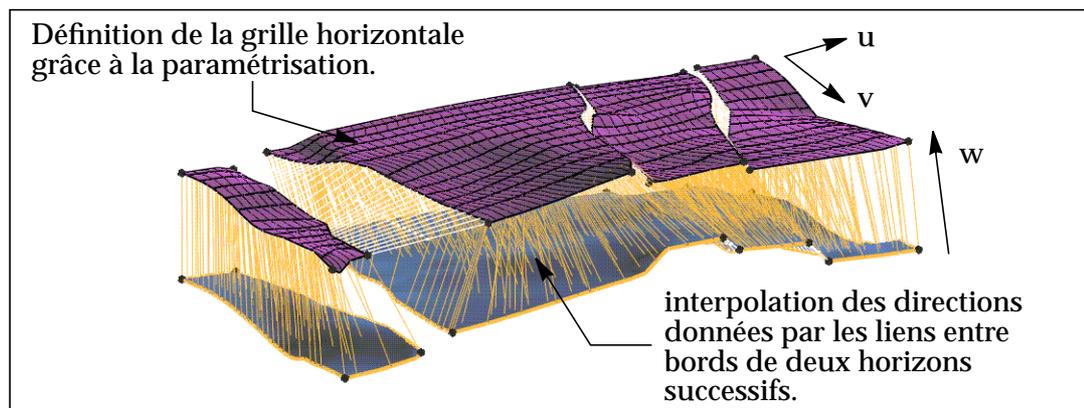


Figure 10-8 : Exemple de construction automatique des vecteurs «guides» à partir des liens installés entre deux horizons.

#### d) Construction de la grille par extrusion

Une fois que la paramétrisation est construite et que les guides suivant le troisième axe sont calculés, la grille peut être construite (quasiment instantanément pour une grille de 150 cellules suivant  $u$ , 50 suivant  $v$  et 100 cellules suivant  $w$ ).

Les vecteurs guides permettent de retrouver les correspondances entre les points des deux horizons (A et B par exemple). Ces deux points ne correspondent pas forcément à des points du maillages des surfaces triangulées, mais se correspondent dans l'espace paramétrique.

Ensuite, il suffit de découper les vecteurs guides en autant de segments que de cellules demandées selon le troisième axe, et de relier les points entre eux pour former les cellules [figure 10-9].

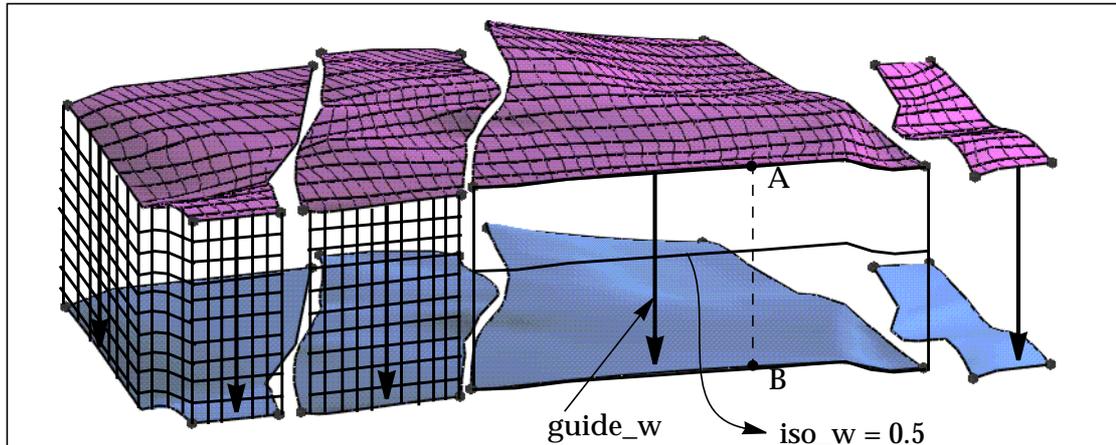


Figure 10-9 : Principe de l'extrusion verticale: empilement des cellules en utilisant la même abscisse curviligne le long des vecteurs guides.

La grille obtenue est une grille «en escalier». Une dernière étape possible, mais non nécessaire, consiste à projeter les cellules du bord sur les surfaces de faille, afin d'obtenir un contact plus précis. L'inconvénient de cette projection est qu'elle crée des cellules dégénérées qui ne sont pas forcément acceptées par tous les simulateurs d'écoulement actuellement.

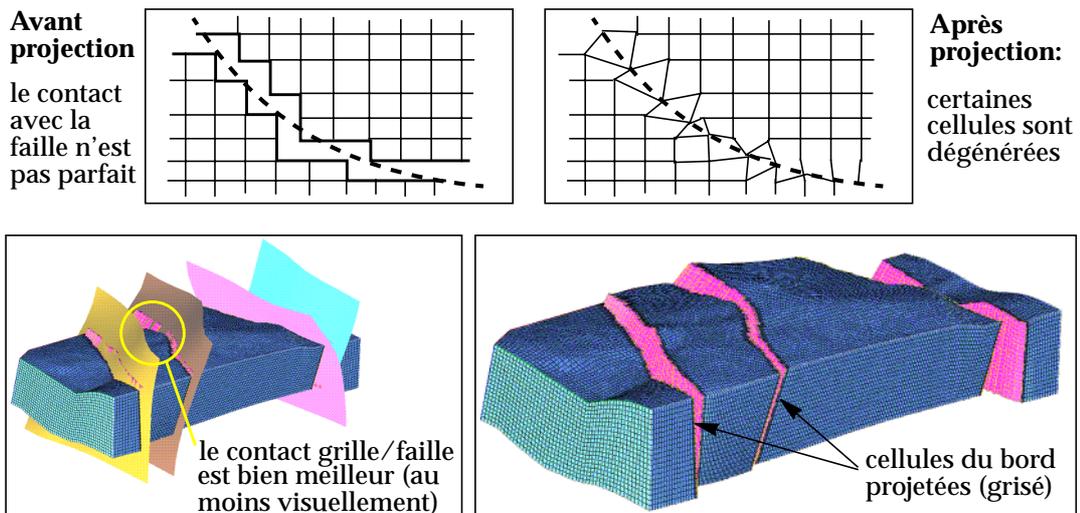


Figure 10-10 : Possibilité de projeter les cellules du bord sur les failles pour améliorer le contact (mais avec création de cellules dégénérées).

### e) Limites de cette technique

Une difficulté se produit lorsque l'on ne dispose que d'un horizon, et que ces vecteurs doivent être installés manuellement, sans pouvoir s'appuyer sur un deuxième horizon.

Ceci peut se produire lorsque l'on connaît la géométrie exacte d'un horizon, et que les horizons supérieurs ne sont définis que par des cartes donnant les épaisseurs entre deux horizons successifs. N'ayant aucune information sur la direction d'extension de la grille, il faut les définir manuellement (par exemple en prenant les normales locales). Dans certains cas (par exemple si l'horizon présente de fortes courbures), cela conduit à des configurations non valides sur lesquelles il faut retravailler [figure 10-11] (a).

Ces problèmes peuvent aussi apparaître du fait que les calculs se font sur des surfaces complexes (et définies en 3 dimensions). Si les surfaces sont trop différentes, on aboutit à des conflits qui créent des grilles non valides [figure 10-11] (b).

Enfin ces problèmes peuvent se produire lorsque l'on dispose de deux horizons, mais que l'on veut extrapoler la grille très loin de ces horizons [figure 10-11] (c).

De gros progrès ont été réalisés récemment sur ce point, afin de traiter ce genre de configurations. De plus, on peut considérer, dans un premier temps, que les cas sur lesquels nous travaillons ne rentrent pas dans ces catégories (c'est à dire que l'on a paramétrisé les différents horizons, que leur topologie est exactement la même, et qu'ils ne sont pas trop différents les uns des autres).

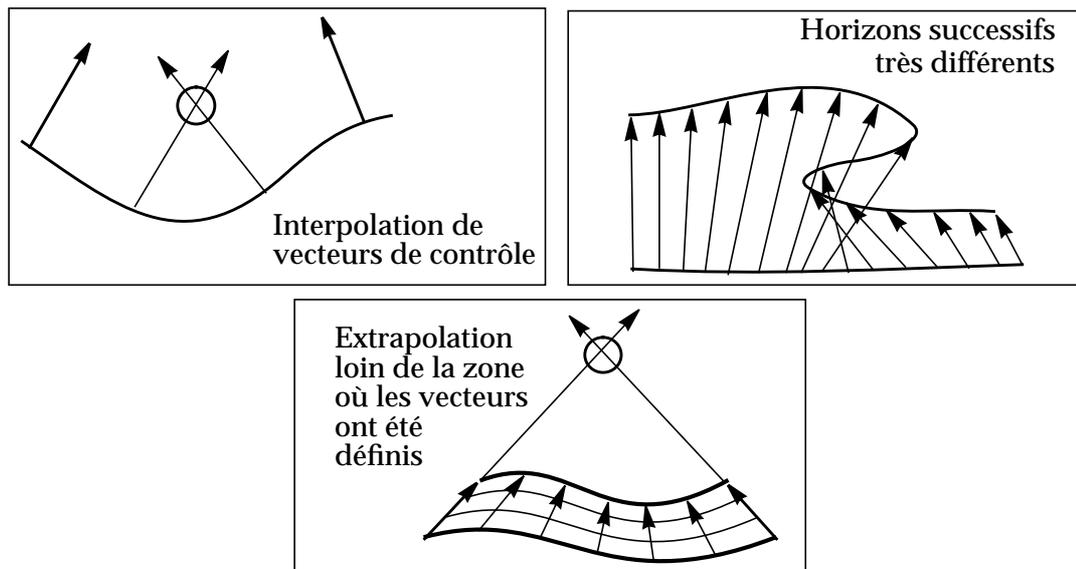


Figure 10-11 : quelques cas conduisant à la création de grilles invalides.

La dernière limitation de cette méthode de construction, est que la géométrie des failles n'est pas respectée, puisque les failles sont approximées au mieux par des plans de même pendage. La plupart des simulateurs d'écoulement ne peuvent prendre en entrée que ce type de grilles. En effet, les équations à résoudre pour déterminer le mouvement des fluides sont complexes et certaines restrictions sont faites pour simplifier la résolution des systèmes (sur la forme des cellules par exemple) [113 - Quettier].

Cependant, cette technologie évoluant, certains utilisateurs préféreraient travailler avec des grilles qui suivent la géométrie des failles au plus près, afin de réaliser des études plus précises.

C'est pourquoi une autre technique de construction est expérimentée actuellement [112 - Cognot 3].

### f) Paramétrisation dans les trois dimensions de l'espace

Plutôt que de réaliser une paramétrisation en deux dimensions et de chercher des vecteurs pouvant guider la construction de la grille dans la troisième direction, cette méthode consiste à réaliser une paramétrisation directement en trois dimensions [*paramétrisation 3D*] (en mettant le modèle surfacique dans son entier en correspondance avec une *grille régulière en trois dimensions*).

Ceci permet de définir certaines failles comme des surfaces (et non plus des courbes) «*iso\_u*», «*iso\_v*» ou «*iso\_uv*». Ainsi, dans l'espace réel, les cellules s'empilent exactement le long des failles sélectionnées. De plus, un certain nombre de contraintes peuvent être ajoutées (*homogénéité* et *orthogonalité* des cellules par exemple).

La figure suivante montre de manière schématique comment la paramétrisation 3D se déroule [figure 10-12].

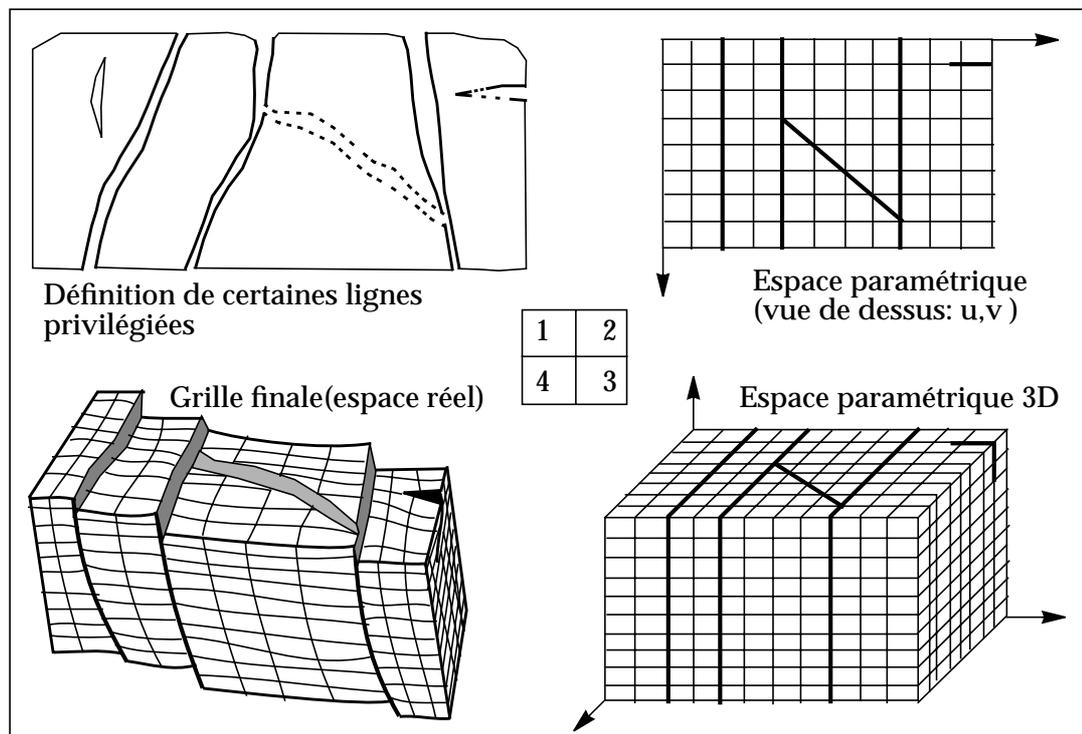


Figure 10-12 : Construction de la grille en utilisant une paramétrisation 3D.

On peut se demander si la paramétrisation en trois dimensions n'ajoute pas un degré de difficulté supplémentaire, et ne risque pas d'être beaucoup plus instable.

En effet, si les vecteurs guide ont tendance à se croiser dans la méthode précédente, on peut imaginer que le problème peut apparaître aussi avec la méthode de paramétrisation 3D.

Pour résoudre ce problème, l'une des contraintes intéressante consiste à forcer le gradient suivant les différentes directions à être toujours positif, ce qui devrait éliminer les problèmes de croisement. Avec cette contrainte, les courbes isovaleurs qui posent problème tendent à devenir parallèles sans se croiser.

Cette méthode est donc très prometteuse, mais elle n'en est qu'au stade expérimental et encore peu utilisée [figure 10-12].

### 3.2 Mise à jour après les simulations

Comme il existe deux manières de construire une telle grille, on peut envisager deux manières de la mettre à jour après un cycle de simulation: soit à partir d'une grille type «*paramétrisation 3D*», soit à partir d'une grille «*extrusion*» (basée sur une paramétrisation 2D).

#### 3.2.1 Grille issue de la paramétrisation 3D

Si la grille est issue d'une paramétrisation 3D, un certain nombre de choses sont définies sous formes de contraintes (notamment les failles *iso\_u*, *iso\_v* et *iso\_uv*). On pourrait donc imaginer de travailler avec la grille initiale durant tout le cycle de simulation.

Cependant, afin de répercuter les modifications de géométrie de manière homogène sur toute la grille, il faudrait un très grand nombre d'itérations lors de l'interpolation, et cela risque d'être long (ce qui est incompatible avec la génération d'un très grand nombre de modèles). Cette homogénéité des cellules est indispensable pour que les méthodes géostatistiques soient valides. En effet, elles sont basées sur le fait que dans l'espace (u,v,w), les points sont à peu près équidistants.

De plus, cette méthode de génération de grille étant encore au stade *expérimental*, elle demande beaucoup de travail manuel (spécification des surfaces de failles qui correspondent à des isovaleurs dans le domaine paramétrique par exemple).

En attendant que cette méthode mûrisse, c'est donc la deuxième solution qui est envisagée (et qui est déjà utilisée dans la version actuelle du logiciel JACTA, mais uniquement avec des failles planes par morceaux pour l'instant).

#### 3.2.2 Grille issue de l'extrusion

L'utilisation de ce type de grille permet d'utiliser au maximum ce qui a été réalisé pour construire des modèles volumiques continus, notamment si l'objet STRUCTURAL-MODEL est utilisé:

- **l'appariement des différents bords d'horizon** se fait de la même manière pour les deux types de modèles
- l'installation des contraintes entre les différentes surfaces du modèle permet de **mettre à jour le modèle surfacique** après chaque itération
- une fois les interpolations réalisées, la reconstruction de la grille demande, a priori, peu de travail (cette affirmation reste encore à vérifier):
  - l'interpolation des horizons met automatiquement la **paramétrisation à jour** (il existe une correspondance directe entre le domaine réel et le domaine paramétrique). La paramétrisation n'a donc pas besoin d'être complètement recalculée. Cependant, il faudra utiliser un grand nombre d'itérations afin de conserver des cellules homogènes.
  - comme les variations de géométrie le long des failles sont relativement restreintes, le **champ de vecteurs qui servent de guide** change peu d'une simulation à l'autre. En partant d'une solution initiale proche du champ de vecteur de la simulation courante, la mise à jour doit donc être assez rapide (il faut pouvoir contraindre les vecteurs qui se trouvent sur le bord).
  - la construction de la grille elle-même prend assez peu de temps.

### 3.2.3 Conclusion sur le passage aux grilles stratigraphiques faillées

Le code réalisé pour construire des modèles volumiques continus puis pour les mettre à jour après chaque simulation semble donc utilisable pour les grilles stratigraphiques faillées construites par extrusion.

Cependant, la construction des grilles faillées est encore un domaine de recherche et des problèmes restent encore à résoudre. Par exemple, la construction d'une paramétrisation correcte n'est pas évidente, du fait des imprécisions numériques par exemple. De plus, comme nous l'avons souligné, la définition des vecteurs guides peut conduire à des configurations qui ne permettent pas toujours de construire une grille faillée.

Ce passage aux grilles faillées devrait être réalisé dans les prochains mois, et conduire à une première version de Jacta permettant de modifier non seulement la géométrie des horizons, mais aussi celle des failles.

Compte tenu des problèmes rencontrés actuellement pour construire ces grilles, les premiers tests ne pourront s'effectuer que sur des modèles assez simples.

Le chapitre suivant donne quelques idées sur ce qui pourrait être fait à plus long terme, pour améliorer les algorithmes actuels, en tenant compte de la hiérarchie du réseau de failles notamment.

---

---

# Chapitre 11 Objectifs restants en structurale: organisation des réseaux de failles (hiérarchie et corrélation en 3 D)

---

---

Les chapitres précédents ont montré comment étudier les incertitudes à l'échelle du réservoir. Cependant, tous les problèmes ne sont pas encore résolus en ce qui concerne le **respect de la hiérarchie initiale** du réseau de failles.

De plus, l'étude des incertitudes ne se limite pas à l'étude des petites variations de la géométrie autour d'un *scénario donné* (ce qui a été choisi comme limitation pour ce travail).

Pour étudier complètement les incertitudes liées à la géométrie des failles, il faut aussi être capable de **proposer différents scénarios** pour le réseau de failles, en fonction des données disponibles. En effet, si connaître les différents styles tectoniques est une chose, arriver à les reconnaître sur les sections sismiques en est une autre. Les interpréteurs se trouvent souvent confrontés à la difficulté de corréler en trois dimensions les interprétations qu'ils ont faites en deux dimensions.

A cela s'ajoutent les problèmes liés au choix du modèle de vitesses pour procéder à la migration.

Ce chapitre décrit brièvement les problèmes, puis donne quelques pistes pour les travaux futurs.

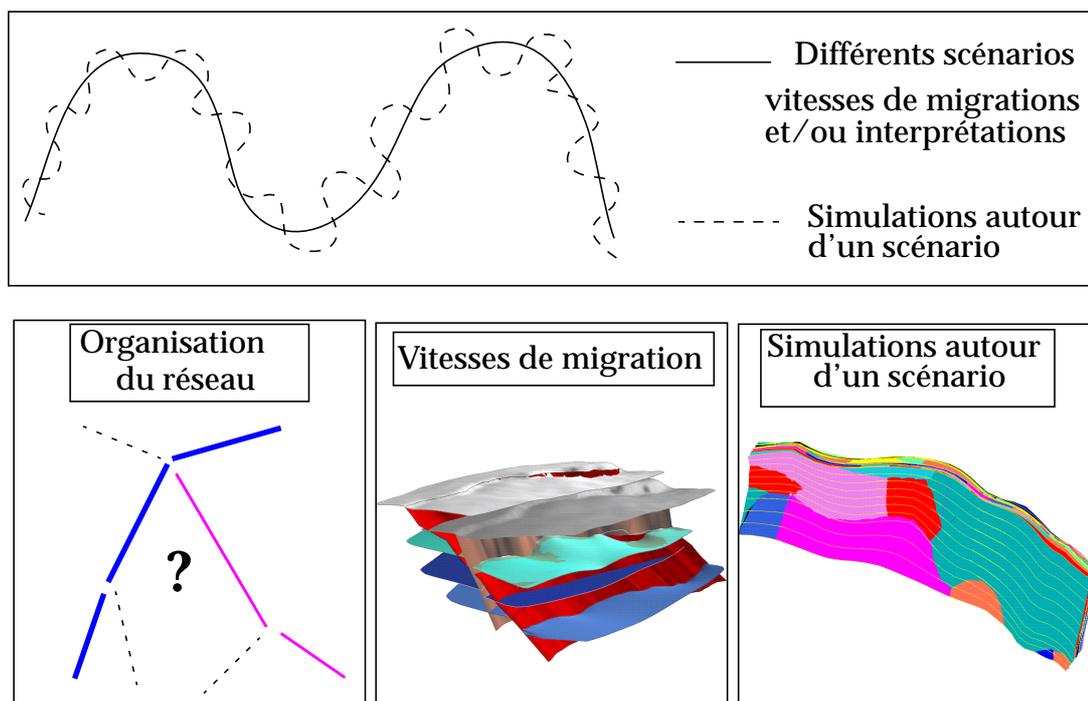
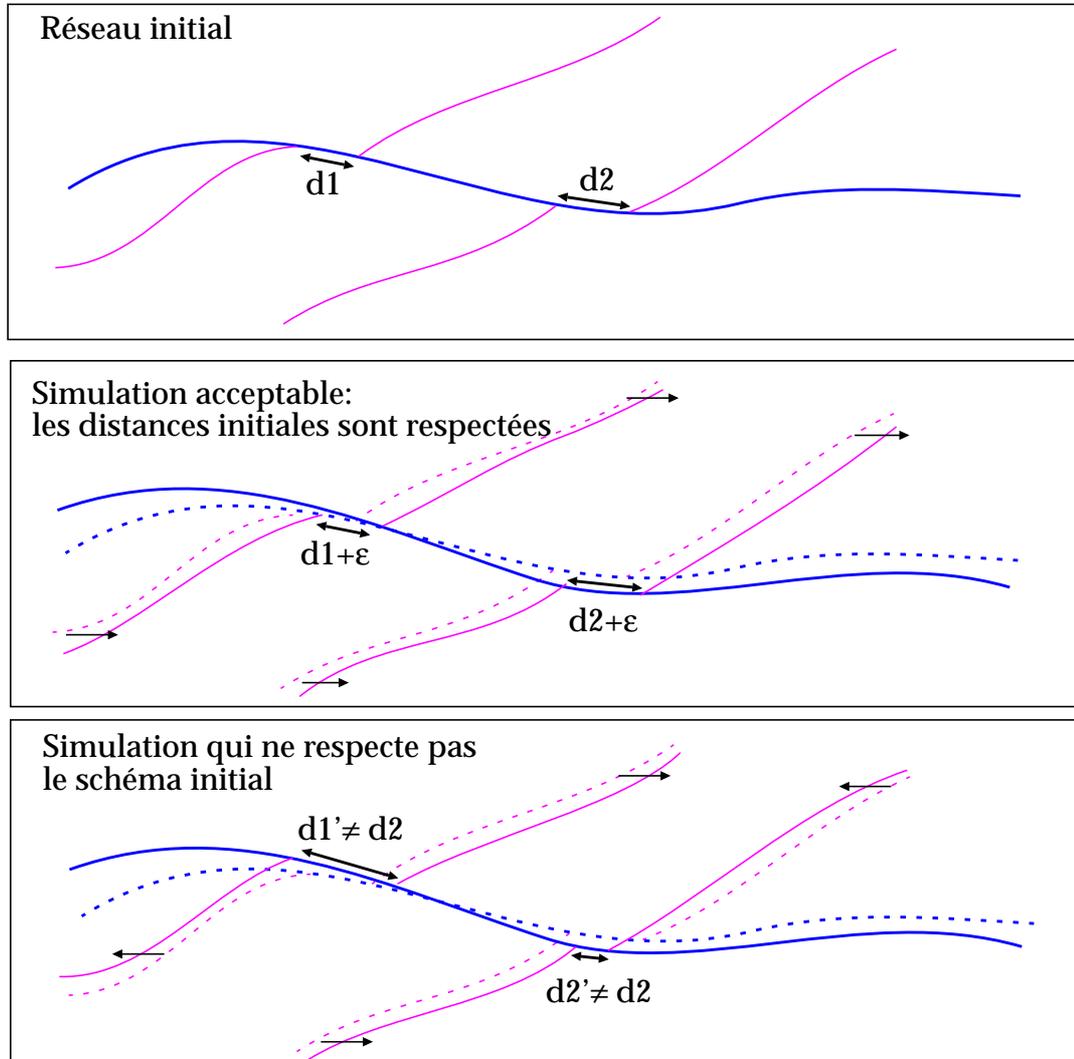


Figure 11-1 : Les différentes échelles de l'étude des incertitudes sur les failles.

## 1. Hiérarchisation

Lors des simulations présentées dans les chapitres précédents, il est important de respecter la hiérarchie des failles. Les failles principales doivent être simulées dans un premier temps et les familles de failles secondaires doivent être simulées ensuite, en respectant le schéma initial [figure 11-2].



**Figure 11-2 :** Respect du schéma de faille initial pendant les simulations: les failles recoupées par une faille plus récente doivent être déplacées dans le même sens afin de respecter les distances initiales.

### 1.1 Comment définir le hiérarchie du réseau de failles

#### 1.1.1 Les systèmes de failles

Les différentes failles présentes dans une aire donnée peuvent être regroupées en **familles**, chacune caractérisées par un *azimut* (angle par rapport au Nord calculé dans le sens des aiguilles d'une montre) et un *sens de rejet* propre. Souvent, on rencontre deux *familles conjuguées* de types de rejet identiques mais de rejets opposés.

Lorsque l'on étudie la hiérarchie dans un réseau de failles, on distingue de plus les failles **majeures** des failles **mineures**. Dans une famille donnée, les failles mineures sont dites *satellites* des failles majeures; celles dont la formation résulte du jeu de la faille majeure sont appelées failles *secondaires*.

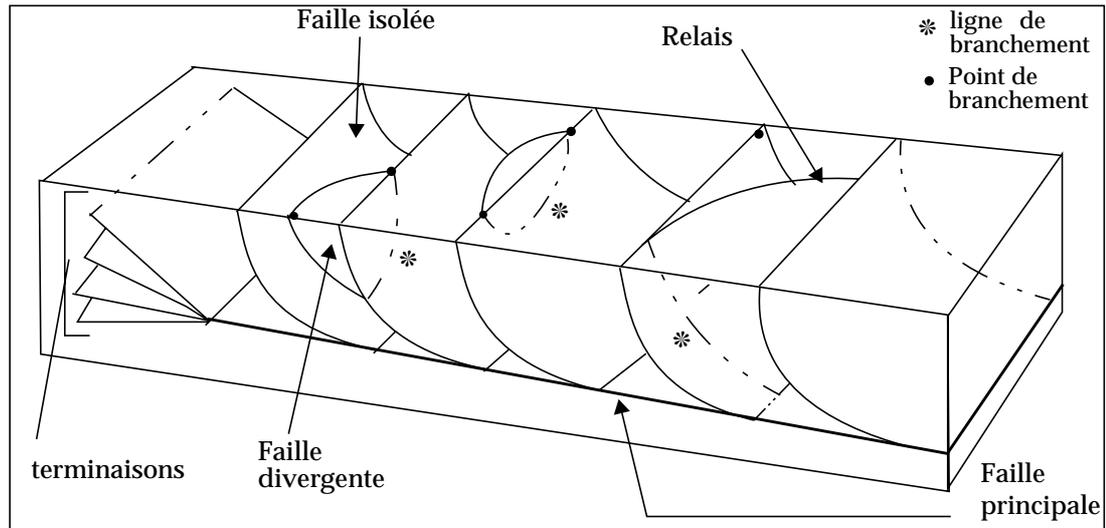


Figure 11-3 : Terminologie des réseaux de failles (d'après [44 - Ramsay]).

Les relations entre les différentes failles d'une même famille peuvent être très complexes à déterminer, surtout lorsque les failles secondaires sont nombreuses et que la modélisation en 3 dimensions n'est pas utilisée. On distingue en effet [figure 11-3]:

- Les failles isolées
- Les failles relayant deux failles principales
- Les failles secondaires divergeant à partir de la faille principale
- ...

### 1.1.2 La détermination structure Majeure/mineure dépend beaucoup de l'échelle d'observation

Comme souvent en géologie, la notion d'échelle est très importante. Une structure mineure à grande échelle peut être une structure Majeure à plus petite échelle et jouer un rôle important dans la compartimentation du réservoir [figure 11-4].

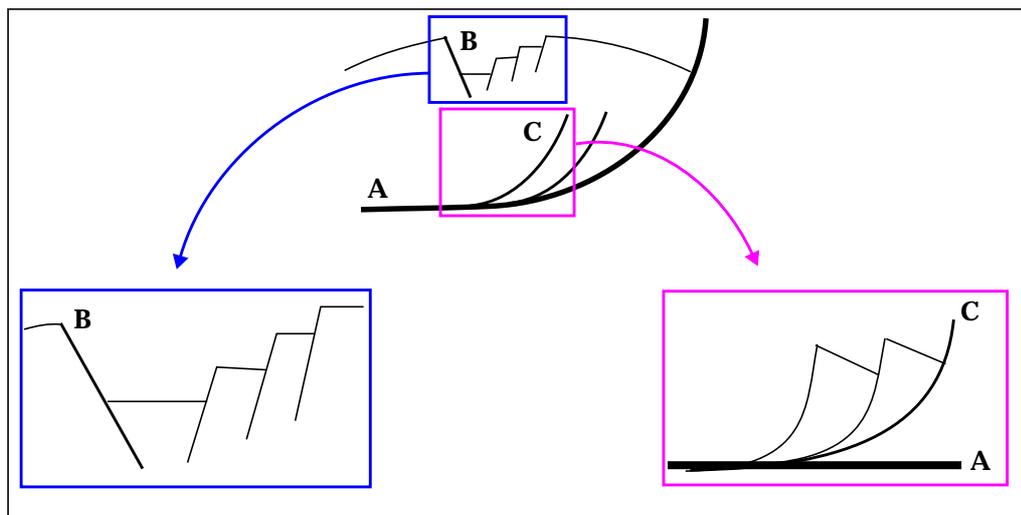


Figure 11-4 : A grande échelle les failles B et C sont des failles mineures de la faille A, tandis qu'à petite échelle, ce sont des failles majeures.

### 1.1.3 Cohérence entre les coupes et les cartes

Pour déterminer le caractère majeur ou mineur d'une faille, on peut utiliser des choix plus ou moins arbitraires. Il faut toutefois garder à l'esprit que ces guides ne sont pas des règles strictes.

- Généralement, les failles mineures s'interrompent sur la faille majeure auxquelles elles sont associées en coupe et elles ont une longueur moindre que la Faille Majeure en carte [figure 11-5].

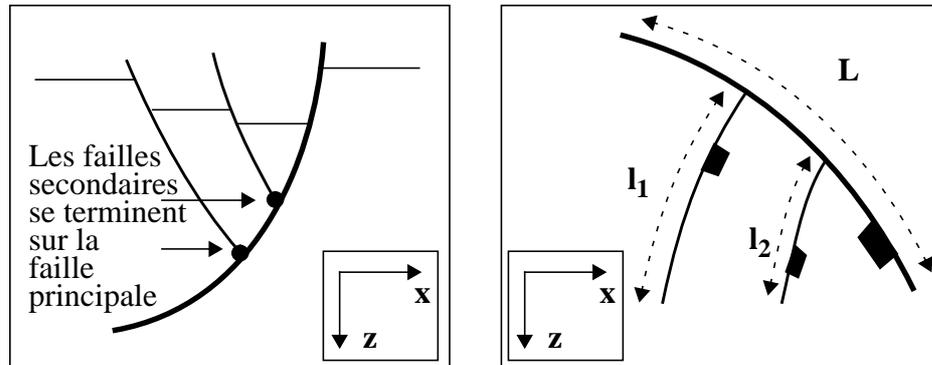


Figure 11-5 : Relation entre les failles secondaires et les failles principales. [a]: en coupe, [b]: en carte.

- De plus, les **échelles relatives** entre les structures Majeures et les structures mineures sont en général **identiques en coupe et en carte**. Il faut toutefois faire attention aux décrochements. En effet, dans ces zones, le rejet horizontal est très supérieur aux rejet vertical et une faille majeure apparaît souvent comme une faille mineure en coupe.
- Enfin, les **géométries en carte et en coupe** sont souvent **semblables**: une faille sans courbure en coupe aura plus facilement une trace rectiligne en carte. De même, une faille listrique en coupe aura le plus souvent une trace courbe en carte [figure 11-6]. Ces règles ne s'appliquent pas toujours ([54 - Walsh 1]).

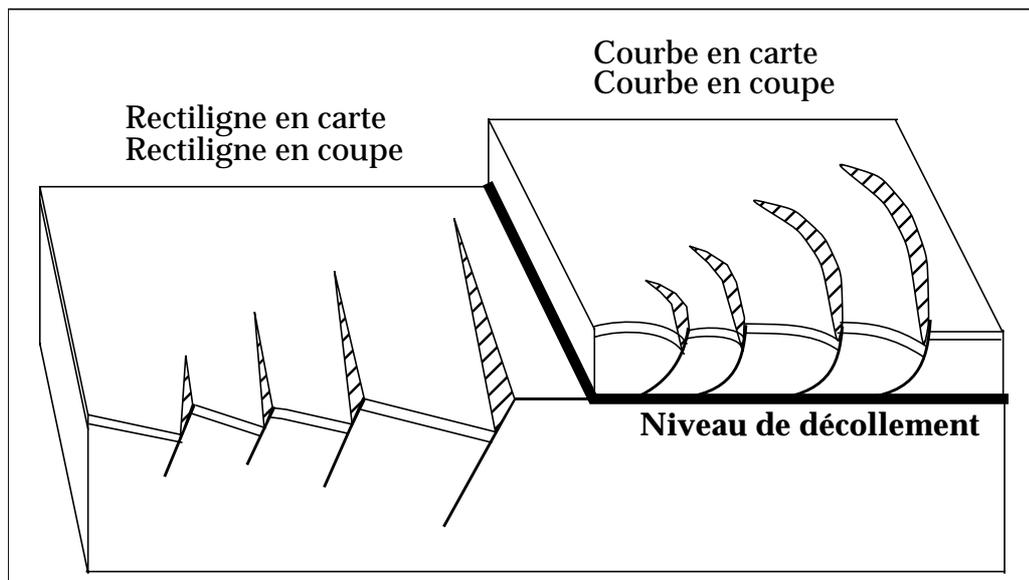


Figure 11-6 : La géométrie des failles est souvent la même en carte et en coupe.

## 1.2 Comment préserver cette hiérarchie lors des simulations

Bien que cette partie du travail n'ait été qu'effleurée, nous avons quelques idées sur la façon de procéder pour respecter la hiérarchie initiale lors des simulations. Pour cela, il faudrait de nouvelles structures de données qui, faute de temps, n'ont pas été mises en place, mais qui le seront peut-être plus tard.

### 1.2.1 Stocker les familles de failles et leur ordre d'apparition

Les structures de données seront décrites à partir de l'exemple simple ci-dessous pour plus de clarté. Supposons que l'on ait un modèle simple issu de trois événements tectoniques successifs ayant conduit à la formation de trois familles de failles [figure 11-7].

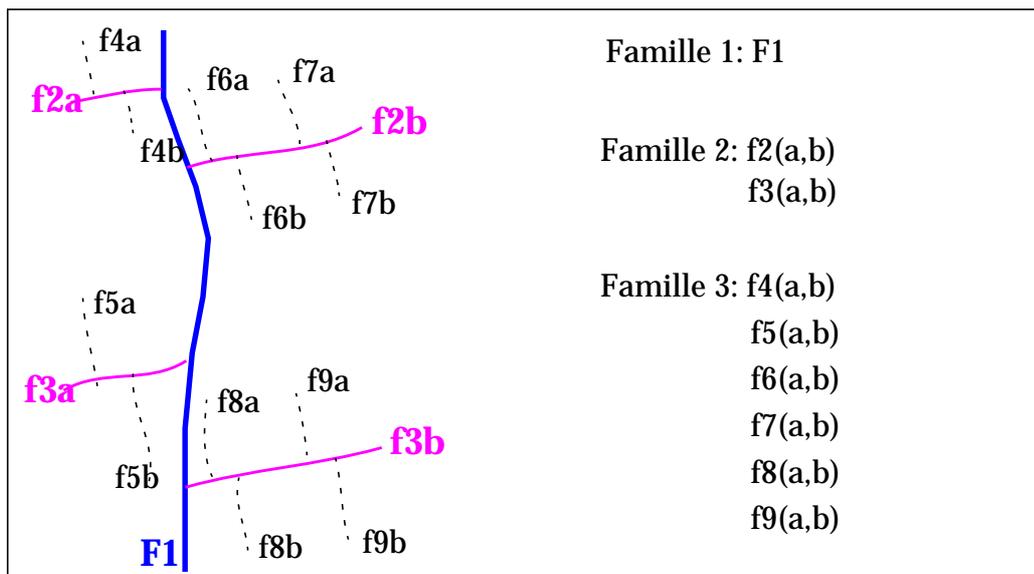


Figure 11-7 : Exemple montrant les trois familles de failles

Plutôt que de regrouper les failles simplement par famille, une notion de chronologie peut être insérée, en regroupant toutes les failles correspondant à un même événement tectonique dans une classe "Groupe de Failles" (ou GFAULTBUNDLE). Une autre classe, "Événement Tectonique" (ou TECTOEVENT) serait, elle, chargée de stocker les relations entre une faille et les groupes de failles qu'elle recoupe [figure 11-8].

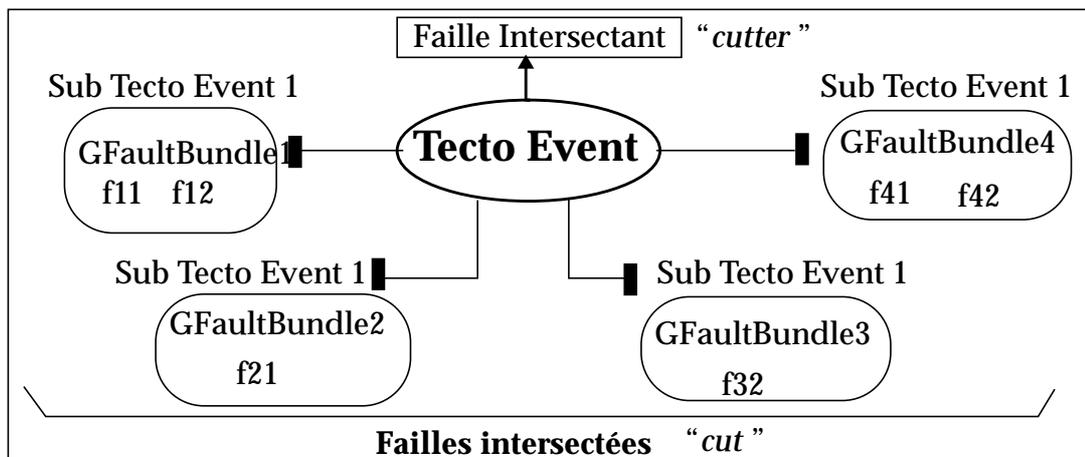


Figure 11-8 : La structure de donnée "TectoEvent": toutes les failles recoupées par une même faille sont stockées.

Ainsi, l'exemple précédent pourrait être représenté comme suit:

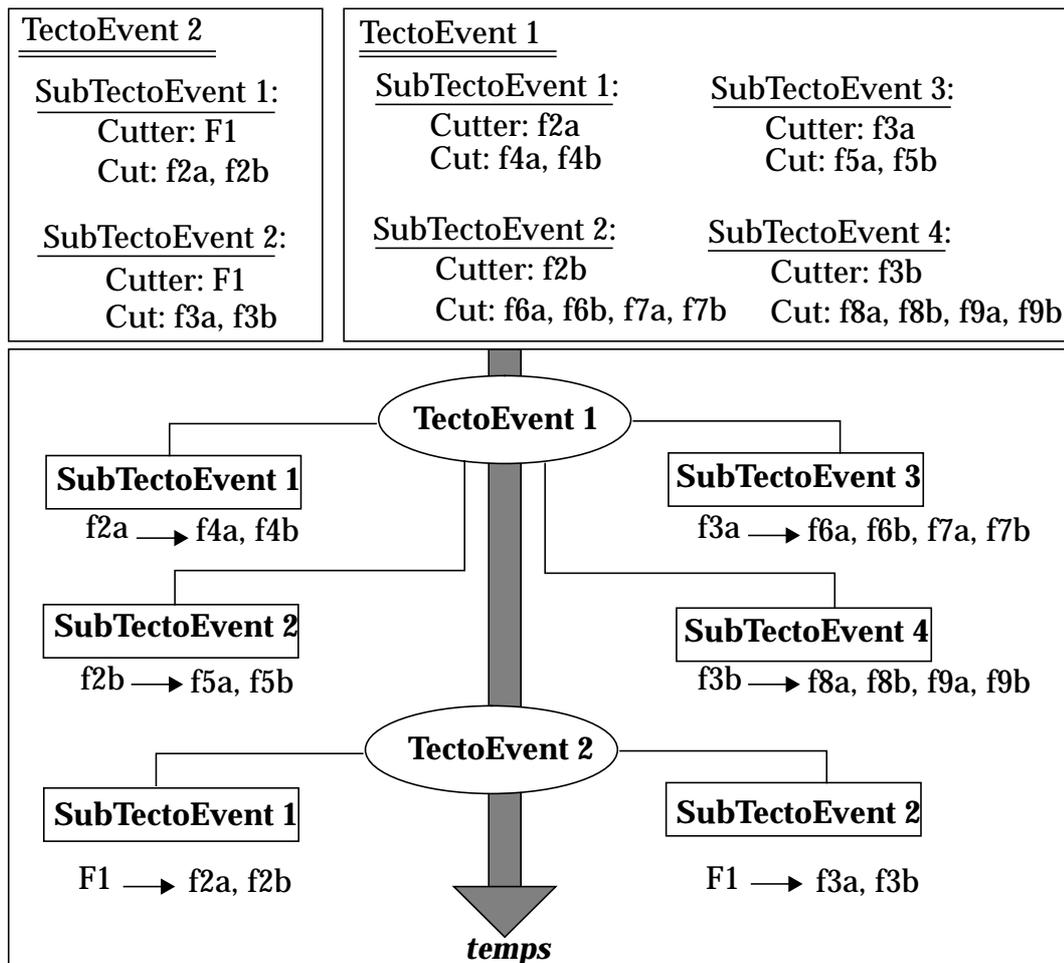


Figure 11-9 : Stockage de la hiérarchie du modèle de la [figure 11-7]

### 1.2.2 Simuler les failles en fonction de cette hiérarchie

Une fois qu'une telle hiérarchie est stockée, il suffit de simuler les failles selon les règles suivantes:

- Simulation de la faille «coupante» la plus récente (F1)
- Simulation de toutes les failles recoupées par la précédente, en corrélant les simulations d'une faille à l'autre à l'intérieur d'une même famille (les nombres aléatoires choisis doivent être proches pour respecter les distances initiales entre les failles de cette famille) (f2a et f2b d'une part, f3a et f3b d'autre part).
- Procéder ainsi de suite jusqu'à simuler les failles coupées par l'événement tectonique le plus ancien.

### 1.2.3 Critique de la méthode

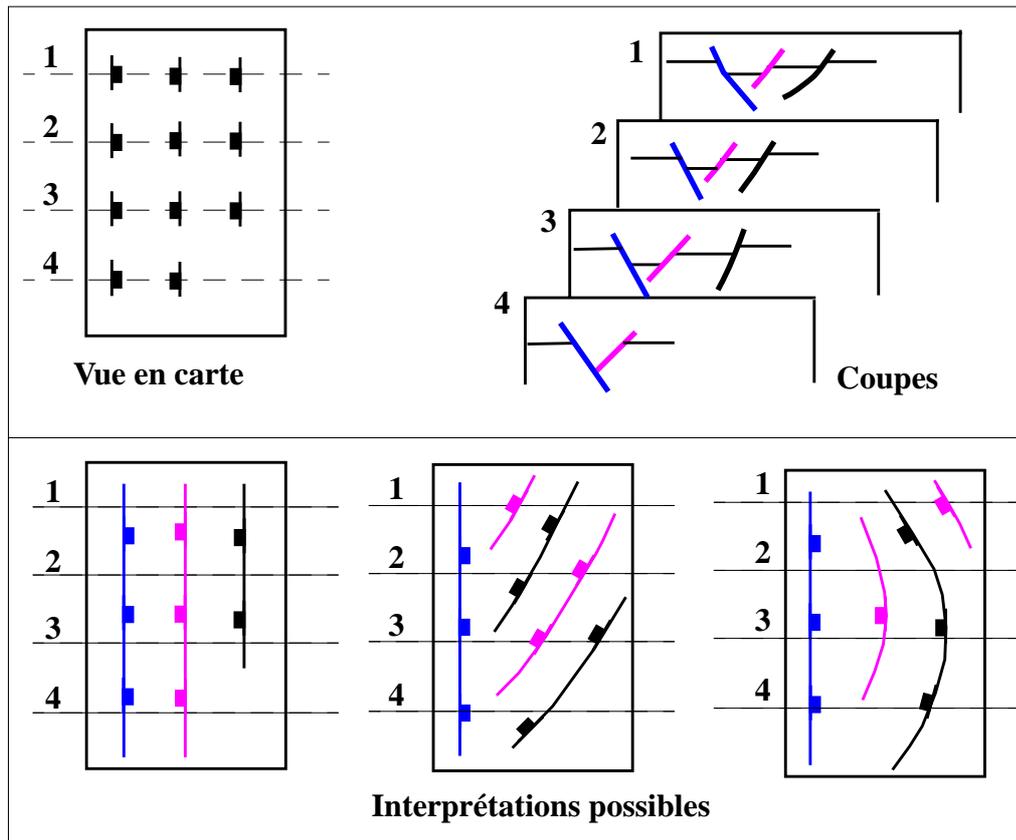
Ce début d'algorithme n'a pas été mis en place faute de temps, mais il est sûr que les cas réels ne sont pas aussi simples que le modèle présenté ici, et que des conflits devraient être gérés. Il reste donc pas mal de travail pour pouvoir utiliser une telle méthodologie sur des cas réels complexes. Toutefois, utiliser une telle méthode me semble être la solution la plus "géologique".

De plus, cette méthode devrait incorporer d'autres éléments géologiques tels que l'épaisseur des couches traversées, la liste des horizons recoupés... Ces critères restent encore à définir pour obtenir une hiérarchie tenant compte de tous les éléments.

## 2. Etude d'un autre type d'incertitudes: proposition de plusieurs scénarios.

### 2.1 le problème de la corrélation des segments de failles en 3D

L'interprétation des sections sismiques est réalisée en deux dimensions actuellement. En fin d'interprétation, l'interpréteur a déterminé un ensemble de marqueurs qu'il doit corrélater en trois dimensions. Généralement les différentes corrélations sont toutes valables du point de vue géologique, et seule sa connaissance de la géologie régionale ainsi que certaines règles simples lui permettent de choisir entre les différentes possibilités [figure 11-10].



**Figure 11-10 :** la corrélation en trois dimensions des sections sismiques interprétées en 2 dimensions n'est pas aisée: **différentes interprétations sont possibles** du point de vue géologique et le géologue doit faire des choix (connaissance de la géologie régionale, études des rejets....) (D'après [121 - Freeman])

Actuellement, l'interpréteur et le structuraliste proposent une solution unique pour l'organisation du réseau de failles, qui a été sélectionnée en fonction de critères plus ou moins objectifs. En plus de l'incertitude sur la position exacte des failles étudiée au cours de nos travaux, il persiste donc des doutes sur le schéma de failles lui-même. Du fait du temps mis pour construire le modèle en trois dimensions, il est relativement rare actuellement que plusieurs scénarios soient testés au cours d'une étude.

Après avoir montré les règles utilisées pour vérifier la cohérence du modèle, cette section décrit comment l'objet STRUCTURALMODEL utilisé pour les simulations pourrait aussi servir à comparer différents scénarios, en éditant le modèle initial "par morceaux". Ceci permettrait de se consacrer aux zones délicates et de conserver le travail effectué sur les zones plus simples du point de vue tectonique.

## 2.2 Règles qui peuvent être utilisées pour vérifier la cohérence

Pour sélectionner les configurations acceptables du point de vue géologique et réduire le nombre de possibilités, le géologue structuraliste dispose de nombreuses règles de bon sens. La plupart des règles détaillées ci-dessous sont tirées d'explications données par les structuralistes (principalement Sylvain Calassou et Gérard Ionesta) et les interpréteurs d'Elf Exploration Production.

### 2.2.1 Tenir compte des rejets

- Généralement, seules les failles de **rejets équivalents**, et de même type sont corrélées: les failles normales et les failles inverses sont corrélées indépendamment. Cependant, en contexte décrochant, il est possible de corréler des failles de types différents, puisque l'on trouve de fortes variations du rejet dans ces zones.
- Il existe des règles permettant de vérifier la cohérence d'un ensemble de failles. Que ce soit en extension ou en distension, on peut généralement appliquer la règle suivante: "le rejet de failles coalescentes et de même regard s'additionne pour produire un segment de faille dite majeure " [figure 11-11].

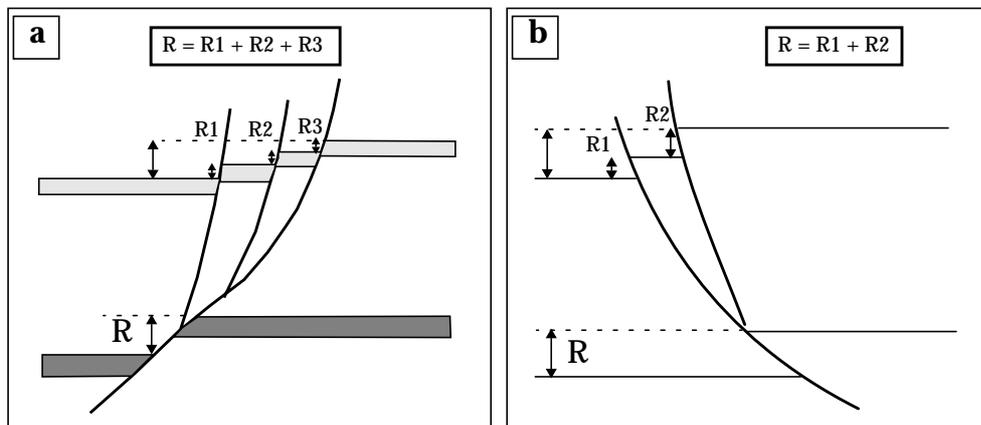


Figure 11-11 : Utilisation des rejets pour vérifier la cohérence d'un groupe de failles. [a]: en extension; [b]: en compression.

Cependant, cette règle ne s'applique pas dans le cas de failles normales *antithétiques*. De plus, en cas d'anomalie, il ne faut pas hésiter à envisager une tectonique décrochante.

- De la même façon, lorsque l'on corréle des failles de rejets différents, il faut vérifier que le rejet maximum est au centre, et qu'il décroît latéralement vers les extrémités [figure 11-12].

Cette règle est toutefois à moduler en fonction de la complexité de la zone. Des précisions sur ce point sont données en fin de ce chapitre.

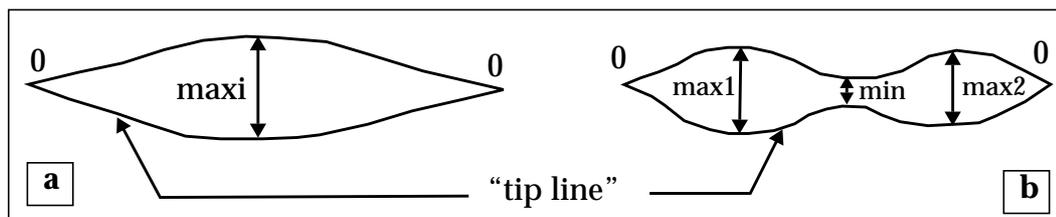


Figure 11-12 : Corrélation à l'aide des rejets: la solution [b] est très improbable.

### 2.2.2 Utiliser les horizons

Deux failles peuvent être corrélées si elles sont cachetées par un même horizon. Par contre, une faille cachetée par un horizon ne peut être corrélée avec une faille affectant cet horizon, sauf s'il existe une variation latérale du rejet de la faille [figure 11-13].

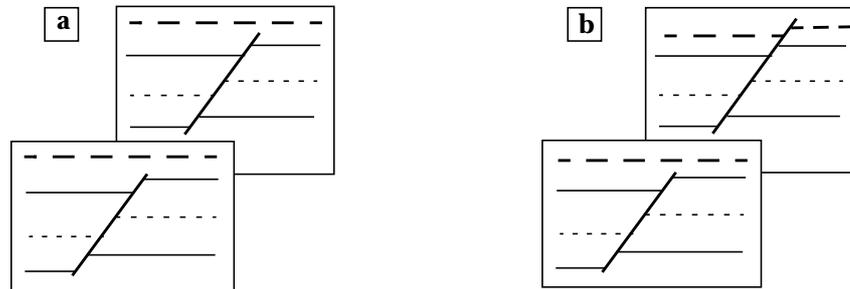


Figure 11-13 : [a]: la corrélation est possible; [b]: la corrélation est déconseillée

Les structuralistes ont sûrement beaucoup d'autres règles pour valider la cohérence de leur modèle, mais elles sont difficiles à formaliser et à obtenir.

### 2.3 Comment tester plusieurs scénarios

Comme cela a été montré précédemment, il est parfois difficile de relier les morceaux de failles bien identifiés pour construire un réseau cohérent. Proposer au structuraliste un outil lui permettant de tester plusieurs scénarios dans une zone délicate du modèle est une autre application du STRUCTURALMODEL que celle utilisée pour simuler la géométrie autour d'un scénario donné.

#### 2.3.1 Les cas de figures où cela pourrait être utile

Dans les zones en «X» ou en «Y», il est parfois difficile de trancher entre les différentes possibilités:

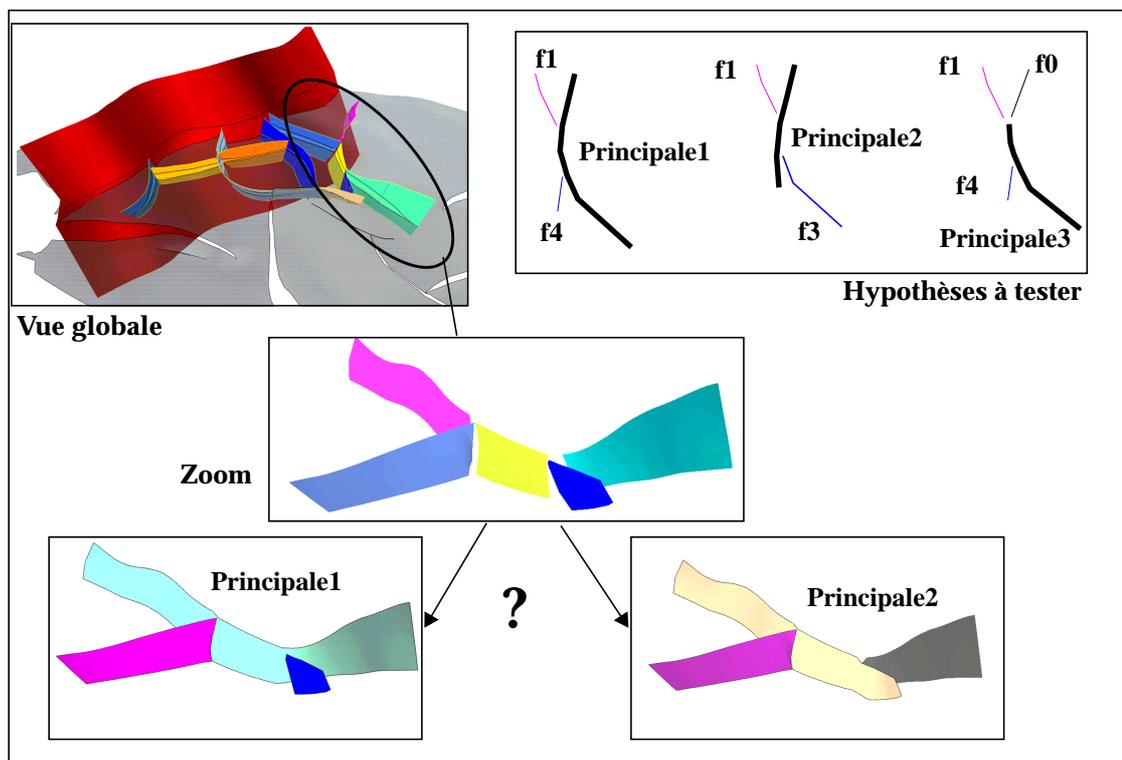


Figure 11-14 : Tester plusieurs hypothèses pour les connections entre failles.

### 2.3.2 Notion de Modèle Composite

Une des possibilités prometteuses de l'objet STRUCTURALMODEL est l'aspect **modulaire** des objets qui le composent. Il est possible de connecter plusieurs failles en une seule de manière **réversible**.

Même si ces fonctionnalités ne sont accessibles actuellement qu'à partir de la représentation sous forme de lignes polygonales, cela ouvre beaucoup de possibilités pour l'édition rapide du réseau de failles et donc le test de plusieurs scénarios [figure 11-15].

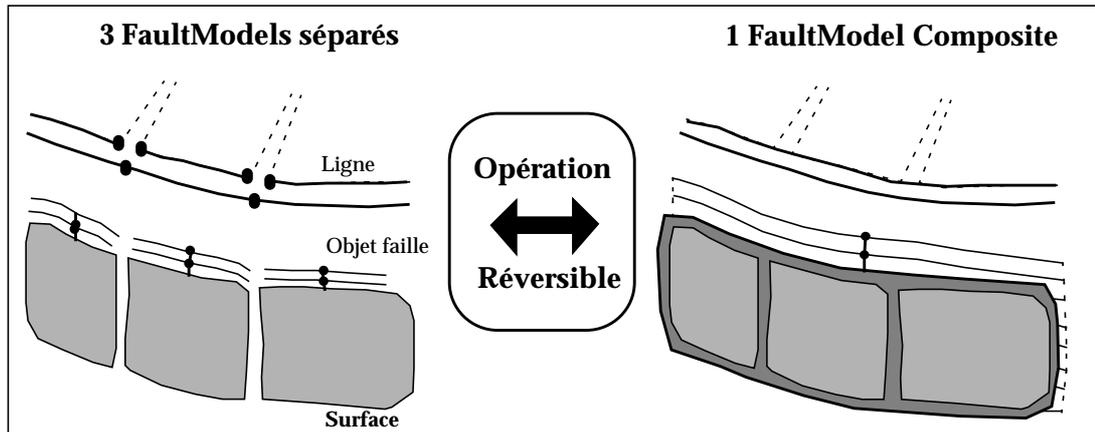


Figure 11-15 : Connecter plusieurs failles grâce au modèle "composite" (l'édition d'une des représentations a une action sur toutes les autres représentations).

### 2.3.3 Procédure proposée

- Détruire les failles de la zone en question

La première étape consiste à *sélectionner la zone* sur laquelle le travail de détail doit être effectué. Une fois la zone validée par l'utilisateur, tous les FAULTMODELS correspondants sont détruits s'ils sont composites. Leur destruction permet de retrouver les petits morceaux de faille à partir desquels ils ont été créés [figure 11-16].

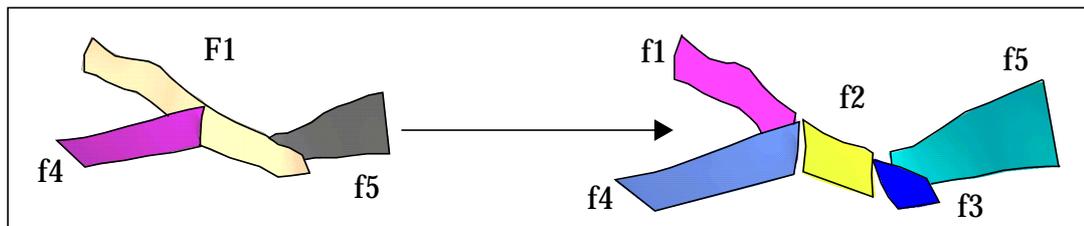


Figure 11-16 : Retrouver l'état initial de la zone à étudier

- Utiliser les liens de type «MERGELINK»

La seconde étape consiste à ajouter des liens d'assemblage en cliquant sur les morceaux de failles qui doivent être reliés en un seul morceau [figure 11-17].

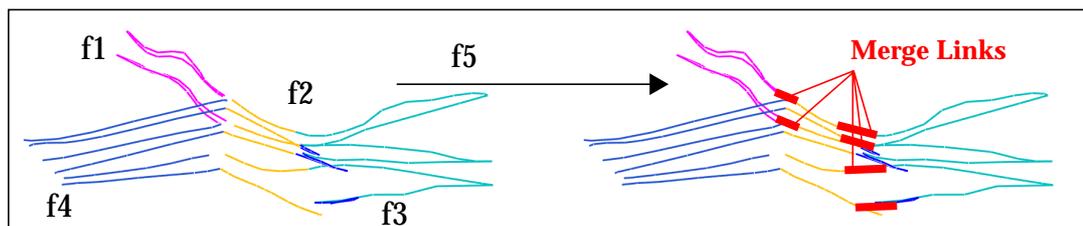


Figure 11-17 : Editer les liens dans la zone délicate

- Reconstruire la zone délicate

Il ne suffit plus, ensuite, que de reconstruire toutes les représentations des nouveaux FAULTMODELS” construits par l’opération précédente [figure 11-18].

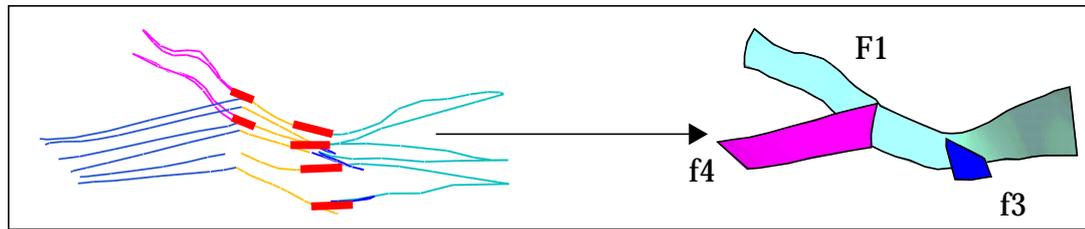


Figure 11-18 : Reconstruction de la partie de modèle éditée

- Vérifier le nouveau modèle

La dernière étape correspond à la vérification du modèle, par l’étude de coupes verticales, de diagrammes de rejets, ou en utilisant les règles décrites précédemment.

### 2.3.4 Problème des connections entre failles

Lorsque le branchement entre deux failles a été déterminé, il reste encore à définir la ligne d’intersection entre ces deux failles. Or dans les zones de branchement, les bords d’horizons construits à partir des données sismiques sont plus écartés que la largeur réelle des failles. Il faut donc utiliser des critères (plus ou moins objectifs) pour déterminer la direction d’extension de la faille secondaire vers la faille principale. Pour cela plusieurs choix sont possibles actuellement:

- **Point le plus proche** sur la faille principale
- **Direction des n derniers points** de la faille secondaire
- **Direction principale** de la **faille secondaire**
- **Direction principale** de la **famille de failles secondaires**
- **Ligne** déterminée par une autre méthode

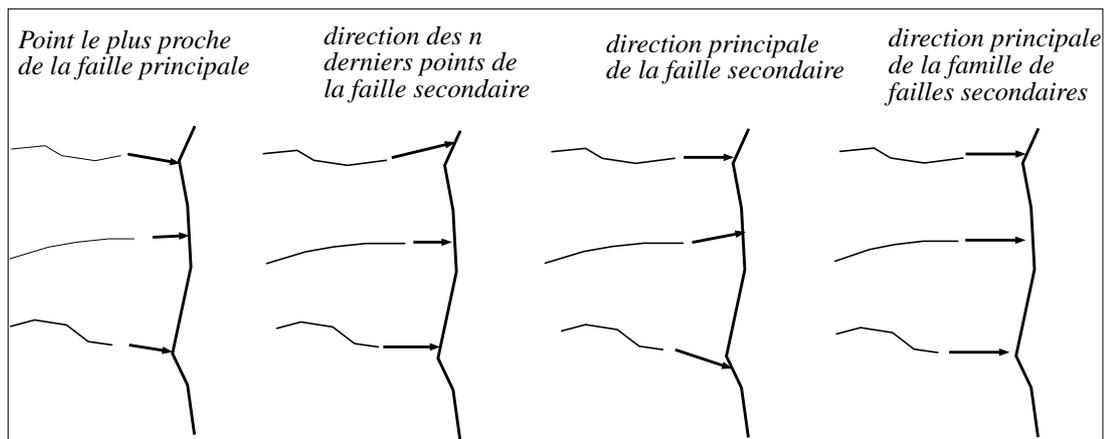


Figure 11-19 : Méthodes actuellement utilisées pour déterminer la direction de connection entre une faille secondaire et sa faille principale.

La section suivante montre comment des études sur la répartition de rejets le long des failles peuvent être utilisées pour déterminer la ligne d’intersection de manière plus géologique.

### 3. Apport de l'étude des rejets

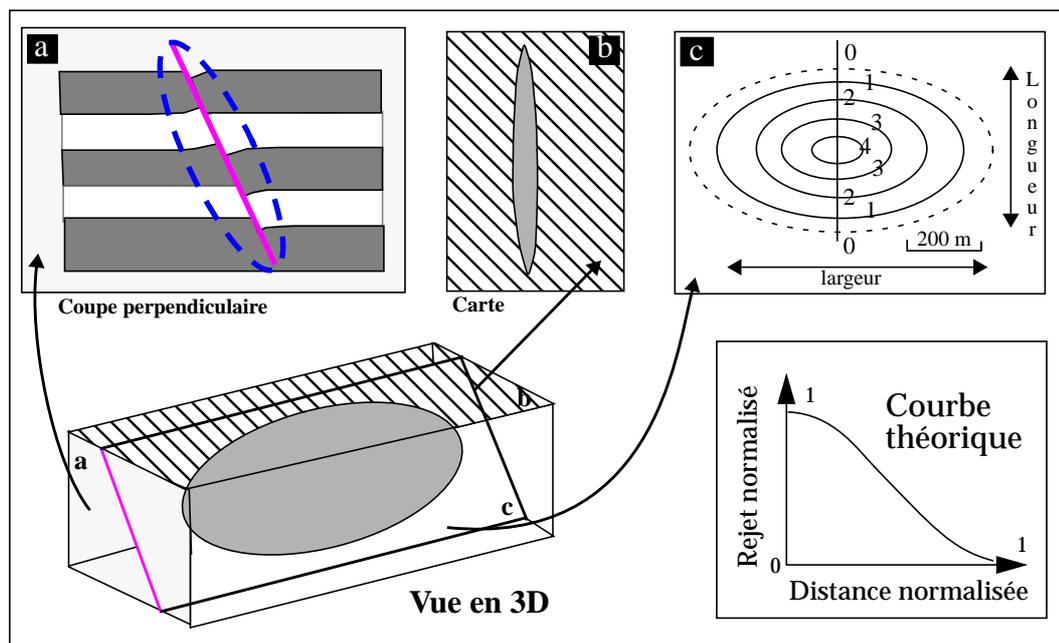
Les rejets des failles sont une propriété très étudiée dernièrement pour identifier les relations anomaliques dans les réseaux de failles, tester la qualité des interprétations, et extrapoler l'information dans les zones pauvres en données. Ces études ont commencé sur des observations de failles isolées dans les mines de charbon d'Angleterre [126 - Rippon], et ont été améliorées ensuite afin de prendre en compte des cas de plus en plus complexes.

#### 3.1 Simplification pour une faille normale isolée

Dès le milieu des années 80, Rippon a étudié les failles visibles dans les mines de charbon du centre de l'Angleterre [126 - Rippon]. Après avoir sélectionné des **failles isolées** dans des **zones de tectonique simple**, quelques règles générales sur la répartition des rejets le long d'une faille ont été ainsi déterminées. Ces travaux ont ensuite été repris par Walsh et Watterson [54 - Walsh 1] de l'université de Liverpool, puis par Barnett [119 - Barnett] sur un plus grand nombre de failles.

##### 3.1.1 Un peu de théorie

Une simplification généralement admise pour les failles normales isolées et ne se terminant pas sur une surface libre, est que les contours d'isovaleurs de rejets tracés sur le plan de faille sont des **ellipses concentriques**, avec le rejet maximum au centre de la faille [figure 11-20].



**Figure 11-20 : Différentes vues d'une surface de faille "elliptique". [a]: vue en coupe qui montre les couches entraînées par le mouvement de la faille; [b]: vue en carte; [c]: projection des rejets sur un plan vertical parallèle à la direction principale de la faille [119 - Barnett]. [d]: profil du rejet normalisé ([54 - Walsh 1])**

L'ellipticité de ces failles (axe 1 / axe 2) a été estimée en moyenne à 2 (avec des variations entre 1.25 et 3 sur le jeu de données des mines du Royaume Uni) [128 - Walsh 3]. Walsh et Watterson ont aussi montré que plus l'ellipticité augmente, moins les diagrammes sont symétriques.

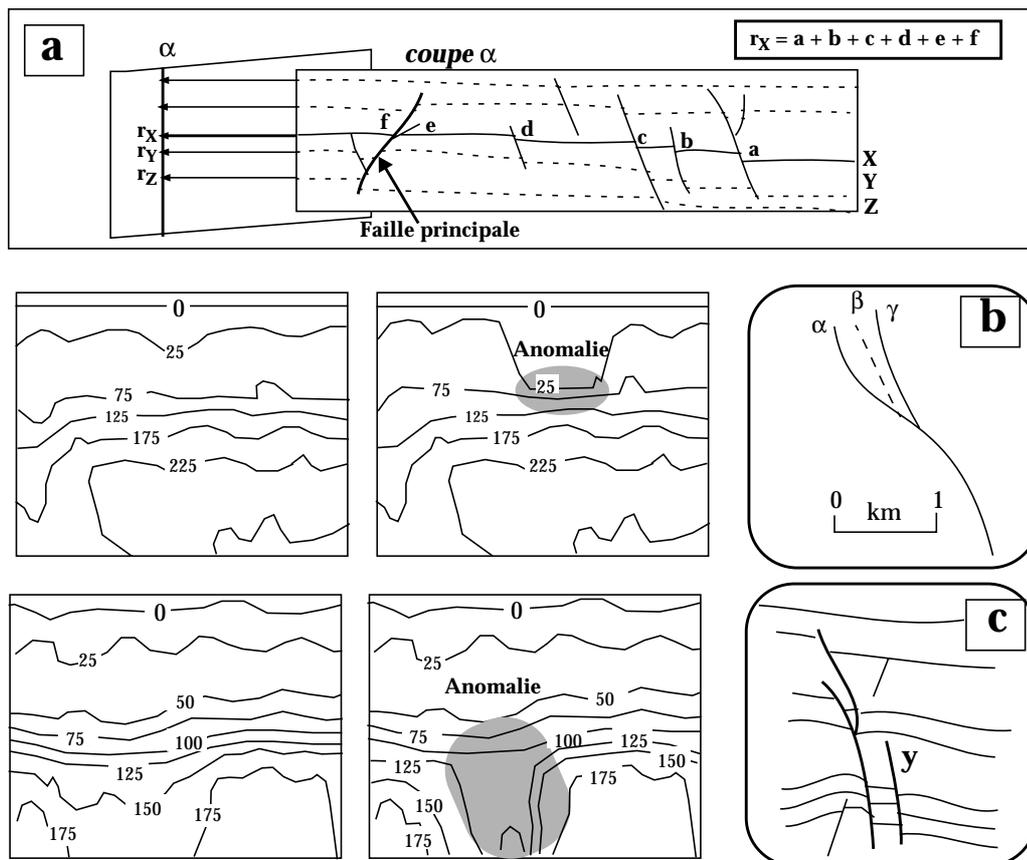
La courbe théorique donnant la valeur du rejet en fonction de la distance au centre de la faille [figure 11-20] (d) est donnée par la formule suivante [128 - Walsh 3] et assez bien respectée par le jeu de données étudié:

$$d = 2(1-r) \sqrt{\left(\frac{1+r}{2}\right)^2 - r^2} \quad \text{avec } r = \frac{\text{distance}}{R_{\max}} \text{ et } d = \frac{\text{rejet}}{D}$$

Selon cette théorie, une organisation régulière des courbes de rejet sur la surface de faille permet donc d'accorder une bonne confiance dans les données. Au contraire, trop d'irrégularités peuvent remettre en cause l'interprétation ou être dues à un manque de données.

### 3.1.2 Précautions à prendre lorsqu'il existe des failles secondaires

Cependant, les failles sont rarement isolées et les règles établies précédemment doivent être remaniées pour étudier les cas réels. Lors d'un événement tectonique, la faille principale est souvent accompagnée de failles secondaires. Or, entre la faille principale et les failles secondaires, il se produit des *interactions mécaniques*, même si ces failles n'ont aucune relation apparente à l'échelle de la carte ou de la coupe. Dans, ce cas, il faut **tenir compte de toutes les failles** correspondant à un même événement tectonique pour obtenir un diagramme régulier pour l'ensemble des failles [129 - Walsh 4], [figure 11-21].



**Figure 11-21 :** [a] Il faut tenir compte de toutes les failles secondaires pour avoir un diagramme régulier. [b]: le diagramme de gauche tient compte de la faille  $\beta$ , au contraire du diagramme de droite; [c]: le diagramme de gauche tient compte de la faille Y, au contraire du diagramme de droite (d'après [129 - Walsh 4]).

### 3.1.3 Technique employée sur un cas réel

Bien que ces théories soient contestables avec les données que nous traitons (les couches accommodant la déformation au cours du temps), et faute de mieux, le calcul de telles courbes d'isovaleurs de rejets a été testé afin de vérifier si la *théorie correspondait aux données* dont nous disposons.

La technique est très simple. Elle consiste à rééchantillonner les deux bords de chaque horizon avec le même nombre de points. Pour chaque couple (p,p'), la distance correspondante ("rejet") est calculée. Si la coupe (p,p') passe entre deux points de l'échantillonnage, une *interpolation linéaire* entre les couples (p1,p2) et (p'1,p'2) qui l'encadrent est calculée [figure 11-22] (a'). Cette valeur est ensuite projetée soit sur un plan vertical lorsque le pendage de la faille est suffisant, soit sur un plan horizontal lorsque le pendage de la faille est très faible, soit directement sur la surface de faille lorsque le pendage varie de long de celle-ci. Lorsque plusieurs failles sont rencontrées sur la même coupe, la valeur projetée correspond à la somme de tous les rejets calculés le long de la coupe [figure 11-22] (a).

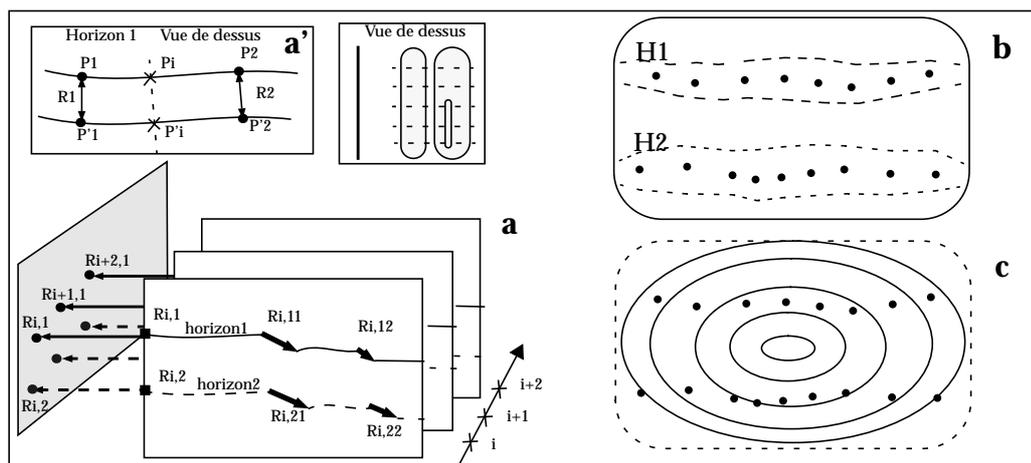
La valeur du rejet qui est calculé peut correspondre soit au rejet vertical, soit au rejet horizontal, soit au rejet total. Les résultats montrés par la suite correspondent à ce dernier calcul. La propriété rejet ainsi créée est ensuite interpolée sur toute la surface et les lignes de contours "iso-rejets" sont calculées [figure 11-22] (c).

#### *Exemple de code 11..1. Calcul du diagramme des rejets pour un cas simple*

```

1 Pour tous les horizons de h=1 à nb_horizons
2   Pour toutes les sections de i=1 à nb_sections
3     Pour toutes les failles de j=1 à nb_failles
4       Calculer le rejet  $rij = [(Pi-P1)*R2+(P2-Pi)*R1]/(P2-P1)$ 
5      $Ri = \sum rij$ 
6     Projeter  $Ri$  sur le plan
7   Interpoler la propriété rejet
8   Calculer les contours isovaleurs

```

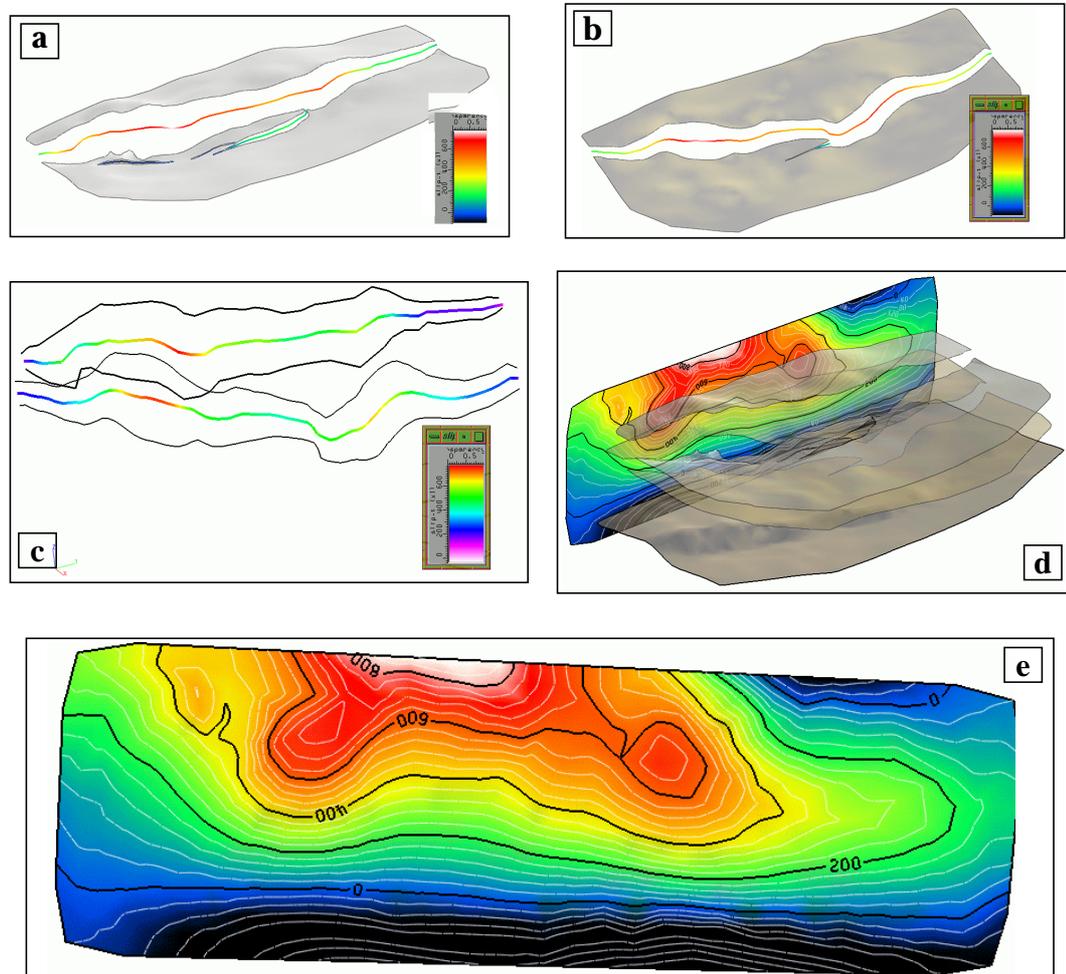


**Figure 11-22 : Technique employée pour calculer les rejets ("strike projection"):** [a] calcul du rejet pour un certain nombre de couples de points de part et d'autre de la faille; [b] projection des valeurs obtenues sur un plan; [c] interpolation et tracé de contours d'isovaleurs.

D'après certains auteurs, il faut que les rejets aient été calculés à partir d'au moins trois horizons pour que le calcul soit valide [119 - Barnett].

### 3.1.4 Résultats obtenus

Cet algorithme a été utilisé sur un cas réel comportant 3 horizons, une faille principale et deux failles secondaires. La faille principale s'amortit au niveau de l'horizon supérieur. Les rejets de la faille principale et des failles secondaires ont été sommés, puis projetés sur un plan vertical. Enfin, les valeurs projetées ont été interpolées avec l'interpolateur D.S.I pour donner le résultat de la figure ci-dessous [figure 11-23].



**Figure 11-23 : Calcul du rejet dans un cas simple.** [a]: calcul des rejets des différentes failles le long de l'horizon supérieur; [b]: calcul des rejets pour chacune des failles le long de l'horizon inférieur; [c] le rejet des failles secondaires est incorporé dans le rejet total projeté sur le plan; [d] interpolation des valeurs de rejet ainsi obtenues; [e]: carte finale des rejets.

Le résultat donne globalement des demi-ellipses concentriques avec un maximum au milieu de l'axe longitudinal de la faille. On observe bien que les petites failles secondaires donnent une forme plus irrégulière sur les contours de rejets. Le fait que l'on n'obtienne que des demi ellipses peut s'expliquer par le fait que les couches accommodent la déformation au cours du temps. Le fait que la faille s'amortisse au niveau de l'horizon supérieur peut aussi avoir une influence: elle ne correspond pas à l'une des hypothèse de base (la faille ne doit pas s'interrompre sur une surface libre).

Ne disposant d'aucune information sur la lithologie ou l'histoire de cette faille, l'analyse du résultat n'a pas été poursuivie. Cependant, le diagramme obtenu correspond assez bien à la théorie proposée.

### 3.2 Amélioration de cette théorie dans le cas d'une tectonique polyphasée

Comme Rippon l'a souligné [126 - Rippon], la théorie est difficile à généraliser pour plusieurs raisons: des lithologies variables le long de la faille, une interaction avec d'autres failles, une tectonique polyphasée dans laquelle les mouvements le long de la faille ont été successivement normaux puis inverses...

#### 3.2.1 Comparaison entre l'état stable et l'état après un séisme unique

Walsh et Watterson ont étudié les différences entre le diagramme obtenu après une série d'événements tectoniques et celui obtenu après un épisode sismique isolé [127 - Walsh 2].

A chaque nouvel événement sismique, la distribution des rejets conserve la même forme, mais du fait de la croissance simultanée de la faille, les incréments de rejets en un point donné ne sont pas identiques. L'état stable serait atteint après une cinquantaine de séismes sur les failles étudiées dans cet article.

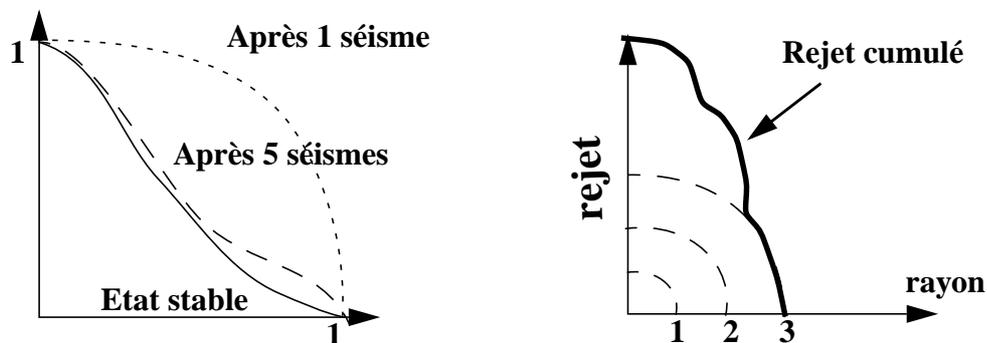


Figure 11-24 : Evolution du diagramme des rejets au cours des séismes successifs (d'après [127 - Walsh 2]).

Lorsqu'on étudie les diagrammes de rejets, il faut donc faire attention au **facteur temps** qui peut déformer les diagrammes obtenus, ainsi qu'aux propriétés mécaniques des roches qui peuvent être modifiées au cours du temps.

#### 3.2.2 Changements de connections au cours du temps

Chris Morley s'est penché, lui, sur le problème plus particulier de l'évolution de ces contours au cours du temps, lorsque des failles grandissent ou se regroupent [125 - Morley]. Cette étude s'est effectuée sur des failles de très grande échelle (failles de quelques centaines de kilomètres de long).

A partir d'une étude de sections sismiques concernant six grandes failles d'un bassin de l'Afrique de l'est (Lac Tanganyika), trois grands types de propagation ont pu être définis [figure 11-25]:

- **Diagramme simple (A)**: avec le rejet maximum au centre de la faille et décroissant graduellement jusqu'aux extrémités (*i.e.* théorie précédente).
- **Propagation asymétrique (B)**: où l'une des extrémités de la faille reste inactive tandis que l'autre extrémité se propage parfois sur de longues distances.
- **Rejet variable (C)**: ceci étant du en partie à l'amalgame de deux segments de faille.

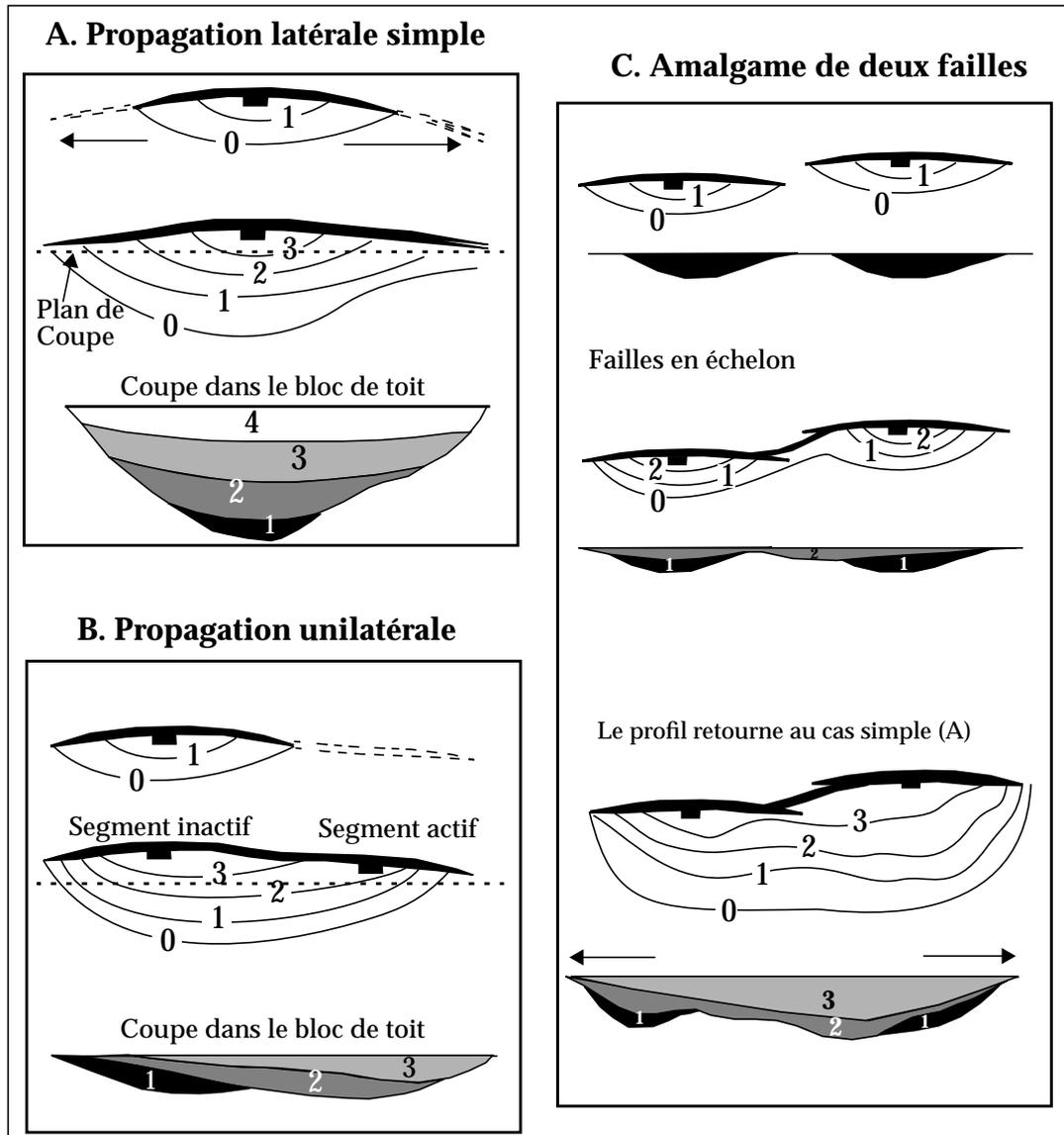


Figure 11-25 : Différentes évolutions possibles d'une faille au cours du temps (d'après [125 - Morley]).

Ces études montrent bien que le profil théorique établi par Rippon, Walsh et Waterson ne correspond pas à tous les cas rencontrés dans la nature et qu'elle ne tient pas compte d'un facteur très important: **le temps**.

Cependant, Morley souligne que les *échelles d'étude* sont très différentes entre les deux modèles et que ce qui est établi sur les grandes failles (échelle du bassin) ne correspond pas forcément à l'évolutions des failles plus petites (échelle du réservoir).

Ce domaine d'étude est encore *récent et très ouvert*. Il me semble important de suivre l'évolution des théories dans ce domaine et de proposer des outils simples correspondant aux modèles ainsi définis. Ceci pourrait aider les structuralistes à mieux comprendre et contrôler l'organisation du réseau de faille, et ainsi limiter les incertitudes liées à l'interprétation.

### 3.3 Modification de la théorie dans les zones d'interaction entre failles

La théorie présentée en début de section n'est valable que dans le cas d'une faille isolée et n'intersectant pas la surface. Ceci est rarement le cas pour les données que nous avons à étudier. Plusieurs failles peuvent interagir de différentes manières: soit lorsque des zones de relais entraînent des interactions mécaniques entre différents segments, soit lorsqu'une faille secondaire se branche sur une faille principale. Ces phénomènes sont beaucoup étudiés actuellement, afin d'améliorer les interprétations.

#### 3.3.1 Les zones de relais

Ayant remarqué que de nombreuses failles sont en fait discontinues et composées d'un grand nombre de segments distincts (non seulement en carte, mais aussi en coupe), Willemse a étudié plus précisément l'influence que plusieurs segments en échelon pouvaient avoir les uns sur les autres [130 - Willemse]. En effet, la géométrie est de plus en plus reconnue comme l'une des propriétés fondamentales des failles [figure 11-26].

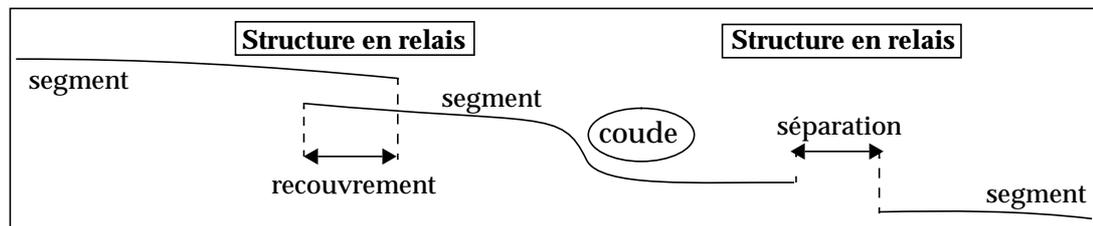


Figure 11-26 : Différentes relations possibles entre les segments d'une même faille (d'après [130 - Willemse]).

Bien que globalement semblables, les contours de rejets sont beaucoup plus perturbés le long de ces failles segmentées que dans le cas simple d'une faille isolée. Des asymétries apparaissent notamment au niveau des relais, du fait de l'interaction mécanique entre les différents segments [figure 11-27].

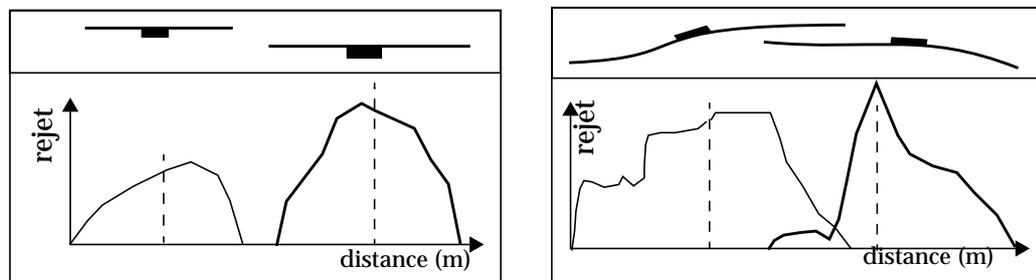


Figure 11-27 : Comparaison des rejets de failles isolées [a] et de failles en interaction mécanique [b] (d'après [130 - Willemse]).

En comparant des observations sur le terrain, et les résultats d'un logiciel d'analyse numérique (*Poly3d*), il a pu tirer quelques conclusions sur l'influence des zones de recouvrements entre deux segments d'une même faille, non seulement sur la position du rejet maximum, mais aussi sur la répartition des contours de rejets le long de la faille. Il a ainsi pu établir les règles suivantes:

1. les distributions de rejets sont invariablement moins régulières le long des failles segmentées que le long des failles isolées et continues
2. le long de chaque segment, le rejet varie de zéro aux extrémités à un ou plusieurs maxima, situés la plupart du temps plus proches de la zone de relais
3. les zones de relais sont caractérisées par une diminution des rejets
4. ces changements semblent liés à l'espacement entre les différents segments qui composent la faille.

Ces observations ont ensuite été confirmées sur un modèle à trois segments de failles, ainsi que sur des exemples réels. Une fois encore, les courbes d'un ensemble de segments de failles ressemblent fort aux contours d'une faille isolée, avec des perturbations notables dans les zones de relais [figure 11-28]

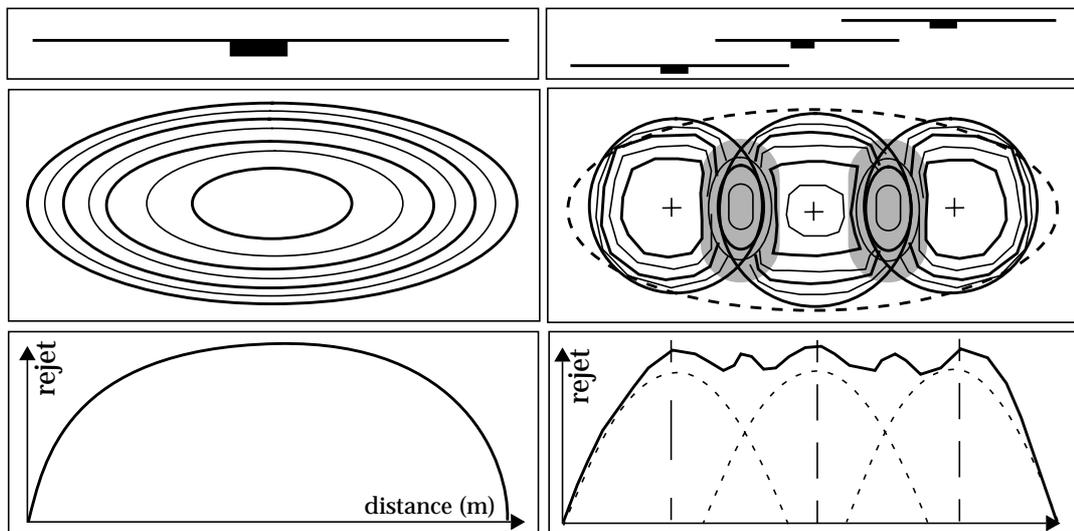


Figure 11-28 : Comparaison entre les caractéristiques d'une faille elliptique isolée (gauche) et celle d'un ensemble de trois segments de faille (droite) avec des zones de recouvrement (grisé), d'après [130 - Willemsse].

### 3.3.2 Les zones de branchement

De la même manière, lorsque plusieurs éléments d'un même système de faille s'intersectent, les failles branchées à une faille principale ou simplement localisées dans la même zone peuvent interagir mécaniquement avec celle-ci et influencer les contours de rejets correspondants. Ainsi, dans de tels cas, il n'est pas rare que la faille principale présente *plusieurs rejets "maximum"* le long de sa surface: un maximum de chaque côté de la ligne de branchement.

L'équipe "Rock Fracture Project" de l'université de Stanford a réalisé une analyse de telles relations entre deux failles ayant chacune un pendage de  $60^\circ$ , et l'angle entre les deux failles variant de  $30^\circ$  à  $180^\circ$  (cas référence = faille plane) ([124 - Maerten 2].

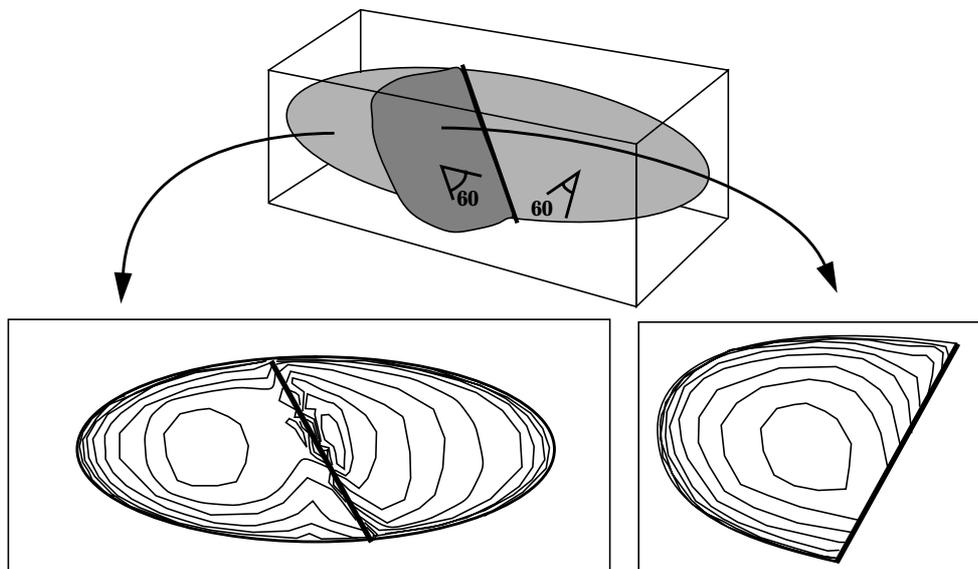


Figure 11-29 : Influence du branchement sur les rejets (D'après [124 - Maerten 2])

Dans tous les cas, les contours de rejets le long de la faille principale présentent un *maximum de part et d'autre de la ligne d'intersection*, ainsi qu'une *forte asymétrie*. Le rejet dans la moitié correspondant au toit de la faille secondaire est supérieur à celui d'une faille planaire isolée de taille équivalente, et le rejet correspondant au mur de la faille secondaire est inférieur à celui d'une faille planaire isolée de taille équivalente.

Le rejet de la faille secondaire est beaucoup plus complexe à analyser, du fait qu'elle est tronquée par la faille sur laquelle elle se branche. Le *rejet maximum* est cependant très *proche de la ligne de contact* dans tous les cas de figure. Les contours de rejets présentent en outre une très *forte asymétrie*.

Cette étude a aussi montré que l'angle entre les deux failles a un effet important sur la forme des lignes de contours.

Enfin, bien que les conditions aux limites de ces modèles soient très simplifiées, les rejets calculés sont très proches de ce qui peut être observé sur la sismique ou sur des modèles analogiques.

Ainsi, ces modifications de contours de rejets peuvent être utilisées pour déterminer l'**endroit exact où deux failles s'intersectent**, et estimer plus précisément l'étendue verticale de la ligne d'intersection. En effet, ces zones sont précisément les plus complexes à analyser sur les données sismiques qui sont souvent brouillées dans les zones de branchement.

L'équipe "*Rock Fracture Project*" de l'université de Stanford, propose de comparer les contours de rejets calculés à partir des données sismiques et le résultats de calculs d'ordre mécanique (logiciel *Poly3d*) pour tester différentes hypothèses [123 - Maerten 1]. Le modèle mécanique dont les contours se rapprochent le plus de ceux calculés à partir des données sismiques est sélectionné.

Cette technique a été appliquée sur des données d'un champ de la mer du Nord. Pour deux des failles (A et B sur la figure), l'analyse de la sismique avait conduit à une interprétation sans contact. Cependant, l'analyse du rejet de la faille A présentant toutes les caractéristiques d'un contact, une comparaison avec deux modèles numériques (le premier sans contact, le deuxième avec contact) a permis de modifier l'interprétation [figure 11-30].

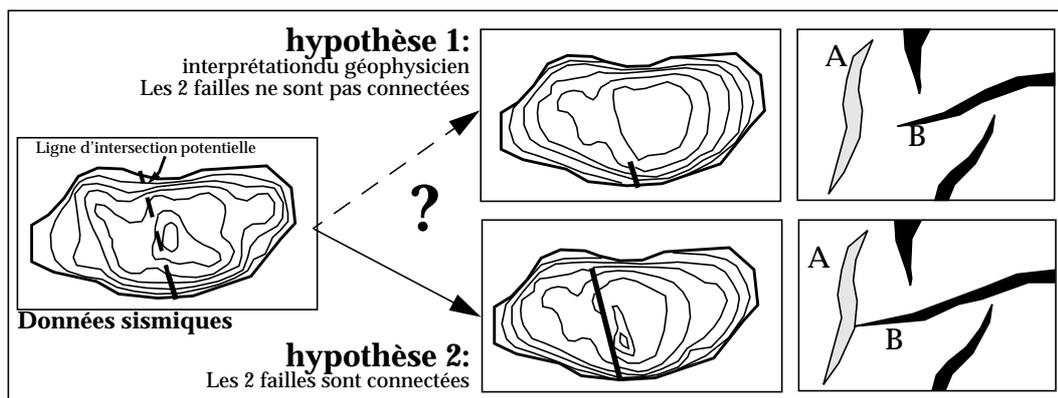


Figure 11-30 : L'étude des rejets peut aider à améliorer l'interprétation du réseau de failles (simplifié d'après [123 - Maerten 1]).

On comprend qu'utiliser de telles méthodes peut aider à réduire considérablement les incertitudes dues à l'interprétation des données sismiques dans les zones complexes.

### 3.3.3 Résultats obtenus avec gOcad sur des données réelles

Sur le jeu de données ci-dessous, la technique précédente a été appliquée afin de vérifier que les intersections entre la grande faille principale et les nombreuses failles secondaires avaient des répercussions sur les contours de rejets.

Sur ce jeu de données, trois horizons, des polygones de failles et des interprétations de failles (courbes) étaient disponibles. Après avoir construit les différentes failles et fermé le modèle, les rejets de la faille principale ont été calculés à partir des trois horizons interpolés. Ces valeurs, projetées sur la surface de faille ont ensuite été interpolées, puis les contours de rejets calculés [figure 11-31].

On observe bien que:

- les contours forment des ***courbes concentriques***, avec de nombreuses *perturbations*, surtout au niveau des intersections avec les failles secondaires
- ***plusieurs maxima*** sont observables, de part et d'autre de chacune des lignes d'intersection
- ***le plus grand maximum n'est pas au centre*** de la faille

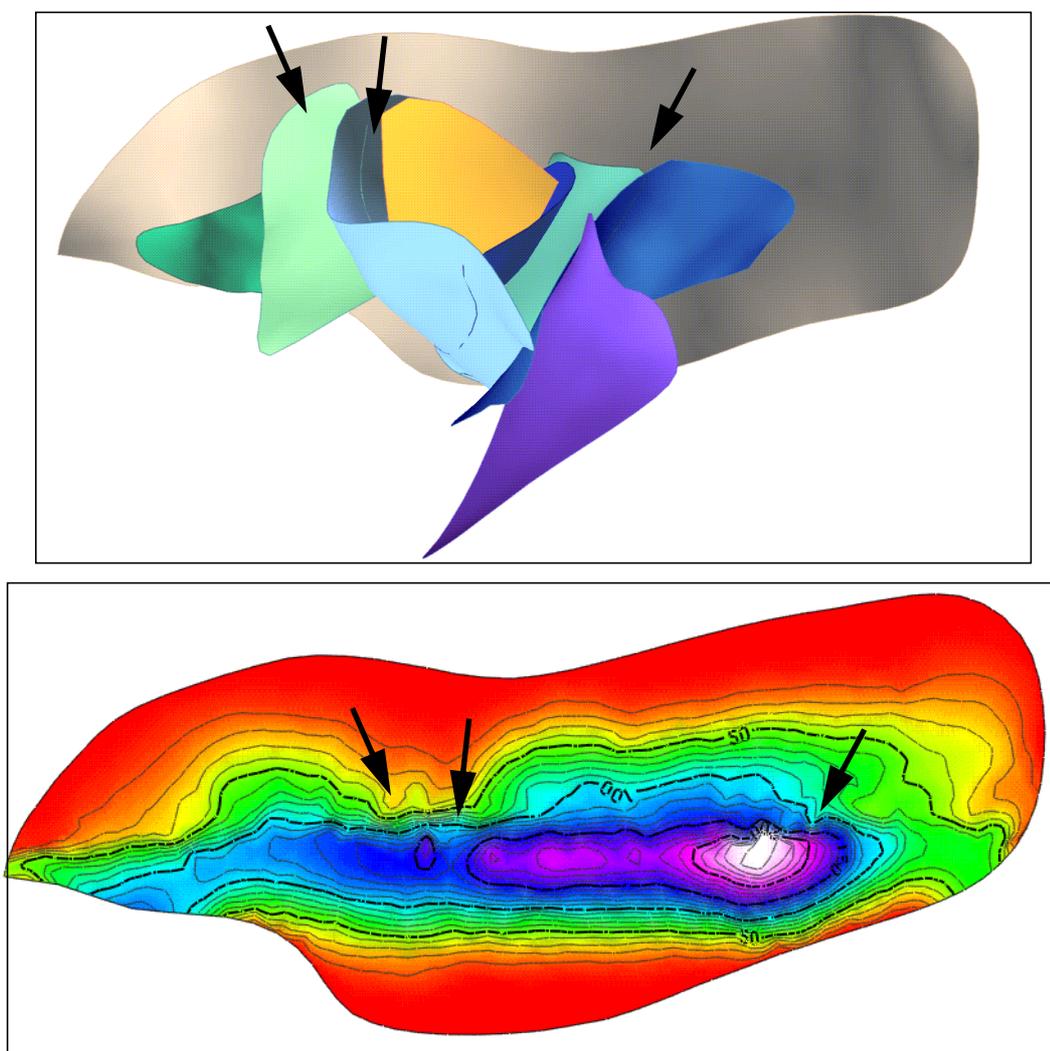


Figure 11-31 : L'analyse des rejets permet de retrouver les principaux contacts de la [figure 11-30] (Modèle initial construit par Christophe Basire [64 - Basire])

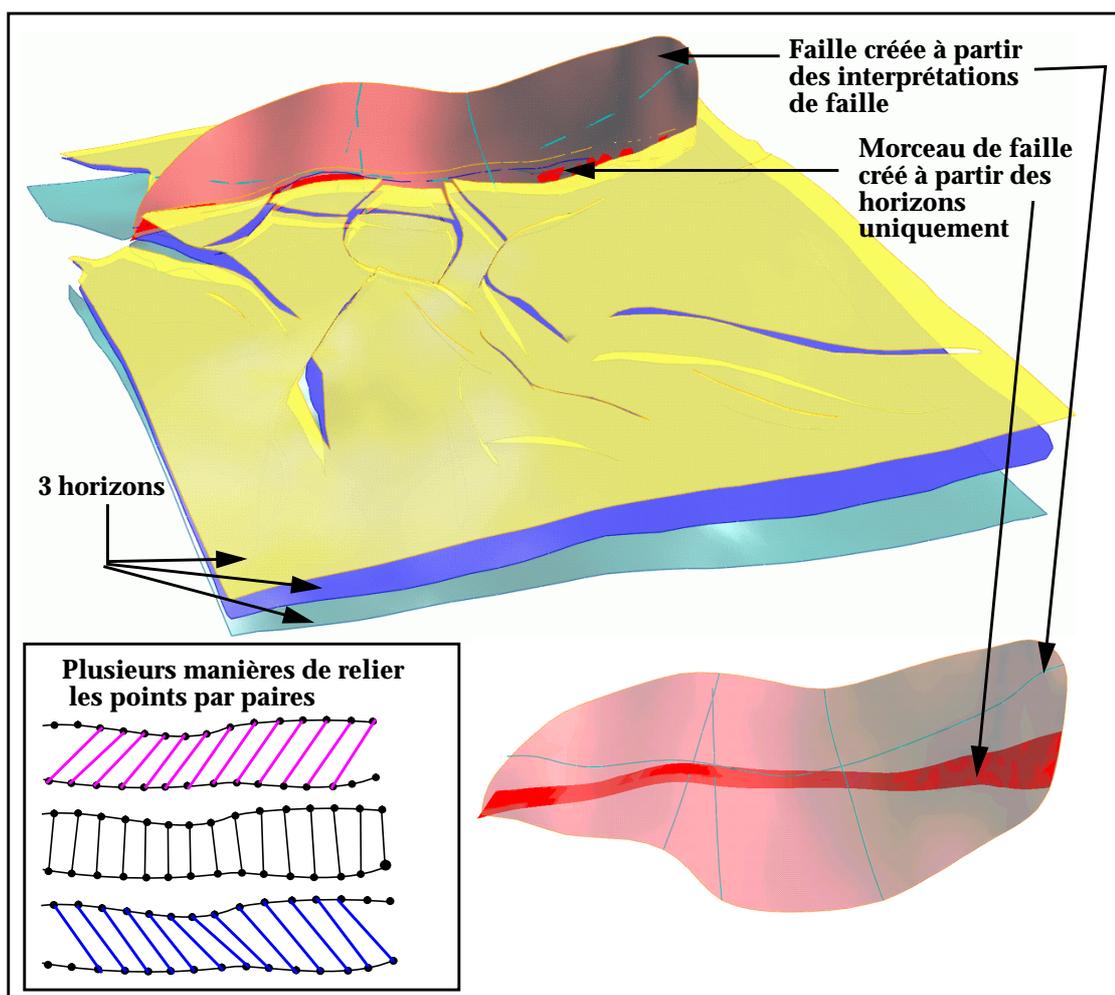
L'étude des rejets permet donc non seulement de vérifier une interprétation, mais aussi peut éventuellement aider à améliorer les contacts entre failles (détection de nouveaux contacts, meilleure position du contact).

### 3.3.4 Amélioration possible de l'utilisation de cette technique dans gOcad

Si le résultat obtenu semble bien correspondre à la théorie, il faut encore améliorer la technique du calcul du rejet, pour éviter les biais.

En effet, dans la plupart des cas, les données auxquelles nous avons affaire ne permettent pas de modéliser la faille dans son entier. Sans information supplémentaire, il est difficile de décider comment réunir les paires de points, de part et d'autre de la faille.

Si ce défaut est relativement peu important pour calculer le rejet sur une faille isolée (à moins qu'elle ne soit à dominante décrochante, le rejet peut être assimilé au rejet vertical), il risque d'avoir une plus grande importance dans les zones plus complexes.



**Figure 11-32 : Lorsque l'on a uniquement la description de la faille au niveau des horizons, comment relier les nœuds des bords d'horizon correspondant aux deux lèvres de la failles?**

Sur la sismique, il est extrêmement difficile d'avoir une idée de la composante horizontale du rejet des failles. Il faut donc utiliser d'autres critères afin de déterminer la direction du rejet.

Lorsque les analyses de carottes le permettent, cette direction peut être déterminée par les indices laissés sur les couches lors du mouvement imposé par la faille (crochons de failles par exemple).

Dans le cas contraire, certaines équipes proposent des techniques pour déterminer les directions du mouvement d'un bloc de faille par rapport à l'autre.

En effet, la surface de faille n'est en fait que la représentation de la limite entre deux blocs supposés rigides qui ont glissé l'un contre l'autre. Même si cette hypothèse de bloc rigide n'est pas respectée totalement, on comprend aisément que la surface de faille obéisse à certaines règles mécaniques pour permettre aux deux blocs de glisser l'un par rapport à l'autre [98 - Thibaut]. Dans sa thèse, Murielle Thibault propose un calcul qui utilise une dérivée seconde, (donc potentiellement instable) pour déterminer le mouvement occasionné par la faille.

Mallet propose une autre technique, qui prend en compte la *courbure* de la faille. En analysant les courbures sur une faille issue d'une tectonique monophasée, on retrouve relativement facilement le sens du mouvement entre les deux blocs. Si la tectonique est polyphasée, cette méthode est plus contestable. En effet, comme Chapman le montre sur une faille normale ayant eu une activité complexe au cours du temps (alternance de phases actives et inactives, inversion du rejet), il est extrêmement difficile de retrouver les composantes du mouvement uniquement à partir de la géométrie actuelle [120 - Chapman].

Cette méthode n'en est qu'à ses balbutiements, mais peut aider dans certains cas, où aucune indication supplémentaire n'est donnée, à calculer le rejet de la faille en incorporant une notion de direction du mouvement.

De plus, des méthodes de dépliage peuvent permettre de mieux comprendre l'histoire de la faille, et ainsi d'affiner les résultats.

Toutes les techniques présentées dans ce chapitre pourraient donc faciliter la description de plusieurs scénarios plausibles pour le réseau de faille, et constituer un autre volet de l'étude des incertitudes sur les failles, à modèle de vitesses constant.

Pierre Thore, lui, se propose de travailler plus particulièrement sur l'incertitude due au modèle de vitesse utilisé pour transformer les interprétations des sections sismiques du domaine temps vers le domaine profondeur [70 - Thore 1].

De manière idéale, il faudrait combiner ce qui a été présenté dans ce mémoire (calcul de plusieurs réalisations autour d'un scénario donné) et ces différents scénarios pour étudier de manière plus complète les incertitudes sur le réseau de faille.



---

---

# Conclusion

---

---

Du fait qu'ils sont situés à grande profondeur, les compagnies pétrolières étudient les réservoirs potentiels à l'aide de *données indirectes* la plupart du temps, et qui contiennent souvent une grande part d'*incertitude*. Les champs étudiés étant de plus en plus profonds et complexes, le risque pris lors des décisions d'implantation de nouveaux puits devient très important, se soldant le plus souvent par un échec (puits sec).

C'est pourquoi, de plus en plus, on passe de la création d'un modèle unique (déjà difficile à obtenir) à la création d'une ***multitude de modèles équiprobables***, permettant de mieux analyser les risques lors de l'exploitation d'un champ.

Afin de balayer toutes les possibilités, des études statistiques sont réalisées à tous les niveaux : étude de la répartition des faciès, de certaines propriétés caractéristiques des couches (porosité, perméabilité), de la circulation des hydrocarbures dans le réservoir, des impacts économiques, et plus récemment, de la géométrie du réservoir.

## 1. Bilan de la thèse

Le travail présenté dans ce mémoire s'inscrit dans la dernière catégorie (*géométrie du réservoir*), et concerne plus particulièrement les ***incertitudes concernant la géométrie des failles***. Ce type d'incertitudes a encore été peu étudié, du fait de la complexité de certains réseaux de failles. Elle se décompose en deux parties : la définition de plusieurs réseaux de failles possibles (*scénarios*), et la génération de petites variations de la géométrie des failles (*réalisations*) autour d'un scénario donné. C'est sur cette dernière partie que nous avons travaillé.

Les principales difficultés rencontrées concernent :

- **l'estimation des incertitudes** : il est déjà difficile de déterminer une géométrie possible pour la faille, les incertitudes sur cette géométrie sont encore plus difficiles à déterminer. Ne disposant pas des données permettant de les définir (sections sismiques), nous considérons qu'elles doivent faire ***partie intégrante de la phase d'interprétation*** de ces sections (ce qui n'est pas encore rentré dans les mœurs...)
- la **prise en compte de ces incertitudes lors de la modélisation**. Beaucoup plus que pour celle des horizons, la géométrie des failles obéit à des règles mécaniques, et toutes les simulations de géométries ne sont pas forcément correctes du point de vue géologique. Malheureusement, les critères permettant d'éliminer les solutions non géologiques sont plus qualitatifs que quantitatifs, et même si de nombreuses études sur des modèles synthétiques sont proposées, elles ne sont pas forcément applicables aux données dont nous disposons, du moins actuellement. C'est pourquoi nous avons considéré que des failles correctes étaient fournies au départ, et que nous nous sommes appliqués à respecter au maximum ces ***géométries de référence*** au cours des simulations.

- le **respect du modèle de référence** : comme cela a été dit à de nombreuses reprises, s'il est relativement aisé de modifier la géométrie d'une faille isolée, le problème se complique lorsqu'il faut en plus mettre à jour un modèle complet (comprenant des horizons et des failles) après chaque simulation. La **gestion des contacts** est sûrement le problème sur lequel nous avons passé le plus de temps, et explique à elle seule pourquoi les études d'incertitudes concernant les failles ont été longtemps laissées de côté.
- **l'utilisation des modèles créés** : enfin, ces simulations doivent servir à faire des analyses (répartition des volumes par exemple) ou pouvoir s'insérer dans une étude plus globale des incertitudes. Les deux aspects ont été abordés, et utilisent une base commune : après un cycle de simulation, il peut être décidé soit de construire un *modèle volumique* et de calculer le **volume** des différentes régions, soit de construire une *grille faillée* permettant d'étudier **d'autres types d'incertitudes**. Même si la deuxième partie n'est qu'entamée, elle pourra bénéficier des structures mises en place pour répondre aux objectifs de la première partie.

Devant la quantité de choses à réaliser, il a fallu faire des choix. Le but que je me suis fixé a été **de mettre au point les structures permettant de balayer l'ensemble des problèmes**, quitte à ce que certains de ces modules ne soient traités qu'au moyen de solutions simples, voire ne soient qu'effleurés. L'ensemble de ce travail est **modulaire**, ce qui permet de se pencher plus particulièrement sur l'un des points pour l'améliorer, sans remettre en cause le reste du travail.

Les différents modules sont au nombre de six :

- la **modélisation des failles** et de leurs incertitudes
- les **méthodes de simulations** s'appliquant sur une faille isolée
- l'utilisation de ces méthodes pour **simuler un réseau de failles**
- la **mise à jour du modèle structural** (surfaccique)
- l'utilisation de ce modèle surfaccique pour faire une **étude volumique**
- **l'intégration** de ces méthodes dans une étude plus complète des incertitudes (construction/mise à jour d'une grille faillée)

Parmi ces modules, deux n'ont été qu'effleurés : la simulation d'un réseau de failles en tenant compte de la *hiérarchie*, et la construction de *grilles faillées*.

Les algorithmes utilisés dans les différents modules, empruntent un maximum de fonctions existantes du logiciel **gOcad**, avec lequel ce travail a été réalisé. L'avantage de travailler avec un logiciel existant depuis 10 ans est que la plupart des briques de base sont écrites. Ainsi, à quelques exceptions près, la majeure partie du travail a consisté à **mettre en place des structures mettant bout à bout des fonctions existantes** pour réaliser les objectifs de chacun des modules précédents. Parmi ces structures, on peut retenir:

- un **objet faille** dont la paramétrisation est entièrement orientée pour tenir compte des *estimations d'incertitude*. Cet objet, en bénéficiant des fonctionnalités des lignes polygonales dont il dérive, permet de modéliser rapidement une faille à partir des données disponibles pour la décrire. Sa paramétrisation mimant une grille est non seulement **flexible**, mais autorise en plus le stockage de **vecteurs d'incertitude variables** tout au long de la faille, et d'interpoler des valeurs d'incertitude si leur estimation n'est fournie qu'en quelques points. De plus, cette structure est **allégée** au maximum afin de favoriser une simulation la plus rapide possible de la géométrie.

- un **gestionnaire de simulations** qui propose différentes manières de modifier la géométrie d'une faille isolée. Ces méthodes sont toutes bâties sur le même principe : la définition d'une **distribution de probabilité locale** en chacun des points de la faille, et l'utilisation d'une **fonction aléatoire corrélée dans l'espace** permettant de simuler une nouvelle position pour chacun des points, tout en corrélant ces simulations le long de la faille. Cette méthode est plus connue sous le nom de **simulation par champ de probabilités** (ou *PField* en anglais). Les fonctions aléatoires que nous proposons autorisent des corrélations globales sur la faille, selon l'une des directions principales ou selon les deux directions principales. Quels que soient les choix de l'utilisateur, cette structure permet d'associer à chaque point de l'objet faille un **vecteur de translation** par rapport à la géométrie de référence. La définition de translations plutôt qu'une réelle modification de la géométrie permet de **conserver intacte la faille de référence**, et d'utiliser d'autres objets pour représenter les différentes réalisations.
- un **gestionnaire de contacts pour le modèle structural** : cette structure est celle qui a demandé le plus de temps et d'efforts, et est essentielle pour la suite des opérations. Elle peut être utilisée sans qu'aucune étude d'incertitudes ne soit réalisée, et résulte d'une collaboration étroite avec *Jean-Claude Dulac* (pour l'organisation des différents éléments composant cette structure) et *Isabelle Duvinage* (qui a écrit la plupart des algorithmes) [103 - Duvinage 3]. L'objet qui en résulte permet de stocker de manière efficace toutes les relations entre les différents éléments du modèle, ainsi qu'une gestion plus globale des opérations, favorisant **l'automatisation** de la mise à jour du modèle surfacique. Ce dernier point est fondamental pour la simulation d'un modèle entier, puisqu'on ne peut imaginer demander à l'utilisateur d'intervenir après chaque simulation. Il faut que la **mise à jour soit complètement automatique**. Ce point comporte certains risques : du fait des imprécisions numériques, la mise à jour du modèle ne se déroule pas toujours bien, et constitue le **facteur limitant** de l'ensemble de ce travail. L'une des clés de cette mise à jour est la possibilité de partager une ligne de contact entre plusieurs surfaces, afin d'assurer un **contact parfait**.
- un **simulateur permettant de visualiser l'évolution des volumes**: cette structure utilise toutes les structures précédentes, et les met bout à bout afin de réaliser un grand nombre de simulations de manière automatique. Elle propose une interface, que nous espérons conviviale, et qui permet d'avoir une vue d'ensemble sur les variations de volumes dans les différentes régions du modèle.

Actuellement, il manque encore deux structures principales pour que l'étude des incertitudes concernant les failles (à *scénario constant*) soit complète:

- un **gestionnaire de hiérarchie**: il permettrait de simuler les failles, non plus de manière indépendante, mais en tenant compte de la hiérarchie. L'une des principales difficultés de cette partie est de **définir la hiérarchie**. En supposant qu'elle peut être fournie, et en utilisant la gestion des contacts de l'objet *StructuralModel*, il suffit de peu de choses pour la stocker et l'utiliser durant les simulations.
- un **gestionnaire de grilles faillées**: beaucoup de structures existantes peuvent être réutilisées pour pouvoir mettre à jour une grille faillée. De plus, cette partie doit être intégrée au logiciel *Jacta*, qui fonctionne déjà avec de telles grilles (failles rectilignes selon la direction z). Les structures déjà mises en place dans ce logiciel pour mettre à jour la grille après une simulation de la géométrie des horizons sont peut-être suffisantes pour que l'intégration des deux techniques se fasse rapidement. Je n'ai pas encore eu le temps de regarder précisément le fonctionnement de ce logiciel, et cette hypothèse reste à vérifier.

## 2. Autres études d'incertitudes liées aux failles

Dans le dernier chapitre de ce mémoire, il a été souligné que l'étude des incertitudes autour d'un scénario donné ne permet pas de faire une étude complète des incertitudes concernant la géométrie des failles, et qu'il serait bien de pouvoir en outre aider les structuralistes à construire **plusieurs scénarios**. Lors de la création de l'objet STRUCTURALMODEL certaines structures ont été décrites afin de pouvoir un jour travailler dans cette direction (édition des liens).

Mais il faut ajouter que les failles *ne se limitent pas à leur géométrie*, et que leurs propriétés de **transmissivité** constituent une part importante de l'incertitude entourant le réservoir. En fonction des couches dans lesquelles elles se sont formées, de leurs dimensions, de leur pendage, et des terrains qu'elles mettent en contact, elles constituent soit une **barrière** pour les fluides, soit un **couloir** par lequel ils peuvent s'échapper.

De nombreuses études, se plaçant souvent à **plus petite échelle**, et utilisant des données différentes (données sismiques à petite échelle, données de puits précises, simulations de failles sub-sismiques), sont réalisées actuellement pour estimer les paramètres qui caractérisent la transmissivité des failles [136 - Knipe 1], [137 - Knipe 2], [144 - Yielding 3].

Différents mécanismes conduisant à la formation de pièges contrôlés par les failles ont été répertoriés. Tous ces processus conduisent à une **forte réduction de la porosité** (volume des vides) **et de la perméabilité** (volume de vide accessible aux fluides) au niveau de la faille:

- La **juxtaposition** de couches poreuses (réservoir) avec des unités de plus faible porosité
- L'entraînement **d'argiles** dans l'épaisseur de la faille (plus connu sous le terme anglais *Smear*)
- Le **broyage de grains de sable** lors de l'entraînement des couches pendant la création de la faille (plus connu sous le terme anglais *Fault Gouge*)
- **La diagénèse** (recimentation d'éléments après dissolution sous pression à l'intérieur d'une faille, perméable à l'origine).

Pour chaque type de mécanisme, des formules sont proposées afin d'estimer si le facteur considéré transforme la faille en barrière ou en couloir pour les fluides. Ces techniques étant basées sur des données souvent *peu fiables*, des **études d'incertitudes** peuvent sûrement être réalisées pour chacun des mécanismes étudiés.

### 2.1 La description des terrains en contact de part et d'autre de la faille

#### 2.1.1 La techniques des Allan Map

Ce type d'étude concerne encore la description géométrique du réservoir, mais en incorporant plus de notions géologiques, concernant le *type de couches en contact*. Elle se réalise en superposant les couches en contact de part et d'autre de la faille, afin de distinguer les contacts sable-sable (permettant a priori la communication entre les deux blocs en contact), les contacts argile-argile et les contacts sable-argile (qui sont a priori les plus favorables à la formation d'un piège). Ceci permet d'analyser les *variations de géométries le long de la faille* (dues au variation de rejet et des caractères physiques de la stratigraphie le long de la faille), qu'une simple section verticale ne permet pas d'analyser. De tels diagrammes sont appelés *Allan Maps*, du nom de leur auteur [131 - Allan].

Dans cet article, l'auteur prend comme hypothèse que les failles ne constituent en elles-mêmes ni une barrière totale, ni un conduit pour les fluides, mais que cette propriété des failles est **fortement liée aux terrains qu'elles mettent en contact**. Afin de simplifier le graphisme, seules les couches poreuses sont dessinées [figure 1].

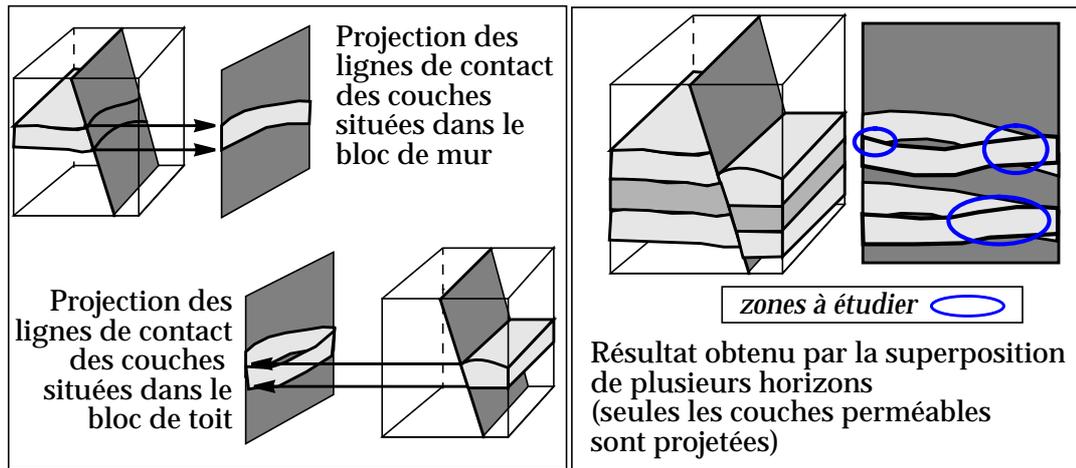


Figure C-1 : Principe de la construction des Allan Map.

L'utilisation de techniques de visualisation telles que la mise en transparence de certaines couches, ainsi que la construction de ces modèles en trois dimension plutôt que sur des plans apporte énormément à cette technique, surtout lorsque le réseau de failles est complexe [135 - Hoffman].

### 2.1.2 Etude des incertitudes à ce niveau

Il serait possible d'étudier les incertitudes concernant ces lignes de contact (et donc concernant les zones de contact entre deux couches réservoir). Par exemple, en étudiant différentes possibilités pour connecter les horizons aux failles. Cette étude est fortement liée à la **propriété rejet** des failles [figure 1].

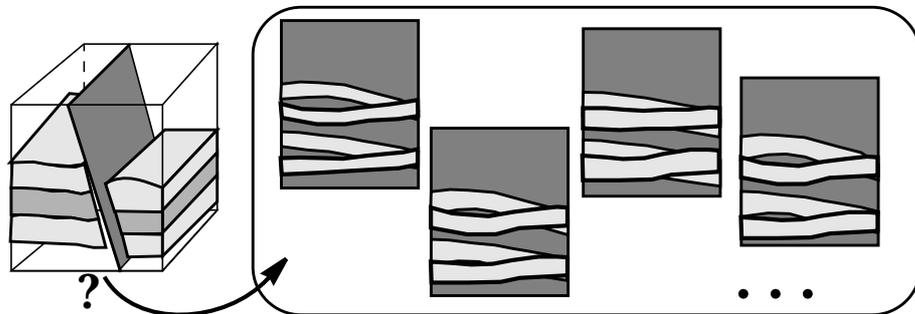


Figure C-1 : Génération d'un ensemble d'Allan Map, en jouant sur le rejet.

Cependant la description géométrique précédente est loin d'être suffisante pour détecter les pièges potentiels. Ainsi, même si deux couches poreuses sont en contact (situation favorable au passage des fluide à travers la faille, a priori), mais que la différence entre la taille des grains est trop importante de part et d'autre, les différences de pressions sont telles que la zone de faille constitue une barrière pour les hydrocarbures, et donc un piège potentiel [144 - Yielding 3]. C'est pourquoi les études géométriques doivent être couplées à des études sur les propriétés physiques des couches.

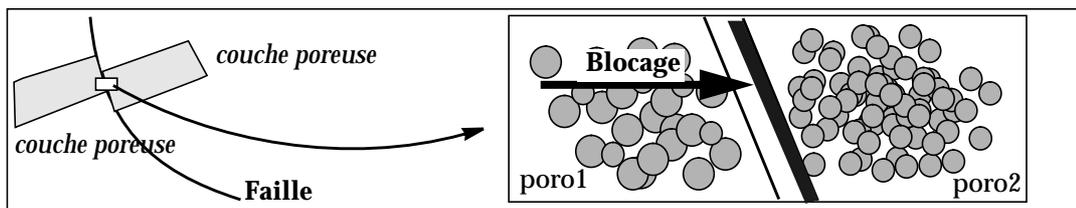


Figure 11-33 : Deux couches poreuses juxtaposées peuvent créer un piège

## 2.2 L'entraînement de matériel à l'intérieur de la faille

D'autres auteurs se sont penchés sur l'effet des sédiments entraînés dans l'épaisseur de la faille. Les recherches actuelles concernent essentiellement l'entraînement d'**argiles** ou de particules fines dans la faille.

De nombreuses formules, souvent *empiriques* sont proposées pour estimer la quantité de sédiments argileux entraînés dans la faille en fonction de paramètres géométriques, qui sont les seuls paramètres réellement disponibles près des failles (rejet, épaisseur des couches d'argile, distance au point où le facteur est calculé) [figure 1].

Parmi ces formules, on trouve: le «*Clay Smear Potential*» [133 - Bouvier], le «*Shale Smear Factor*» (SSF) [139 - Lindsay], le «*Shale Gouge Ratio*» (SGR) [144 - Yielding 3].

Le dernier terme peut prêter à confusion, puisque la même abréviation est très utilisée dans les compagnies pétrolières pour calculer un autre paramètre, le «*Smear Gouge Ratio*», et qui varie de manière inverse. Cette méthode calcule le rapport entre la quantité de sable et la quantité d'argile ayant été déplacés au delà d'une certaine portion du plan de faille.

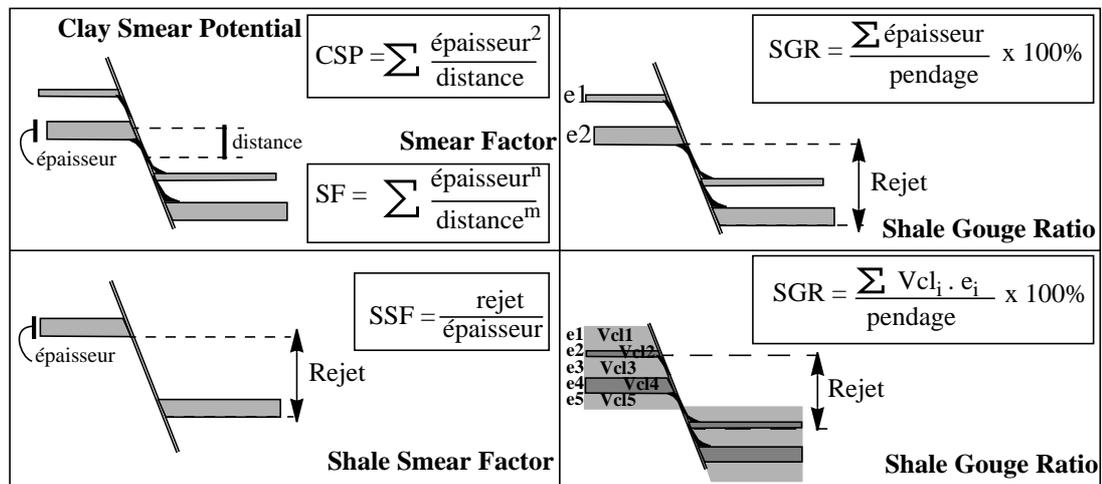


Figure C-1 : Différentes manières d'estimer l'entraînement des argiles dans l'épaisseur de la faille (les termes anglais sont ceux couramment utilisés).

Enfin, si ces études sont réalisées à propos des couches argileuses, les auteurs pensent qu'elles peuvent être facilement transposées aux grains de sables écrasés, puisqu'aucun terme concernant la lithologie n'intervient dans les formules.

Même si ces hypothèses peuvent paraître simplistes, elles présentent l'avantage de ne pas voir la propriété de transmissivité de la faille de manière booléenne (faille complètement scellée ou non scellée), mais comme une *propriété pouvant varier le long de la faille*, ce qui correspond plus aux observations.

Cependant, ces formules étant expérimentales, il doit exister des incertitudes sur les paramètres estimés, et il pourrait être utile d'étudier ces incertitudes.

## 2.3 La diagénèse

Encore peu d'études sont réalisées à ce sujet, et nous n'en dirons pas plus, de peur de dire des choses incorrectes. Cependant, on peut se reporter à des travaux réalisés par Hans Borge [132 - Borge], qui travaille notamment sur la cimentation de quartz sous l'effet de la pression dans les bassins.

## 2.4 Le calcul de la transmissivité des failles

Le calcul de la perméabilité des failles afin d'étudier les écoulements de fluides est réalisé actuellement en affectant un *coefficient de transmissibilité constant* pour chacune des failles (ce qui n'est déjà pas tâche aisée). Ce coefficient de transmissibilité est une combinaison des perméabilités de part et d'autre de la faille (variables le long de la faille), et de la perméabilité propre de la faille (qui est jusqu'à présent supposée constante).

Les études décrites précédemment montrent que ces formules sont très réductrices (puisque la perméabilité des failles est fortement variable) et de plus en plus de travaux tendent à associer un coefficient de transmissibilité en chacune des cellules coupées par la faille, en s'appuyant sur ces formules empiriques [141-Manzocchi1], [142-Manzocchi2]. Cet aspect des failles est à l'étude actuellement dans gOcad [140 - Levy 2].

Les résultats obtenus pourraient eux aussi faire l'objet d'une étude de sensibilité en fonction des incertitudes associées aux différents paramètres.

## 2.5 La modélisation de l'épaisseur des failles

Enfin, l'un des derniers domaines de recherche, concerne le modèle utilisé pour représenter les failles. Plutôt que de les représenter comme une surface ou un plan d'une grille faillée, ces travaux s'attachent à représenter la *géométrie interne* de la faille [134 - Harris], [136 - Knipe 1],[143 - Rives]. En effet, la distribution, les connectivités entre les failles sub-sismiques ayant chacune leur propre perméabilité et leur orientation influent énormément sur le comportement de la zone faillée.

La représentation des failles par une surface plane limite l'étude du comportement des fluides dans cette zone. C'est pourquoi, il est intéressant de pouvoir passer de cette représentation «grossière» à une représentation plus fine de la géométrie de la faille, autorisant une étude précise de ses propriétés. Ce type d'étude repousse encore plus loin la difficulté pour l'étude des failles, et se place à une échelle très différente de la notre (*échelle subsismique*). Cependant, il pourrait être utile de tenir compte des résultats de ces études, afin d'avoir une estimation globale de l'incertitude autour des failles.

La relation entre notre objet faille paramétrique et une représentation sous forme de grille (*faille épaisse*) pourrait éventuellement permettre de faire le lien entre ces études fines, et l'étude réservoir à plus grande échelle.

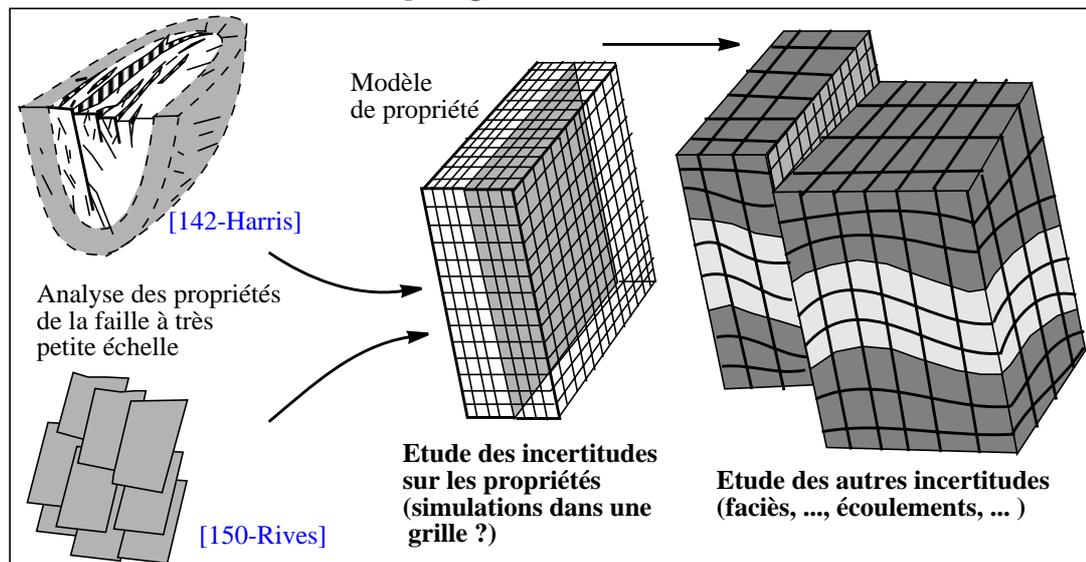


Figure 11-34 : Incorporation des incertitudes sur les propriétés de la faille

Les failles ayant un impact déterminant dans plus de 70% des réservoirs [136 - Knipe 1], on comprend que de nombreux programmes de recherche les étudient de plus en plus précisément.

Comme on le voit, le travail présenté dans ce mémoire constitue une petite partie de l'étude des incertitudes concernant les failles. L'apport de ce travail concerne surtout la prise en compte des incertitudes pour modéliser la géométrie de failles non seulement *non verticales* ce qui constitue une restriction très couramment utilisée jusqu'à présent [68 - Samson], [138 - Lia], mais en plus *non rectilignes* selon l'axe vertical.

Outre les incertitudes concernant la géométrie (à la fois des failles et des horizons), il serait bon de se pencher sur les incertitudes liées aux propriétés des couches et à la manière même dont les failles sont modélisées (zone épaisses endommagées plutôt que des surfaces). Ce sont les thèmes de recherche les plus prometteurs concernant les failles actuellement.

# Bibliographie

## Introduction.

- [1 - Barbier] Barbier Y., (1980), *Dictionnaire du Pétrole*, S.C.M..
- [2 - Catalayaud] Catalayaud P., Gout C., Legrand X., Modiano T. Rives T., (1995), *L'approche modeleur 3D en opérationnelle: une réalité*, Magazine Produire n° 8 , pp 12-17 (magazine interne Elf Exploration Production)
- [3 - Damsleth] Damsleth E., Dubrule O., (1999), *The Pros and Cons of Petroleum Geostatistics*, EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse).

## Chapitre 1: GOCAD- Informatique C++.

- [4 - Cognot 1] Cognot R., (1996), *La méthode D.S.I.: Optimisation, Implémentation et Applications*, Thèse de l' Institut National Polytechnique de Lorraine, Nancy.
- [5 - Duey] Duey R., (1999), *Reservoir-Scale Decisions: New reservoir analysis tools can help E&P companies maximize their assets for minimal cost*, Oil and Gaz World. (trouvé sur le site: [www.geoquest.com/pub/news/pr\\_articles/ogwrld\\_res/index.html](http://www.geoquest.com/pub/news/pr_articles/ogwrld_res/index.html))
- [6 - Mallet 1] Mallet J-L., (1997), *Discrete Modeling for Natural Objects*, Journal of the International Association for Mathematical Geology, Vol. 29, Number 2.
- [7 - Mallet 2] Mallet J-L. (2000), *Geomodeling*, A paraître.
- [8 - Booch] Booch G., (1992), *Conception Orientée Objet et Applications*. Addison Wesley Publishing Company.
- [9 - Ellis] Ellis A., Stroustrup B.,(1991), *The Annotated C++ Reference Manual*, AddisonWesley Publishing Company.
- [10 - Stroustrup] Stroustrup B., (1997), *The C++ Programming Language*, 3rd. ed., Addison Wesley.

## Chapitre 2 : Techniques d'étude des failles.

- [11 - Dresen] Dresen G., Gwilis U., Kluegel Th., (1991), *Numerical and analogue modelling of normal fault geometry*, The Geometry of Normal Faults, Geological society Special Publication n° 56, Ed. A.M. Roberts, G. Yielding and B. Freeman, pp 207-217.
- [12 - Jackson 1] Jackson J. A. (1987), *Active normal faulting and crustal extension*. Continental Extension Tectonics, Geological Society Special Publication n° 28, pp 3-17.

- [13 - Kranz] -Kranz R. W., (1991), *Normal fault geometry and fault reactivation in tectonic inversion experiments*, The Geometry of Normal Faults, Geological society Special Publication n° 56, pp 219-229.
- [14 - Mc Clay 1] Mc Clay K.R., Waltham D.A., Scott A. D., and Abousetta A. (1991), *Physical and seismic modelling of listric normal fault geometries*, The Geometry of Normal Faults, Geological society Special Publication n° 56, pp 231-239.
- [15 - Mc Clay 2] Mc Clay K.R., Ellis P. G., (1987), *Analogue models of extensional fault geometries*, Continental Extensional Tectonics, Geological society Special Publication n° 28, pp 109-126.
- [16 - Schenck] Schenck D., Daum S., (1999), *Open Computing Platform due for Shared Earth Modeling*, Oil and Gas journal.
- [17 - Tipee] Tipee B. (1998), *Collaborative effort seeks computing platform for Shared Earth Modeling*, Oil and Gas Journal (Nov. 98).
- [18 - WWW 1] www.grand-canyon.az.us (site web du grand canyon du colorado)
- [19 - WWW 2] www.flagshipgeo.com (site commercial du logiciel stratimagic)
- [20 - Wernicke] Wernicke B., Burchfiel B.C. (1982), *Modes of extensional tectonics*, Journal of Structural Geology, 4, 2, pp 105-115.
- [21 - Yielding 1] Yielding G., Badley M.E., Freeman B., (1991), *Seismic reflection from normal faults in the northern North Sea*, The Geometry of Normal Faults, Geological society Special Publication n° 56, pp 79-89.

#### Chapitre 2: Généralités sur les failles.

- [22 - Abrahamsen] Abrahamsen P., (1999), *faults and horizon modelling*, (Stochastic aspect of Structural Modelling, Keenote speaker), EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse).
- [23 - Brun] Brun J. P., Choukroune P., (1983), *Normal faulting, block tilting and decollement in a stretched crust*, *Tectonics*, 2, n° 4, pp 345-356
- [24 - Cacas] Cacas M-C., (1999), *Nested Geostatistical and Object-Oriented Modelling of Fracture Systems*, EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse)
- [25 - Davis] Davis G. H., (1984), *Structural Geology of Rocks and Regions*, 492p, Wiley, New York..
- [26 - Dula] Dula W. F., (1991), *Geometric models of listric normal faults and roll over folds*, AAPG Bull. vol. 75, n°10, pp 1609-1625.
- [27 - Foucault] Foucault A. et Raoult J-F, (1996), *Dictionnaire de géologie*, Edition Masson, 4ième édition.
- [28 - Fougère] Fougère E., (1985), *La tectonique en extension intracontinentale:étude de terrain (sud du Nevada) et modélisation analogique*, Thèse de 3ième Cycle UPM Paris VI.

- [29 - Gibbs 1] Gibbs A.D. (1987), *Development of extension and mixed-mode sedimentary basins*, Continental Extension Tectonics, Geological Society Special Publication n° 28, pp 19-33.
- [30 - Gidon] Gidon M., (1987), *Les structures tectoniques*, Manuels et Méthodes, BRGM Editions.
- [31 - Ghosh] Ghosh S.K., (1993), *Structural Geology Fundamentals and Modern Developments*, Pergamon press.
- [32 - Grau] Grau G., (1990), *La recherche pétrolière*, Hermes (56 p.)
- [33 - Gringarten] Gringarten E. (1996), 3-D Description of Fractured Reservoirs, Mathematical Geology, Vol. 28, n° 7, pp881-893.
- [34 - Hardman] Hardman R. F. P. et Booth J.E., (1991), *The significance of normal faults in the exploration and production of North Sea hydrocarbons*, The Geometry of Normal Faults, Geological Society Special Publication n° 56, pp 1-13.
- [35 - Jones P.] Jones P. B., (1989), *Quantitative Geometry of Thrust and Fold Belt Structures*, American Association of Petroleum Geologists, Tulsa.
- [36 - Junger] Junger A. (1976) , *Tectonics of the southern California borderland*, in Howell, D.G., ed.,Aspects of the geologic history of the California continental borderland: Pacific Section,American Association of Petroleum Geologists, Miscellaneous Publication 24, pp 486-498.
- [37 - Jackson 2] Jackson J., Mc Kenzie D. (1983), *The geometrical evolution of normal fault systems*, Journal of Structural Geology, Vol. 5, n° 5, pp 471-482.
- [38 - Mc Clay 3] Mac Clay K.R. (1992), *Glossary of thrust tectonics terms* (pp 419-433), Thrust Tectonics, London Chapman and Hall, 446 p.
- [39 - Mann] Mann P., Hempton M.R., Bradley C. Burke K., (1983), *Developpement of pull-apart basins*, Journal of Geology, vol 91, pp529-554
- [40 - Michel] Michel J-P., Fairbridge R.W., (1996), *Dictionary of Earth Sciences / Dictionnaire des Sciences de la Terre* (anglais-francais / francais-anglais), 2e édition, Masson.
- [41 - Munthe] Munthe K.L., Holden L., Omre H., Damsleth E., Heffer K., Olsen T.S., Watterson J., (1993), *Subseismic Faults in Reservoir Description and Simulation*, Society of Petroleum Engineers (SPE paper n° 2650)0.
- [42 - Naylor] Naylor M.A., Mandl G & Sypesteijn C., (1986), *Fault geometry in basement induced wrench faulting under differential initial stress states*, Journal of Structural Geology, 8: 737-752.
- [43 - Omre] Omre H., Sølna K., Dahl N., Tørudbakken B., (1994), *Impact of Fault Heterogeneity in Fault Zones on Fluid Flow*, Proceedings of the 3rd North Sea Oil and Gaz Reservoirs Conference (Trondheim 92), Kluwer Academic Publishers, pp 185 - 200.
- [44 - Ramsay] Ramsay J.G., Huber M. I., (1987), *The Techniques of Modern Structural Geology*, (Volume 2: Folds and Fractures), Academic Press Inc.

- [45 - Richard] Richard P., Ballard J-F., Coletta B. and Cobbold P., (1989), *Naissance et évolution de failles au dessus d'un décrochement de socle: modélisation analogique et tomographie*, C. R. de l'Accadémie des Sciences, 309, 11, pp 2111-2118.
- [46 - Shelton] Shelton J.W., (1984), *Listric normal faults: an illustrated summary*, AAPG Bulletin n° 68, n° 7, pp 801-815.
- [47 - Sheriff] R. E. Sheriff, L. P. Geldart (1995), *Exploration Seismology*, Chapitre 10 (Geologic interpretation of reflection data), Second Edition, Cambridge University Press
- [48 - Suppe] Suppe J., (1983), *Geometry and kinematics of fault-bend folding*, American Journal of Science, Vol. 283, pp 684-721.
- [49 - Tron 1] Tron V., Brun J. P., (1991), *Experiments in brittle-ductile systems*, Tectonophysics 188, n° 1/2, pp 71-84 .
- [50 - Tron 2] Tron V., (1988), *Rifting oblique: modélisation analogique*, D.E.A. Université de Rennes (138 p) .
- [51 - Vendeville 1] Vendeville B., Cobbold P.R., (1988), *How normal faulting and sedimentation interact to produce listric fault profiles and stratigraphic wedges*, Journal of Structural Geology, Vol. 10, n° 7, pp 649-659.
- [52 - Vendeville 2] Vendeville B., (1987), *Champ de failles et tectonique en extension, modélisation expérimentale*, Mém. Doc. Centre. Arm. Et. Struct. Socles, Rennes, 392 pages
- [53 - Vendeville 3] Vendeville B., (1991), *Mechanisms generating normal fault curvature: a review illustrated by physical models*, The Geometry of Normal Faults, Geological society Special Publication n° 56, pp 241-249.
- [54 - Walsh 1] Walsh J.J., Watterson J., (1988), *Dips of normal faults in British Coal Measures and other sedimentary sequences*, Journal of Structural Geology, Vol. 145, pp 859-873.
- [55 - Yielding 2] Yielding G., Walsh J., Watterson J., (1992), *The prediction of small-scale faulting in reservoirs*, First Break, Vol. 10, n° 12.

#### Chapitre 2 : Détermination de la géométrie des failles.

- [56 - Gibbs 2] Gibbs A.D., (1983), *Balanced cross-section construction from seismic sections in areas of extensional tectonics*, Journal of Structural Geology, Vol. 5, n° 2, pp153-160.
- [57 - Davison] Davison I., (1986), *Listric normal fault profiles: calculation using bed length balance and fault displacement*, Journal of Structural Geology, Vol 8, n° 2, pp 209-210.
- [58 - Galera] Galera C., Bennis C., Moretti I., (1999) , *New tool for geologically coherent 3D bloc diagram*, 1999 American Association of Petroleum Geologists Annual Meeting (AAPG - San Antonio)

- [59 - Roberts] Roberts A.M., Yielding G, Freeman B. (1991), *The Geometry of Normal Faults*, Geological society Special Publication n° 56, Preface.
- [60 - Verall] Verall P. (1981), *Structural Interpretation with Application to North Sea Problems*, Joint Association for Petroleum Exploration Courses (U.K.) 3,
- [61 - White 1] White N. J., Jakson J.A., Mac Kenzie D. P., (1986), *The relationship between the geometry of normal faults and that of the sedimentary layers in their hanging wall*, Journal of Structural Geology, Vol. 8, n° 8, pp 897-909.
- [62 - White 2] White N., (1987), *Constraint on the measurement of extension in the brittle upper crust*, Norsk Geologisk Tidsskrift, Vol 67, pp 269-279
- [63 - White 3] White N. J., Yielding G., (1991), *Calculating normal fault geometries at depth: theory and examples*, The Geometry of Normal Faults, Geological society Special Publication n° 56, pp 1-13.

### Chapitre 3: Chaîne de traitements - sources d'incertitudes.

- [64 - Basire] Basire C., (1998), *3D Structural and Sedimentary Modelling of the Mandaros Field*, rapport CSNS Elf GRC-Imperial College.
- [65 - Henry] Henry G., (1994), *Géophysique des bassins sédimentaires*, Editions Technip.
- [66 - Faily] Faily-Berthet, M., (1996), *Etude Quantitative de la Résolution Sismique par la Réponse Impulsionnelle*, Thèse de Doctorat, Université de Pau et des Pays de l'Adour.
- [67 - Juliard] Juliard C., (1997), *Evaluation de l'information disponible en sismique sur les vitesses*, Thèse de Doctorat, Université de Pau et des Pays de l'Adour.
- [68 - Samson] Samson P., Dubrule O., Euler N. (1996), *Quantifying the impact of Structural Uncertainties on Gross-Rock Volume Estimates*, Society of Petroleum Engineers (SPE paper n° 35535).
- [69 - Shuguang] Shuguang M., Journel A., (1997), *Stochastic Mapping of Layer Surfaces Constrained by Seismic Data and Deviated Well Paths*, Proceedings of the 15th Gocad Meeting, June 97, Nancy.
- [70 - Thore 1] Thore P., (1998), *Champs de Vitesse et Incertitudes Géophysiques*, Mémoire d'habilitation à diriger les recherches.
- [71 - Thore 2] Thore P., (1997), *Incertainces structurales: application à la construction des cartes et au calcul de volume roche*, Rapport interne Elf Exploration Production, EP/P/PRO/RGP/N 95-020, 77 pages.
- [72 - Vincent] Vincent G., Corre B., Thore P., (1999), *Managing Structural Uncertainty in a Mature Field for Optimal Well Placement*, S.P.E. Reservoir Eval. & Eng., Vol. 2, No 4, (Aout 1999).

- [73 - Houlding] Houlding S., (1994), *3D Geoscience Modeling, Computer techniques for Geological characterization*, Springer Verlag, 297p.
- [74 - WWW 3] [www.openspirit.com/publicns.htm](http://www.openspirit.com/publicns.htm) (site détaillant le projet "OpenSpirit" )

#### Chapitre 4-5: Paramétrisation et utilisation de l'objet Faille.

- [75 - Cognot 2] Cognot R., Lecour M. Thore P., (1996), *Geometrical uncertainties on Faults*, Articles et Abstracts du 13<sup>ième</sup> gOcad meeting (Juin 96).
- [76 - Dubrule 1] Dubrule O., Basire C., Bombarde S., Samson Ph., Segonds D. and Wonham J., (1997), *Reservoir Geology using 3-D Modeling Tools*, SPE paper n° 38659.
- [77 - Duvinage 1] Duvinage I., Mallet J-L. (1998), *From autopicked horizons to 3D Fault Blocks Boundary Lines*, 3D Modelling of Natural Objects: a Challenge for the 2000's., (Nancy).
- [78 - Eck] Eck M., DeRose T., Duchamp T., (1995), *Multiresolution Analysis of Arbitrary Meshes*, Computer Graphics (SIGGRAPH Conf. Proc.), pp 173-182 (Aout 1995).
- [79 - Floater] Floater M., (1997), *Parametrization and smooth approximation of surface triangulations*, Computer Aided Geometric Design, Vol. 14, No 3 (Avril 1997), pp 231-250.
- [80 - Krasner] Krasner G.E., Pope S.T., (1988), *A cookbook for using the Model-View-Controller user interface paradigm in Small-talk-80*. Journal of Object-Oriented Programming, Vol. 1, n° 3, pp 26-49.
- [81 - Lecour 1] Lecour M., (1996), *Failles: de l'acquisition sismique à la modélisation 3D*, Mémoire de Diplôme d'Etudes Approfondies, Université Nancy I.
- [82 - Levy 1] Levy B., (1999), *Topologie Algorithmique: combinatoire et plongement*, Thèse de l' Institut National Polytechnique de Lorraine, Nancy.
- [83 -Meagher] Meagher D., (1982), *Geometric modeling using octree encoding*, Comput. Graph. Image Process. Vol. 19, pp 129-147.
- [84 - Mostad] Mostad P., (1999), *Stochastic modelling of seismic faults*, EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse).
- [85 - Wietzerbin ] Wietzerbin L. (1994), *Modélisation et paramétrisation d'objets naturels de formes complexes en trois dimensions. Application à la simulation stochastique de la distribution d'hétérogénéités au sein des réservoirs pétroliers*, Thèse de l' Institut National Polytechnique de Lorraine, Nancy

#### Chapitre 6: Méthodes de simulation.

- [86 - Audinet] Audinet C. (2000), *Etude des incertitudes au niveau des failles: quelques améliorations*, Mémoire de Diplôme d'Etudes Approfondies, Université Nancy I

- [87 - Bagtzoglou] Bagtzoglou A., Stirewalt G., Henderson D., Seida S., (199?), *Evaluation of uncertainty in geological framework models at Yucca montain, Nevada*, .
- [88 -Deutsch 1] Deutsch C., Wang L., (1996), Hierarchical Object-Based Stochastic Modeling of Fluvial Reservoirs, *Mathematical Geology*, Vol 28, No 7, pp. 857-880.
- [89 -Deutsch 2] Deutsch C., Journel A., (1993), *GSLib: Geostatistical Software Library and User's Guide*, Oxford University Press, New York 1993.
- [90 - Froidevaux] Froidevaux R. (1993), *Probability Field Simulation*, Geostatistics Troia'92, Vol. 1, Quantitative Geology and Geostatistics, Kluwer Academic Publishers, Amilcar Soares Ed..
- [91 - Journel 1] Journel A., (1989), *Fundamentals of Geostatistics in Five Lessons*, Short Course in Geology, Vol. 8, American Geophysical Union, 40p.
- [92 - Journel 2] Journel A., (1999), *Past, Present and Future of Petroleum Geostatistics*, EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse).
- [93 - Lecour 2] Lecour M., Cognot R., Thore P., Dulac J-C., (1998), *Modelling faults taking uncertainties into account*, 1998 Society of Exploration Geophysicists Annual Convention ( SEG - New Orleans).
- [94 - Lecour 3] Lecour M., (1998), *Modelling faults taking uncertainties into account*, 3D Modelling of Natural Objects: a Challenge for the 2000's. (Nancy).
- [95 - Mallet 3] Mallet J-L., Shtuka A. (1997), *Periodic P-Field based Simulator*, Articles et Abstracts du 16ième gOcad meeting (Novembre 97 - Dallas)
- [96 - Srivastava] Srivastava, M., R., 1992. *Reservoir characterization with probability field simulation.*, S.P.E. Paper 24753 (S.P.E. Washington 92).
- [97 - Suro] Suro-Pérez V., Journel A., (1990), *Stochastic Simulation of Lithofacies for Reservoir Characterization*, Proceedings of the 2<sup>nd</sup> European Conference on Mathematics of Oil Recovery (Paris), pp 3-10.
- [98 - Thibaut] Thibaut M., *Géométrie des surfaces de faille et dépliage 3D. Méthodes et applications*, Thèse de Doctorat, Grenoble 1.
- [99 - Tran] TranT. (1999), *Developments in object-based modelling: Object parameterization and data conditioning*, EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse).
- [100 - Viseur] Viseur S., Shtuka A., Mallet J-L., (1998), *New Fast Stochastic Boolean Simulation of Fluvial Deposits*, Society of Petroleum Engineers (S.P.E paper n° 49281).
- [101 - Vistellius] Vistellius A., (1976), *Mathematical Geology and Development of the Geological Sciences*, Journal of the International Association for Mathematical Geology, Vol. 8, n° 1, (Fev 76).

**Chapitre 7-8 : Création de modèles volumique.**

- [102 - Duvinage 2] Duvinage I., (1999), Mallet J-L., Dulac J-C, (1999), *Constructing structural reservoir models without computing intersections*, 1999 Annual Conference of the International Association for Mathematical Geology (IAMG - Trondheim).
- [103 - Duvinage 3] **Duvinage I.,(1999), *Construction semi-automatique de modèles volumiques à partir de semis de points issus de l'interprétation sismique (titre provisoire)*,Thèse de l' Institut National Polytechnique de Lorraine, Nancy (sera soutenue le 23 juin 2000 ...)**
- [104 - Dulac] Dulac J-C., Massot J. (1998), *Automatic Fault Network Construction from Fault Interpretations*, Gocad Meeting Septembre 1998 (New Orleans).
- [105 - Euler 1] Euler N., Sword C. H., Dulac J-C, (1998), A new tool to seal a 3D Earth Model: a cut with constraints, 1998 Society of Exploration Geophysicists Annual Convention ( SEG - New Orleans)
- [106 - Euler 2] Euler N., (1999), *Modélisation volumique, Contraintes et libertés*, Thèse de l' Institut National Polytechnique de Lorraine, Nancy
- [107 - Lasseter] Lasseter, T., (1992), *An interactive 3-D modeling system for integrated interpretation in hydrocarbon reservoir exploration and production*, Computer Graphics in Geology: Three-Dimensional Computer Graphics in Modeling Geologic Structures and Simulating Geologic Processes, Ed. Plug R. and Harbaugh J., Springer-Verlag , pp 189 - 198.
- [108 - Mariez] Mariez O., (1998), *Modélisation de solides par synthèse de l'analyse d'image 3D et de modèles à base de surfaces non-variétés*, Thèse de l' Institut National Polytechnique de Lorraine, Nancy
- [109 - Orlic] Orlic B., (1997), *Predicting Subsurface Conditions for Geotechnical Modelling*, Ph. D Thesis, Delft University of Technology, Netherlands (193 p.).
- [110 - Requicha] Requicha A. , (1980), *Representations for Rigid Solids: Theory, Methods and Systems*. ACM Computing Surveys, December 1980.
- [111 - Weiler] Weiler, K. 1988, *The radial edge structure: a topological representation for non-manifold geometric boundary modelling*. Geometric modelling for CAD Applications, pages 3-36, North Holland.

**Chapitre 9: Etudes volumétriques : simulations géométriques de modèles volumiques.**

- [112 - Cognot 3] Cognot R., Mugerin C., Mallet J-L, (1999), *Fitting Curvilinear Stratigraphic grids to Horizons and Faults*, Articles et Abstracts du 19<sup>ième</sup> gOcad meeting (Nancy - Juin 99)
- [113 - Quettier] L. Quettier, (1994), *Méthodologie d'utilisation des modèles: Maillage*, Rapport interne Elf E. P. (EP/P/PRO/GIN LQ/lq -n° 94-224).

- [114 - Huang ] Huang Y-G, (1990), *Modélisation et Manipulation des Surfaces Triangulées*, Thèse de l' Institut National Polytechnique de Lorraine, Nancy.
- [115 - Lecour 4] Lecour M., Thore P., Dulac J-C, (1999), *Stochastic fault and fault networks modelling for structural uncertainties study*, EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse).  
(Sélectionné pour une publication dans *Petroleum Geoscience*)
- [116 - Lecour 5] Lecour M., Thore P., Dulac J-C, Cognot R., (1999), *From a-priori to stochastic structural models: how to incorporate uncertainties during fault modelling*, 1999 Annual Conference of the International Association for Mathematical Geology ( IAMG - Trondheim).
- [117 - Levy 2] Levy B., Mallet J-L., (1998), *Non-Distorted Texture Mapping for Sheared Triangulated Meshes*, Computer Graphics (SIGGRAPH Conf. Proc.) (SIGGRAPH - Orlando)
- [118 - Levy 3] Levy B., Mallet J-L., (1998), *Faulted Stratigraphic Grids: Direct construction*. Articles et Abstracts du 17<sup>ième</sup> gOcad meeting (Nancy - Juin 98)

#### Chapitre 10 Etude des réseaux de failles: hiérarchisation et corrélation en 3D.

- [119 - Barnett] Barnett J. A. M., Mortimer J., Rippon J.H, Walsh J.J. and Watterson J., (1987), *Displacement geometry in the volume containing a single normal fault*, AAPG Bulletin, Vol. 9, n° 8, pp 925-937.
- [120 - Chapman] Chapman T.J., Meneilly A.W., (1991), *The displacement patterns associated with a reverse-reactivated normal growth fault*, The Geometry of Normal Faults, Geological Society Special Publication n° 56, pp 183-191.
- [121 - Freeman] Freeman B., Yielding G., Badley M., (1987), *Fault corrélation during seismic interprétation*, First Break Vol. 8, n° 3.
- [122 - Gibbson] Gibbson J.R., Walsh J.J., Watterson J. (1989), *Modelling of bed contours and cross-sections adjacent to planar normal faults*, Journal of structural Geology, 11, pp 317-328.
- [123 - Maerten 1] Maerten L. (1998) , *Inferring Fault Linkage from Seismic Data and Numerical Analysis: Application to the Oseberg Syd Reservoir Compartmentalization*, American Association of Petroleum Geologists 1998 Annual Convention (AAPG - Salt Lake City).
- [124 - Maerten 2] Maerten L., Willemse E. J-M, Pollard D., Rawnsley K. (1999) , *Slip distributions on intersecting normal faults*, Journal of Structural Geology vol. 21, n° 3, pp 259-271.
- [125 - Morley] Morley C. K., (1999), *Patterns of Displacement Along Large Normal Faults: Implications for Basin Evolution and Fault Propagation, Based on Examples from East Africa*, AAPG Bulletin, Vol. 83, n° 4, pp 613 - 634.
- [126 - Rippon] Rippon J.H., (1985), *Contoured patterns of the throw and the hade of normal faults in the coal measures (westpalian) of North east Derbyshire*, Pro-

ceeding of the Yorkshire Geological Society Vol. 45, part 3, pp 147-161.

- [127 - Walsh 2] Walsh J.J., Watterson J., (1987), *Distributions of cumulative displacement and seismic slip on a single normal fault surface*, Journal of Structural Geology, Vol. 9, n° 8, pp 1039 - 1046.
- [128 - Walsh 3] Walsh J.J., Watterson J., (1989), *Displacement gradients on fault surfaces*, Journal of Structural Geology, Vol. 11, n° 3, pp 307-316.
- [129 - Walsh 4] Walsh J.J., Watterson J., (1991), *Geometric and kinematic coherence and scale effects in normal fault systems*, The Geometry of Normal Faults, Geological Society Special Publication n° 56, pp 193-203.
- [130 - Willemse] Willemse E. J-M., Pollard D.D., Aydin A., (1996), *Three dimensional analysis of slip distributions on normal fault arrays with consequences for fault scaling*, Journal of Structural Geology, Vol. 18, pp 295-309.

#### Conclusion ( étude des propriétés des failles).

- [131 - Allan] Allan, U., (1989), *Model for Hydrocarbon Migration and Entrapment Within Faulted Structures*, AAPG Bulletin, V. 73, n° 7 (July 89), pp 803-811.
- [132 - Borge] Borge H., (1999), *Quantification of mechanisms for generating overpressures in the Haltenbanken region*, 1999 Annual Conference of the International Association for Mathematical Geology (IAMG - Trondheim).
- [133 - Bouvier] Bouvier J.D., Kaars-Sijpesteijn C. H., Kluesner D.F., Onyejeckwe C.C., Van der Pal R. C., (1989), *Three-dimensional seismic interpretation and fault sealing investigations*, Nun River Field, AAPG Bulletin, V. 73, pp 1397-1414.
- [134 - Harris] Harris S. D., Mc Allister E., Knipe R. J., Elliot L., Ingham D. B. (1999), *Scaling of fluid flow associated with flow through fault damage zones and networks*, 1999 Annual Conference of the International Association for Mathematical Geology (IAMG - Trondheim).
- [135 - Hoffman] Hoffman K., Taylor D., Schnell R., (1996), *3-D improves/speeds up fault plane analysis*, The Leading Edge (Fev. 96), pp 117 - 122.
- [136 - Knipe 1] Knipe R. J., Jones G., Fisher Q. J., Mc Allister E., Harris S.J., Needham T., Porter J. R., Edwards E., Rowe J., Bolton A., Henson D., (1999), *Fault Seal Evaluation: Recent advances, limitations and future directions*, conférence FORCE (Forum for Reservoir Characterization and Reservoir Engineering), (Stavanger 99)
- [137 - Knipe 2] Knipe R., (1997), *Juxtaposition and seal Diagrams to help Analyze Fault Seals in Hydrocarbon Reservoirs*, AAPG Bulletin, Vol. 81, n° 2, (Février 1997), pp 187-195.

- 
- [138 - Lia] Lia O., Omre H., Tjelmeland H., Holden L., Egeland T., (1997), *Uncertainties in Reservoir Production Forecasts*, AAPG Bulletin, V. 81, n° 5 (Mai 97), pp 775-802.
- [139 - Lindsay] Lindsay N.G., Murphy F.C., Walsh J.J., Watterson J., (1993), *Outcrop studies of shale smear on Fault surfaces*, International Association of Sedimentologists Special Publication 15, pp 113-123.
- [140 - Levy 2] Levy B., Mallet J-L, (1998), *Connectivities accross faults in a Stratigraphic Grid: Notion of Transmitter*, Articles et Abstracts du 17ième gOcad meeting (Juin 98 - Nancy)
- [141-Manzocchi1] Manzocchi T., Nell, P., Yielding G., Childs C., Watterson J., (1999), *On the representation of fault zone heterogeneity in reservoir simulation models*, EAGE/SPE International Symposium on Petroleum Geostatistics (Toulouse).
- [142-Manzocchi2] Manzocchi T., Walsh J. J., Heath A., Nell P., Yielding G., (1999), *A new method for attaching fault transmissibility multipliers to flow simulation models: pros and cons*, conférence FORCE (Forum for Reservoir Characterization and reservoir Engineering), (Stavanger 99)
- [143 - Rives] Rives T., Benedicto A., (1999), *Segmented Faults - Implications for Exploration and Production*, conférence FORCE (Forum for Reservoir Characterization and reservoir Engineering), (Stavanger 99)
- [144 - Yielding 3] Yielding G., Freeman B., Needham D., (1997), *Quantitative Fault Seal Prediction*, AAPG Bulletin, Vol. 81, n° 6 (Juin 1997), pp 897-917.