

# Mémoire de Magister

Spécialité : Génie mécanique  
CFAO

## OPTIMISATION DES TRAJECTOIRES D'OUTIL EN USINAGE 2/3 AXES DE SURFACES GAUCHES DE MODELES ET DEGRES QUELCONQUES

Présenté par

**Bilel HASSAID**

*Soutenu devant le jury suivant le 21-07-1999*

M.CHELGHOUM Abdelkrim

M.BELAIDI Idir

M.MOHAMMEDI Kamel

M. NOUR Abdelkader

M.BELAIDI Abderrahmane

M.IGHILAZA Abdelkrim

M. RACHEDI Kamel

Professeur à l'U.B, Président de jury

Directeur de mémoire et Rapporteur

Maître de Conf. à l'U.B, Examineur

Maître de Conf. à l'U.B, Examineur

MACC à l'U.B, Examineur

MACC à l'U.S.T.H.B, Examineur

MACC à l'U.B, invité

## **Remerciements**

A l'issue de la réalisation du présent travail, je tiens à exprimer mes profonds et sincères remerciements à :

Monsieur le Recteur de l'Université de Boumerdès, Monsieur Abdelkrim CHELGHOUM, pour l'honneur qu'il me fait en présidant le jury de soutenance ;

Mon directeur de thèse, Monsieur Idir BELAIDI, Docteur en C.F.A.O à l'Université de Boumerdès et responsable du Laboratoire de CFAO du Département de Mécanique de la Faculté des Sciences de l'Ingénieur de l'Université de Boumerdès (Ex. INGM), qui m'a permis de faire mes premiers pas en CFAO dès 1991 et pour les encouragements qu'il m'a prodigué;

Messieurs Kamel MOHAMMEDI, Maître de Conférences à l'Université de Boumerdès, Abdelkader NOUR, Maître de Conférences à l'Université de Boumerdès, Abderrahmane BELAIDI Chargé de cours à l'Université de Boumerdès, Abdelkrim IGHILAZA, Chargé de Cours à l'Université de Bab Ezzouar, Kamel RACHEDI Chargé de cours à l'Université de Boumerdès, pour l'intérêt manifesté à l'égard de mon travail, pour leurs remarques et suggestions pertinentes et pour m'avoir honoré en voulant bien être membres du jury de soutenance ;

Monsieur Malik DIB, Maître Assistant à l'Université de Bab Ezouar, pour ses remarques et suggestions ;

Monsieur Fayçal HOCINE, Directeur Général de l'ex. INGM, pour son précieux soutien ;

Mon père pour ses lectures répétées et pertinentes du présent mémoire ;

Monsieur le Directeur de la Base Centrale Logistique, le Colonel Ali AKROUM, pour m'avoir permis de continuer ce travail pendant la période de mon service national ;

Mon ami Mohand-Oulhadj CHALLALI, pour ses encouragements et son soutien ;

Et, à tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

## **Dédicaces**

Je dédie ce travail,

A la mémoire de mon grand-père, pour m'avoir appris à aimer la mécanique dès mon jeune âge, et pour m'avoir inculqué les bases du raisonnement logique;

A mon père, pour tous ses sacrifices et son soutien inconditionnel, et pour m'avoir appris à vénérer le travail et ne tolérer que le travail bien fait;

A ma mère, pour son amour et son soutien chaleureux;

A mon frère Lotfi et mes sœurs Chebilia, Mériem et Adila;

A ma femme Samia et mes enfants Farah, Aymen et Ilyas;

A la mémoire des mes amis Salih GUENDOUCZ et Azedine SAIDJ, qui ont payé de leur vie leur amour pour leur patrie;

A tous ceux et celles qui portent l'Algérie au plus profond de leur cœur.

## Table des matières

Remerciements	I
Dédicace	II
Table des matières	III
Table des illustrations	VII
Liste des abréviations – Notations	IX
Résumés	X
<i>Introduction</i>	1
<i>1<sup>ère</sup> Partie : Modélisation géométrique et usinage de surfaces gauches</i>	3
<b>1. Modélisation géométrique</b>	3
<b>1.1 Introduction</b>	3
<b>1.2 Classification des modèles</b>	4
<b>1.2.1 Modèle intégral de Bezier</b>	4
1.2.1.1. Courbe de Bézier non rationnelle	4
1.2.1.2. Surface de Bézier non rationnelle	5
1.2.1.3. Courbes de Bézier rationnelles	5
1.2.1.4. Surfaces de Bézier rationnelles	6
<b>1.2.2 Modèle par morceau B_Spline</b>	7
1.2.2.1 Fonction de base B-spline	7
1.2.2.2 Courbes B-spline non rationnelles	8
1.2.2.3 Surfaces B-spline non rationnelles	9
1.2.2.4 Courbes B-spline rationnelles	9
1.2.2.5 Surfaces B-spline rationnelles	9
1.2.2.6 Courbe et surfaces NURBS	10
<b>1.3 Conclusion</b>	10



<b>2. Synthèse des méthodes de planification et de calcul des trajectoires d'outil</b>	11
<b>2.1 Introduction</b>	11
<b>2.2 Planification des trajectoires</b>	12
<b>2.2.1 Classification</b>	12
2.2.2.1 Trajectoires en spirales par décalage de contours	12
2.2.2.2 Trajectoires en spirales par réduction de contour	12
2.2.2.3 Trajectoires en zigzag parallèles	13
2.2.2.4 Trajectoires en zigzag iso paramétriques	13
2.2.2.5 Trajectoires uni directionnelles iso paramétriques	13
2.2.2.6 Trajectoires uni directionnelles parallèles	14
2.2.2.7 Courbes de remplissage	14
<b>2.2.2 Problèmes posés</b>	15
2.2.2.1 La hauteur de crête	15
2.2.2.2 L'erreur de corde	18
2.2.2.3 Relations entre erreur de corde et hauteur de crête	20
2.2.2.4 Tolérances de CFAO	21
2.2.2.5 Modèle de prédiction de la Tolérance de CFAO	26
2.2.2.6 Les interférences et collisions	27
<b>2.2.3 Etude comparative et problématique</b>	28
<b>2.3. Méthodes de calcul des trajectoires d'outils</b>	31
<b>2.4 Synthèse bibliographique des techniques d'usinage d'ébauche</b>	33
<b>2.4.1 Usinage 2 axes ½</b>	33
2.4.2.1 Méthode d'Herman	33
2.4.2.2 Méthode de Loney et Ozoy	34
2.4.2.3 Méthode de Lee et Chang	36
2.4.2.4 Techniques d'évidement de poches planes	36
<b>2.4.2 Problèmes posés et étude critique</b>	41
<b>2.5 Synthèse bibliographique des techniques d'usinage de finition</b>	41
<b>2.5.1 Classification</b>	41
2.5.2.1 Méthode Automatic Programing Tools " APT "	42
2.5.2.2 Méthode de Bobrow	43
2.5.2.3 Méthode de Choi, Lee, Hwang et June	44
2.5.2.4 Méthode de Choi et June	46
2.5.2.5 Méthode de Hwang	47
2.5.2.6 Méthode de Lee Choi et Chang	49
2.5.2.7 Méthode de Oliver, Wysocki et Goodman	50
2.5.2.8 Méthode de Suresch et Yang	51
2.5.2.9 Méthode de Marschal et Griffiths	52
2.5.2.10 Méthode de Kim et Ko	53
2.5.2.11 Méthode de Kim et Kim	54
<b>2.5.2 Etude comparative et problématique</b>	55

<b>2<sup>ème</sup> Partie : Usinage de formes gauches à trajectoires optimisées</b>	<b>57</b>
<b>1. Système d'usinage élaboré</b>	<b>57</b>
<b>1.1 Aperçu général</b>	<b>57</b>
<b>1.2 Conception flexible</b>	<b>61</b>
1.2.1 Problème posé	61
1.2.2 Solutions conventionnelles	61
1.2.2 Solutions par sub-réduction de degré	62
<b>1.3 Méthode d'usinage d'ébauche</b>	<b>64</b>
1.3.1 Présentation générale	64
1.3.2 Choix d'un type de trajectoires	66
1.3.3 Discrétisation des surfaces du modèle	68
1.3.3.1 Principe de discrétisation	68
1.3.3.2 Subdivision d'un carreau de Bézier	68
1.3.3.3 Tolérances de discrétisation	70
1.3.4 Intersection plan de coupe /modèle	72
1.3.4.1 Transformation de coordonnées	73
1.3.4.2 Calcul des points d'intersection plan de coupe/ modèle facettisé	74
1.3.4.3 Calcul des points d'intersections exactes	76
1.3.5 Construction des contours	78
1.3.5.1 Etablissements des contours	78
1.3.5.2 Orientation des contours	78
1.3.6 Génération des trajectoires d'outil	80
1.3.6.1 trajectoires de contournage	82
1.3.6.2 Polygonisation des contours	89
1.3.6.3 Trajectoires d'évidements	89
1.3.6.4 Jumelage des trajectoires de contournage et d'évidement	89
<b>1.4 Méthode d'usinage de finition proposée</b>	<b>90</b>
1.4.1 Choix d'un type de trajectoires	90
1.4.2 Calcul des points de contacts outil/piece	90
1.4.3 Calcul du pas transversal	90
1.4.4 Calcul des points de localisation de l'outil	91
1.4.5 Gestion des interférences	91
1.4.5.1 Considérations générales	91
1.4.6 Génération des informations pour une reprise locale d'usinage	96
<b>1.5 Conclusion</b>	<b>97</b>

<b>2. Mise en œuvre informatique et validation</b>	98
<b>2.1 Introduction</b>	98
<b>2.2 Environnement de développement utilisé</b>	100
<b>2.3 Module de Conversion / Discrétisation</b>	101
2.3.2 Algorithmes et implémentation	101
2.3.3 structure de données	105
2.3.4 Exemples d'exécution	107
<b>2.4 Module d'usinage d'Ebauche</b>	108
2.4.1 Organigramme général	108
2.4.2 Algorithmes et Implémentation	108
2.4.3 Structure de données	114
2.4.4 Exemples d'exécution	115
<b>2.5 Module d'usinage de Finition</b>	118
2.5.1 Organigramme général	118
2.5.2 Algorithmes et Implémentation	118
2.5.3 structure de données	122
2.5.4 Exemples d'exécution	122
<b>2.6 Validation</b>	123
2.6.1 Objet de l'étude comparative	123
2.6.2 Chaîne expérimentale du processus de validation	123
2.6.3 Présentation du logiciel de CFAO « <i>SurfCAM</i> »	125
2.6.4 Conditions expérimentales	125
2.6.5 Résultats obtenus	125
2.6.5.1 Limitations communes à «SURFCAM» et à la méthode proposée	125
2.6.5.2 Possibilités étendues de la méthode proposée	126
2.6.5.3 Analyse des trajectoires d'usinage obtenu	127
2.6.6 Conclusion	131
 <b><i>Conclusion générale</i></b>	 132
<b>Références bibliographiques</b>	<b>XIII</b>
<b>Annexes</b>	
<b>Annexe I : Détails de calcul de la hauteur de crête</b>	<b>XX</b>
<b>Annexe II : Organigrammes</b>	<b>XXXIII</b>
<b>Annexe III : Intégration des modules sous AutoCAD V.14</b>	<b>XXXIX</b>

## Table des illustrations

Figure I 1.1	Courbe de Bézier non rationnelle cubique	4
Figure I 1.2	Influence des poids $w_i$ sur la forme d'une courbe de Bézier rationnelle	6
Figure I 2.1	Trajectoires en spirales par décalage de contours	12
Figure I 2.2	Trajectoires en spirales par réduction de contour	12
Figure I 2.3	Trajectoires en zigzag parallèles	13
Figure I 2.4	Trajectoires en zigzag iso paramétriques	13
Figure I 2.5	Trajectoires unidirectionnelles iso paramétriques	14
Figure I 2.6	Trajectoires unidirectionnelles parallèles	14
Figure I 2.7	Trajectoires par courbes de remplissage	15
Figure I 2.8	La hauteur de crête et son aspect sur la surface usinée.	15
Figure I 2.9	Mesure de la hauteur de crête $h$ .	16
Figure I 2.10	Approximation locale de la surface par une sphère.	16
Figure I 2.11	Invariance de la hauteur de crête entre usinage 3 axes et 5 axes à l'outil hémisphérique.	18
Figure I 2.12	Erreur de corde et tolérance de conception $T_c$ .	18
Figure I 2.13	Calcul de l'erreur de corde.	19
Figure I 2.14	Relations entre erreur de corde et hauteur de crête	21
Figure I 2.15	Modèle mathématique (a) et représentation infographie (b) d'une surface gauche.	22
Figure I 2.16	Définition de l'intervalle de tolérance de CFAO théorique.	23
Figure I 2.17	Influence de TC et de TF sur l'intervalle de tolérance de CFAO théorique.	23
Figure I 2.18	Relation entre TC et TF. Configuration théorique, Configuration critique réelle.	24
Figure I 2.19	Interférence outil/pièce locale ( $R > (\rho)$ ).	27
Figure I 2.20	Passe de finition ( $R < \min(\rho)$ )	28
Figure I 2.21	Passe de finition ( $R > \min(\rho)$ ) nécessitant une reprise locale d'usinage	28
Figure I 2.22	Reprise locale d'usinage ( $R < \min(\rho)$ )	28
Figure I 2.23	Point de contact outil/pièce et point de contrôle outil.	31
Figure I 2.24	Discretisation polyédrique de la surface.	32
Figure I 2.25	Calcul des CL points par construction de surface parallèle.	32
Figure I 2.26	Planification des trajectoires d'Herman.	34
Figure I 2.27	Planification des trajectoires de Loney et Ozoy.	35
Figure I 2.28	Trajectoires d'outil de Loney et Ozoy.	35
Figure I 2.29	Trajectoires d'évidement de poche de Persson.	37
Figure I 2.30	Élimination des boucles.	38
Figure I 2.31	Projection du contour sur la surface du fond avec respect de la tolérance de corde.	39
Figure I 2.32	discretisation des surfaces pour les calculs d'intersections.	39
Figure I 2.33	Utilisation des ponts dans le cas de présence d'îlots.	40
Figure I 2.34	construction des trajectoires d'outils et « graphe des sous poches monotones »	41
Figure I 2.35	La méthode A.P.T.	42
Figure I 2.36	Passe d'usinage par plans de coupe quelconques (a), et parallèles à $\{ox, oz\}$ (b).	34
Figure I 2.37	utilisation du modèle C.S.G pour le calcul des trajectoires d'outil	44
Figure I 2.38	Quadrillage du domaine.	45
Figure I 2.39	Passage du pas curviligne $l$ au pas cartésien $lp$ .	45
Figure I 2.40	Zone de contact outil/pièce.	46
Figure I 2.41	Élimination des interférences outil/pièce.	47
Figure I 2.42	Relations topologique entre CC points voisins.	48
Figure I 2.43	Restriction de la forme du contour extérieur de la pièce.	48
Figure I 2.44	Élimination des interférences outil/pièce.	49
Figure I 2.45	Élimination des interférences outil/pièce.	50
Figure I 2.46	Détails de calcul de l'erreur de corde	51
Figure I 2.47	Taux de réduction de la taille du CL File.	52
Figure I 2.48	Trajectoires d'usinage de finition.	53
Figure I 2.49	Discretisation de la surface.	54
Figure II 1.1	Organigramme général du système de CFAO proposé	57
Figure II 1.2	Structure du système de CFAO proposé.	58
Figure II 1.3	Détails des modules du système de CFAO proposé	59
Figure II 1.4	Principe et applications de la sub-réduction	63
Figure II 1.5	Organigramme général de la méthode d'ébauche proposée	65
Figure II 1.6	Déplacements à vide excessifs des trajectoires en zigzag parallèles.	66

Figure II 1.7	Transition entre sous poches, trajectoires par décalage de contours	67
Figure II 1.8	Optimisation des déplacements a vide des trajectoires en zigzag parallèles	67
Figure II 1.9	Discrétisation auto adaptative d'un carreau de Bézier	68
Figure II 1.10	Courbe de Bézier rationnelle de degré 5 subdivisée en deux courbes rationnelles de degré 5	69
Figure II 1.11	Boite mini-maxi d'une facette gauche	70
Figure II 1.12	Exemple de singularité de la méthode d'évaluation de linéarité/planité de la facette	71
Figure II 1.13	Planéité et linéarité des bords à l'aide des vecteurs normaux et tangents aux coins	71
Figure II 1.14	Discrétisation auto adaptative d'un carreau de Bézier	72
Figure II 1.15	Repère local lié aux plans de coupe	74
Figure II 1.16	Déterminations des paramètres approchés ( $u^*, v^*$ ) des points d'intersection P1 et P2.	75
Figure II 1.17	Orientation des contours. outil à droite et outil à gauche.	80
Figure II 1.18	Méthode de planification des trajectoires proposée.	81
Figure II 1.19	Tolérance de Surépaisseur de Finition (TSF).	81
Figure II 1.20	Calcul des points de contrôles de l'outil.	82
Figure II 1.21	Passage du point de contact outil/pièce au point de localisation de l'outil	83
Figure II 1.22	Trajectoires de contournage à l'outil cylindrique.	83
Figure II 1.23	Points de rebroussement sur la trajectoire d'outil donnant lieu à une interférence.	84
Figure II 1.24	Auto-intersection de la trajectoire d'outil donnant lieu à une interférence.	84
Figure II 1.25	Intersection inter- contour.	84
Figure II 1.26	Discontinuités de la trajectoire d'outil.	84
Figure II 1.27	Calcul du pas de plongée dans le cas de l'outil cylindrique.	85
Figure II 1.28	Passage d'un point de contacte outil/pièce au point de localisation de l'outil correspondant.	85
Figure II 1.29	Calcul du pas de plongée dans le cas de l'outil hémisphérique.	86
Figure II 1.30	Passage d'un point de contacte outil/pièce au point de localisation de l'outil	87
Figure II 1.31	Intersection partie torique/partie torique.	87
Figure II 1.32	Intersection partie torique/bout de fraise	88
Figure II 1.33	Intersection génératrice/bout de fraise.	88
Figure II 1.34	Intersection génératrice/bout de fraise.	88
Figure II 1.35	Polygonisation des contours.	89
Figure II 1.36	Coefficient de recouvrement des trajectoires parallèles $R_c$ .	89
Figure II 1.37	Zone de contacte outil/pièce	91
Figure II 1.38	Elimination de l'interférence par incrémentation de la cote z de l'outil.	92
Figure II 1.39	Répercussions de l'incrémentement de la cote z sur les conditions de formation de copeau.	93
Figure II 1.40	Passé de finition ( $R > R_{max}$ ) nécessitant une reprise locale d'usinage	96
Figure II 1.41	Reprise locale d'usinage ( $R > R_{max}$ )	97
Figure II 2.1	Application externe autonome	98
Figure II 2.2	Application externe autonome avec accès partagé a la base de données	99
Figure II 2.3	Application totalement intégrée au logiciel	99
Figure II 2.4	Environnement de développement	100
Figure II 2.5	Exemple de simulation du processus de subdivision	106
Figure II 2.6	Arbre quaternaire issue du processus de subdivision	106
Figure II 2.7	Etat de la pile et de la liste au cours du processus de subdivision	107
Figure II 2.8	Surface de Bézier de degré (6x6).	107
Figure II 2.9	Facétisation de la surface avec une tolérance de 0.2mm(a) et 0.02mm (b).	108
Figure II 2.10	Structure de données des segments d'intersections	115
Figure II 2.11	Structure de données des courbes d'intersections	115
Figure II 2.12	Courbes d'intersection surface/plan (a) et surface/reseau de plans parallèles (b).	116
Figure II 2.13	Trajectoire d'outil correspondant a une passe d'évidement	117
Figure II 2.14	Trajectoire d'outil correspondant à une passe d'évidement	117
Figure II 2.15	Trajectoire d'usinage de finition	122
Figure II 2.16	Organigramme du processus de validation	124
Figure II 2.17	Comparaison des résultats de construction des courbes d'intersection	126
Figure II 2.18	Trajectoires d'ébauche obtenues avec SURFCAM. (a) Trajectoires. (b) Usinage simulé	127
Figure II 2.19	Trajectoires de semi-finition obtenues avec SURFCAM.	128
Figure II 2.20	Trajectoires de finition obtenues avec SURFCAM.	128
Figure II 2.21	Trajectoires d'ébauche obtenues avec le système élaboré pour une TSF très importante	129
Figure II 2.22	Portion des trajectoires d'ébauche obtenues avec le système élaboré pour une $TSF_f = 0.5mm$ .	129
Tableau II 2.1	comparaison des résultats de génération des trajectoires d'outils obtenues	131

## NOTATIONS

A : vitesse d'avance.  
CCP : point de contact outil/pièce,  
CLP : point de localisation de l'outil,  
h : hauteur de crête,  
IT(CF<sub>th</sub>) : intervalle de tolérance de CFAO théorique,  
IT(F<sub>p</sub>) : intervalle de tolérance de forme prescrite,  
K : effort spécifique de coupe,  
 $\vec{n}$  : vecteur normal unitaire,  
N : vecteur normal,  
P<sub>max</sub> : profondeur de passe maxi,  
R : rayon de l'outil hémisphérique,  
r : petit rayon de l'outil torique,  
 $\rho$  : rayon de courbure de la surface,  
TC : tolérance de conception,  
TC<sub>u</sub> : tolérance de conception utile,  
TF : tolérance de fabrication,  
TSF : tolérance de surépaisseur de finition,  
T : vecteur tangent,

## LISTE DES ABREVIATIONS

APT: Automatic Programming Tools.  
CAO : Conception Assistée par Ordinateur.  
CFAO : Conception et Fabrication Assistée par Ordinateur.  
CL-File: Cutter Localization File.  
DCN : Directeur de Commande Numérique.  
DNC: Direct Numerically Control.  
MOCN : Machine Outil à Commande Numérique.  
PADL2 : « Part and Assembly Description Language » Logiciel de CAO volumique (CSG) développé à l'université de Rochester, USA.

## RESUME

Après un bref aperçu sur les fondements de modélisation de courbes et surfaces en CFAO, ce travail présente d'abord une synthèse bibliographique approfondie des méthodes et techniques d'usinage 2/3 axes et situe l'état de l'art.

Ensuite, un système d'usinage 2/3 Axes de surfaces gauches de modèles et de degrés quelconques, composé de trois modules, est développé. Ce système se caractérise en outre par la génération de trajectoires d'outils optimisées sous contrainte des tolérances de CFAO prescrites.

En effet, l'intégration au système proposé d'un module de conversion de surfaces de Bézier de degré élevé en modèle équivalent de degré plus faible et des surfaces NURBS vers les modèles de Bézier, est une fonctionnalité appréciable qui le rend compatible à tous systèmes de CAO surfaciques.

La méthode d'usinage d'ébauche, basée sur une nouvelle stratégie de planification combinant des topologies différentes, permet l'optimisation des trajectoires d'outils en 2 axes  $\frac{1}{2}$  pour l'usinage de surfaces composites et l'opportunité d'élimination de la passe de demi-finition. L'algorithme de calcul des courbes d'intersections surface/plan assurant la robustesse dans la génération des trajectoires fait appel à une technique de discrétisation auto-adaptative de la surface en facettes.

En finition, Le système d'usinage génère des trajectoires par plans parallèles qui garantissent le respect des tolérances de CFAO prescrites avec l'élimination systématique des interférences et la possibilité de reprises locales de l'usinage.

Le système proposé a été implémenté en langage C++, sous l'environnement de développement Microsoft Visual C++, et intégré au logiciel de CAO « AutoCAD V14 » et validé par une étude comparative de résultats par rapport au logiciel de CFAO « Surfcam ».

*Mots clefs : Trajectoires d'outils, tolérances CFAO, NURBS, Bézier, intersections, discrétisation, interférences.*

## ABSTRACT

The study of fundamentals of curve and surface modeling in computer aided design and manufacturing (CAD/CAM), and a thorough research on 2 and 3 axis machining methods and techniques, led us to state of the main problems.

We have developed a 2/3 axis machining for free form surfaces of arbitrary polynomial degree and mathematical model. The technique generates optimal constrained tool paths, and takes design (Chordal deviation) and manufacturing (scallop high) tolerances into consideration.

A conversion module is supplied, which is a helpful tool for designers in surface data exchange, from high degree polynomial Bézier to lower, or from polynomial NURBS to polynomial Bézier. The rough machining method is based on the use of combined topologies to focuses on the optimization of 2 axis1/2 tool paths for composite surface machining and offer the opportunity to eliminate the semi finishing stage.

The surface/plan intersection computing algorithm results in curves that are efficient tool paths the method is founded on a auto-adaptive technique for partitioning the surface into facets.

The machining finishing method gives tool paths by using the “parallel planes” procedure. This ensures the prescribed design and manufacturing allowances, and avoids interference and machining of uncut local region.

The supplied system was worked up in object oriented C++, using Microsoft Visual C++, and was integrated into the “AutoCAD R14” software. Comparing the results with these obtained with “SurfCAM” tested the system’s efficiency.

Keywords: Tool path, CAD/CAM tolerances, NURBS, Bézier, intersection, interference, discretisation.



## ملخص

بعد التذكير الموجز للمبادئ الأساسية في مجال رياضيات المنحنيات والأسطح المستعملة في مجال التصميم و الصنع بواسطة الحاسوب، تطرقنا أولاً إلى بحث وثائقي معمق حول طرق حساب مسار الأدوات.

ثم إنطلاقاً من نتائج البحث الوثائقي، قمنا بإقتراح طريقة جديدة لتخطيط و حساب مسار الأدوات. لغرض تطبيق هذه الطريقة قمنا بإنجاز برنامج حاسوب متكون من ثلاث أجزاء، باستطاعته أن يقوم بإنتاج مسار الأدوات للتحكم في آلة ذات محورين ونصف لصناعة أشكال مصممة بواسطة أسطح من نماذج و درجات مختلفة و ذلك بفعالية عالية.

بالفعل فإن جزء تحويل الأسطح المكونة للنمردج إلى أسطح من نوع Bézier ذات الدرجة المنخفضة يمكن من إستعمال هذا البرنامج مع أي برنامج للتصميم بواسطة الحاسوب.

الطريقة الجديدة المقترحة تسمح بتحسين المردودية و ذلك بناءً على إستعمال مسارات الأدوات من أشكال مختلفة. فعالية حساب منحنيات تقاطع الأسطح المبنى على أساس التقسيم ذات طابع التكيف الذاتي يول للطريقة مستوى عالي من الإستقرار .

لقد تم إنجاز هذا البرنامج بواسطة ورشة صناعة برامج الحاسوب ميكروسوفت

C++ و تم إدماجه إلى برنامج التصميم « AUTOCAD 14 » باستعمال أدوات « ARX » .

في الأخير قمنا بمقارنة للنتائج المتحصل عليها مع نتائج نفس التطبيقات بواسطة برنامج " SurfCAM " الصناعي وسجلنا نتائج جد مشجعة على مستوى أوقات الصنيع و كذا

حجم برامج التحكم « ISO » .

كلمات أساسية :

مسار الأدوات، أسطح Bézier، التقسيم، تداخل، تسامح، تقاطع الأسطح.

# *Introduction*

## Introduction

La conception et fabrication assistées par ordinateur ou CFAO occupe une place importante en productique. Sa mise en œuvre permet d'améliorer la productivité et de couvrir des plages de produits très variés et complexes, réservés autrefois à des techniques spécifiques coûteuses qui n'étaient pas à la portée de tous, à l'exemple des matrices de forge, de fonderie, d'emboutissage et plus généralement de toutes les pièces comportant des formes libres nécessitant la réalisation de gabarits ou de maquettes. La CFAO a permis également d'introduire des concepts nouveaux tels que l'ingénierie simultanée, la modélisation paramétrique et variationnelle mais aussi et surtout le travail de groupe. A titre d'exemple, quelque 4000 ingénieurs de spécialités différentes, implantés respectivement aux Etats Unis d'Amérique, en Europe et au Japon ont contribué à la conception et réalisation du BOEING 777, composé de trois millions de références. Ces concepts sont des outils puissants qui permettent aux utilisateurs de revaloriser davantage le savoir-faire et de mieux se positionner sur le marché. Les niveaux de qualités exceptionnelles des produits que nous utilisons quotidiennement tels que les véhicules personnels, les matériels domestiques (électroménager, Hi-Fi, vidéo, etc.) en sont la preuve.

L'avènement des Machines-outils à commande numérique au début des années cinquante a créé un besoin de développement de systèmes de pilotage de ces machines. Les premiers systèmes de Fabrication Assistée par ordinateur, capables de générer automatiquement les trajectoires d'outil, étaient restreints à l'usinage de formes relativement simples et présentaient des limitations au niveau de la gestion des interférences, du contrôle des collisions, du contrôle de la qualité et de la productivité, en raison des capacités limitées des moyens de calculs disponibles à cette époque. Avec l'évolution extraordinaire de l'informatique et du génie logiciel, les systèmes n'ont cessé de gagner en puissance de traitement et en degré d'automatisation. Ces développements portent sur deux niveaux distincts de l'usinage. En ébauche, le critère prépondérant étant la productivité, les améliorations se sont beaucoup plus orientées vers l'optimisation des trajectoires d'outil. Par contre en finition, le premier critère à respecter étant la qualité, les développements sont orientés beaucoup plus sur la maîtrise des tolérances de conception et de fabrication des surfaces mécaniques, ce qui justifie le nombre important de travaux de recherches publiés durant ces deux dernières décennies dans ce domaine.

Dans le contexte de notre présente recherche, nous nous intéressons à cet aspect fondamental de la CFAO, à savoir la génération automatique et l'optimisation des trajectoires d'outils pour les opérations d'ébauche et de finition pour l'usinage des formes complexes sur des machines outils à commande numérique 3Axes. A cet effet, nous avons structuré ce document en deux parties, organisées comme suit :

Le premier chapitre de la première partie est dédié à l'introduction des fondements mathématiques des modèles géométriques les plus couramment utilisés en CFAO. Une synthèse bibliographique détaillée des méthodes d'usinage de surfaces gauches fait l'objet d'un deuxième chapitre, afin de pouvoir faire le point sur l'état de l'art et de cerner les problèmes posés dans ce domaine. Cela nous a permis de mettre en évidence deux aspects bien distincts inhérents à la génération des trajectoires d'outils. Le premier aspect concerne la planification des trajectoires d'outils qui met en œuvre différentes topologies de trajectoires et différentes techniques de planification, le deuxième étant le calcul proprement dit des points de trajectoires d'outil correspondant à la phase de mise en œuvre de la stratégie d'usinage adoptée.

Le premier chapitre de la deuxième partie présente le système de CFAO proposé pour la génération et l'optimisation des trajectoires d'outils en usinage d'ébauche et finition de surface en 3Axes. La notion de construction flexible, basées sur les méthodes de transformation entre modèles B-spline/Bézier et sur les méthodes de conversion de degré des modèles de Bézier est présentée de façon succincte dans un premier temps. Nous introduisons ensuite les méthodes d'usinage d'ébauche et de finition. D'une part, la méthode d'ébauche développée permet l'optimisation des trajectoires d'outils en 2 axes  $\frac{1}{2}$  pour l'usinage de surfaces gauches composites. En effet, un algorithme de calcul des courbes d'intersections surface/plan, basé sur une technique de discrétisation auto-adaptative de la surface en facettes sous contraintes du respect de tolérances imposées par l'utilisateur, est développé et assure une génération robuste de trajectoires. Une nouvelle approche d'évidement des poches qui élimine la passe de demi-finition et optimise les déplacements à vide est également introduite. D'autre part, la méthode d'usinage de finition proposée permet la génération de trajectoire d'usinage en 3Axes garantissant le respect des tolérances de CFAO et l'élimination systématique des interférences, avec la possibilité de reprises locales d'usinage éventuelles.

Le deuxième chapitre de la deuxième partie est consacré à la mise en œuvre informatique du système de CFAO proposé. Les différents algorithmes constituant les modules de conversions, de génération des trajectoires d'ébauche et de calcul des trajectoires de finition ainsi que les structures de données développées sont détaillés et illustrés par des exemples d'exécutions. Une étude comparative des performances du système proposé avec un logiciel de CFAO industrielle a été réalisée et nous a permis de valider ce travail.

Une conclusion générale clôture ce présent mémoire en reprenant de façon succincte la synthèse des résultats obtenus et en présentant les perspectives d'extension des développements algorithmiques présentés et d'exploitation des résultats obtenus dans le cadre d'une intégration au niveau industriel.

## *1<sup>er</sup> PARTIE*

*Modélisation géométrique et usinage des surfaces gauches.*

## Chapitre I.1 Modélisation géométrique

### 1.1 Introduction

La conception de formes complexes, particulièrement utilisées en aéronautique, et en carrosserie d'automobile, a conduit au développement de modèles mathématiques très variés. L'avènement de l'informatique et de la commande numérique a permis de concevoir et de réaliser des pièces aux formes les plus diverses. De nos jours, tous les systèmes de CFAO utilisent les modèles polynomiaux, et permettent de modéliser les formes les plus complexes.

En effet, les avantages offerts par les formes polynomiales, en terme de diversité des formes modélisables, de simplicité des traitements et manipulations et de rapidité de construction, leur confèrent la préférence par rapport aux autres formes mathématiques, telles que les fonctions trigonométriques, exponentielles, logarithmiques ou autres.

Les modèles implémentés dans la grande majorité des systèmes de C.F.A.O utilisent les formes polynomiales paramétriques et ne diffèrent de se fait que dans la base de polynômes utilisés. On retrouve ainsi les polynômes de Bernstein pour le modèle de Bézier, les fonctions splines pour le modèle B-spline, et les polynômes d'Hermite pour le modèle de Coons. L'utilisation de ces modèles sous leur forme rationnelle permet d'en étendre considérablement les propriétés [Léon '91], [Bensalah '90]. En effet, sous cette forme, ils permettent une représentation mathématique unique pour toutes les formes géométriques standard telles que les lignes, les coniques, les arcs et cercles, les plans ainsi que les formes libres de courbes et de surfaces. De plus, ils offrent un grand degré de liberté permettant de générer une large variété de formes, tout en vérifiant la propriété de l'invariance affine.

Les paragraphes qui suivent présentent d'une façon synthétique les modèles les plus utilisés en CFAO. Nous verrons ainsi les modèles intégraux ensuite les modèles continus par morceaux et cela dans les représentations non rationnelles et rationnelles.

## 1.2 Classification des modèles

### 1.2.1 Modèle intégral de Bézier

#### 1.2.1.1 Courbe de Bézier non rationnelle

Une courbe de Bézier est une forme polynomiale à base de polynômes de Bernstein.

La  $i^{\text{eme}}$  fonction Bernstein de degré  $n$ , notée  $B_{i,n}(t)$ , étant définie par :

$$B_{i,n}(t) = \binom{n}{i} \cdot (1-t)^{(n-i)} \cdot t^i \quad \text{avec} \quad \binom{n}{i} = \frac{n!}{i! \cdot (n-i)!}$$

Une courbe de Bézier non rationnelle  $C(t)$ , de degré  $n$ , dont les  $P_i$  sont les pôles (Figure 1.1), sera définie par :

$$C(t) = \sum_{i=0}^n B_{i,n}(t) P_i \quad 0 \leq t \leq 1$$

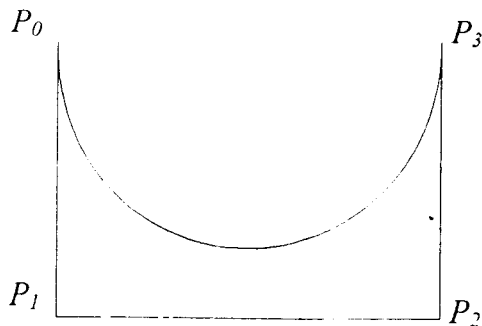


Figure 1.1 Courbe de Bézier non rationnelle cubique.

De part les propriétés des fonctions de base utilisées, le modèle de Bézier non rationnel possède les caractéristiques géométriques principales suivantes:

- Propriétés des frontières : La courbe passe par le premier et le dernier pôle tout en étant tangente au polygone.
- Invariance affine : Une transformation affine est appliquée à la courbe si elle est appliquée aux pôles.
- Enveloppe convexe : La courbe de Bézier est toujours à l'intérieur du polygone convexe formé par les pôles.
- Comportement global : Le déplacement d'un pôle change la forme globale de toute la courbe. Le polygone formé par les  $P_i$  représente une approximation globale de la courbe.

- Propriété de la variation décroissante: Aucun plan n'a plus d'intersections avec la courbe qu'avec le polygone de contrôle.
- Propriété de symétrie: Lorsque nous changeons  $t$  en  $t-1$ , la courbe est parcourue dans le sens inverse.

### 1.2.1.2 Surface de Bézier non rationnelle

Une surface bi-paramétrique de Bézier non rationnelle de degré  $m$  suivant le premier paramètre noté  $u$ , et  $n$  suivant le second paramètre noté  $v$ , dont les  $P_{ij}$  représentent le réseau de contrôle, est définie par:

$$S(u, v) = \sum \sum B_{i,m}(u) B_{j,n}(v) P_{ij} \quad 0 \leq u \leq 1 \text{ et } 0 \leq v \leq 1$$

Les fonctions de base de Bernstein étant inchangées, les propriétés des courbes citées ci dessus restent valables.

### 1.2.1.3 Courbes de Bézier rationnelles

La définition du modèle rationnel repose sur le concept de coordonnées homogènes [Léon'91], [de Casteljau'85]. L'application projective notée  $H$  représente la relation de passage entre les points de  $R^4$  et  $R^3$  pour les courbes spatiales, entre les points de  $R^3$  et  $R^2$  pour les courbes planes.

$$H\{(x, y, z, w)\} = \begin{cases} \left( \frac{x}{w}, \frac{y}{w}, \frac{z}{w} \right) & \text{si } w \neq 0 \\ \text{un point à l'infini sur la ligne} \\ \text{partant vers le point } (x, y, z) \end{cases}$$

Une courbe de Bézier rationnelle  $C^h(t)$ , de degré  $n$ , dont les  $P_i^h$  sont les pôles dans  $R^4$ , sera définie, de la même façon que pour le cas non rationnel, par :

$$C^h(t) = \sum_{i=0}^n B_{i,n}(t) P_i^h \quad 0 \leq t \leq 1$$

La courbe associée dans  $R^3$  sera obtenue en appliquant la transformation  $H$ .

$$C(t) = H(C^h(t)) = H\left( \sum_{i=0}^n B_{i,n}(t) P_i^h \right) \quad 0 \leq t \leq 1$$

Le polygone  $P_n^h$  étant remplacé par  $P_n$ , dont les sommets auront pour coordonnées dans  $R^4$ :



$$P_i^h = (wP)_i = \begin{cases} w_i P_i \cdot x \\ w_i P_i \cdot y \\ w_i P_i \cdot z \\ w_i \end{cases} \quad i = 0..n$$

L'expression de  $C(t)$  dans  $R^3$  sera donc donnée par :

$$C(t) = \frac{\sum_{i=0}^n B_{i,n}(t) w_i P_i}{\sum_{j=0}^n B_{j,n}(t) w_j} \quad 0 \leq t \leq 1$$

Les propriétés des courbes de Bézier rationnelles sont semblables à celles des courbes non rationnelles. De plus, les poids  $w_i$  procurent un paramètre de contrôle supplémentaire de la forme de la courbe. En effet, si  $w_i$  croît la courbe est attirée d'avantage vers le pôle  $P_i$ , elle s'en éloigne dans le cas contraire (figure 1.2). La propriété de l'enveloppe convexe se trouve donc conditionnée par les valeurs des  $w_i$ , et ne reste valable que si ces derniers sont tous positifs [Léon'91].

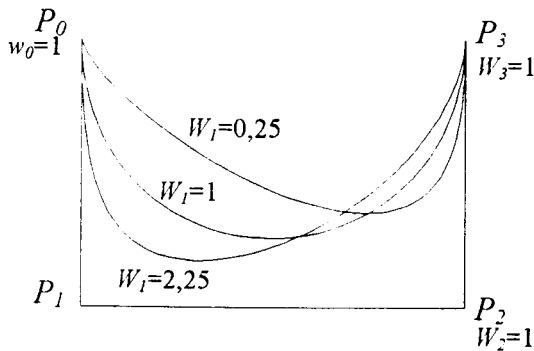


Figure 1.2 Influence des poids  $w_i$  sur la forme d'une courbe de Bézier rationnelle.

#### 1.2.1.4 Surfaces de Bézier rationnelles

Une surface bi-paramétrique de Bézier rationnelle  $S(u, v)$ , de degré  $m$  en  $u$  et  $n$  en  $v$ , dont les  $P_{ij}$  représentent le réseau de contrôle, est définie par:

$$S(u, v) = H(S^h(u, v)) = H \left\{ \sum_{i=0}^m \sum_{j=0}^n B_{i,m}(u) B_{j,n}(v) P_{ij}^h \right\}$$

soit :

$$S(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n B_{i,m}(u) B_{j,n}(v) w_{ij} P_{ij}}{\sum_{i=0}^m \sum_{j=0}^n B_{i,m}(u) B_{j,n}(v) w_{ij}}$$

Les fonctions de base étant inchangées, les propriétés des courbes citées ci-dessus restent valables.

## 1.2.2 Modèle par morceau B-Spline

### 1.2.2.1 Fonction de base B-spline

Etant donnée une séquence croissante de nombres réels  $T = \{t_0, \dots, t_i, t_{i+1}, \dots, t_m\}$ , la  $i^{\text{ème}}$  fonction B-spline de degré  $p$  notée  $N_{i,p}(t)$ , est définie par:

$$N_{i,0}(t) = \begin{cases} 1 & \text{si } t_i \leq t \leq t_{i+1} \\ 0 & \text{sinon} \end{cases}$$

$$N_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+1+p} - t}{t_{i+1+p} - t_{i+1}} N_{i+1,p-1}(t)$$

la convention  $\frac{0}{0} = 0$  est adoptée.

Les  $t_i$  sont appelés "nœuds" et forment la "séquence nodale" représentée par le "vecteur nœuds" «  $T$  ».

Les fonctions  $N_{i,p}(t)$  ont les propriétés importantes suivantes:

- Non négativité:  $N_{i,p}(t) \geq 0$  quels que soient  $i$ ,  $p$  et  $t$ .
- Partition unité:  $\sum_{i=0}^n N_{i,p}(t) = 1 \forall t \in [t_0, t_m]$ ,  $n$  est le nombre total de fonctions B-spline de degré  $p$  pour un vecteur nœud donné,  $n$  étant lié à  $p$  et  $m$ .
- Support local:  $N_{i,p}(t) = 0$  si  $t$  est à l'extérieur de l'intervalle  $[t_i, t_{i+p+1}]$ .
- Dérivabilité: Toutes les dérivées des fonctions B-spline existent. Pour un nœud de multiplicité  $k$ , elles sont  $(p-k)$  continûment dérivable.
- Extremum: Excepté le cas  $p=0$ , les fonctions B-spline ont un seul maximum.

La forme de la courbe B-Spline étant directement liée au choix de la séquence nodale, nous distinguerons dans la pratique :

- Les B-Spline non périodiques : le premier et le dernier nœud ayant une multiplicité de  $p+1$ , soit  $t_0 = t_1 = \dots = t_p$  et  $t_{m-p} = t_{m-p+1} = \dots = t_m$ .
- Les B-Spline uniformes : Pour lesquelles les valeurs nodales vérifient :  $t_{i+1} = t_i + d$  pour  $p \leq i \leq m-p-1$ ,  $d > 0$ .
- Les B-Spline non uniformes : L'espacement des nœuds étant quelconque. La relation précédente n'est plus vérifiée.

### 1.2.2.2 Courbes B-spline non rationnelles

Une courbe B-spline non rationnelle  $C(t)$ , de polygone caractéristique  $P_i$ , de séquence nodale  $T$  et de degré  $p$  est définie par:

$$C(t) = \sum_{i=0}^n N_{i,p}(t)P_i, \quad 0 \leq t \leq 1$$

Le nombre de nœuds  $m$ , le degré  $p$ , et le nombre de pôles  $n$  sont liés par la relation:  $m=n+p+1$ .

Les caractéristiques des fonctions de base Spline procurent aux courbes B-spline des propriétés géométriques très intéressantes, qui justifient leur large utilisation dans les systèmes de CFAO. Parmi ces propriétés nous citerons :

- Une multiplicité de  $(p+1)$  des nœuds extrêmes engendre des conditions aux extrémités de la courbe similaires à celles des courbes de Bézier.
- Invariance affine: Une transformation affine est appliquée à la courbe si elle est appliquée aux pôles.
- Enveloppe convexe locale: La courbe est contenue dans une enveloppe convexe des pôles, c'est-à-dire si  $t \in [t_j, t_{j+1}]$  et  $p \leq j < m-p-1$ , alors  $C(t)$  est dans l'enveloppe convexe des pôles  $P_{j-p}, P_j$ .
- Le polygone formé par les pôles  $P_j$  représente une approximation linéaire par morceaux de la courbe.
- Comportement local : Si on déplace un pôle, la courbe sera localement déformée et affecte la courbe dans  $p+1$  intervalles.
- Phénomène commutateur : En se déplaçant le long d'une courbe de  $t=0$  à  $t=1$  les fonctions B-spline agissent comme un commutateur, d'un intervalle à l'autre, en passant par un nœud une fonction « s'éteint » une autre « s'allume ».
- A l'intérieur d'un intervalle non nul  $[t_i, t_{i+1}]$  toutes les dérivées existent, au niveau d'un nœud de multiplicité  $k$ , elle est  $p-k$  fois continûment dérivable.
- Propriété de la variance décroissante : Aucun plan n'a plus d'intersections avec la courbe qu'avec le polygone de contrôle.
- Type Bézier: Une courbe B-spline sans nœuds internes est identique à la courbe de Bézier de même polygone de contrôle.

### 1.2.2.3 Surfaces B-spline non rationnelles

Une surface bi-paramétrique B-Spline non rationnelle de degré  $p$  suivant le premier paramètre noté  $u$ , et  $q$  suivant le second paramètre noté  $v$ , dont les  $P_{ij}$  représentent le réseau de contrôle, est définie par:

$$S(u, v) = \sum_{i=0}^m \sum_{j=0}^n N_{i,p}(u) N_{j,q}(v) P_{ij}$$

Les vecteurs  $U = \{u_0, \dots, u_i, u_{i+1}, \dots, u_m\}$  et  $V = \{v_0, \dots, v_j, v_{j+1}, \dots, v_n\}$  forment les vecteurs nœuds non uniformes et non périodiques relatifs respectivement aux fonctions  $N_{i,p}(u)$  et  $N_{j,q}(v)$ .

Excepté la propriété de la variation décroissante les propriétés des fonctions surfaciques de base B-spline non rationnelles sont analogues à celles des courbes, les fonctions de base étant inchangées.

### 1.2.2.4 Courbes B-spline rationnelles

De la même façon que pour les courbes de Béziérs, une courbe B-Spline rationnelle  $C(t)$ , de  $R^3$  sera définie comme étant la transformée de la courbe B-Spline non rationnelle  $C^h(t)$ , de  $R^d$  par la projection  $H$  précédemment définie.

$$C(t) = H\{C^h(t)\} = H\left\{\sum_{i=0}^n N_{i,p}(t) P_i^h\right\} = \frac{\sum_{i=0}^n N_{i,p}(t) w_i P_i}{\sum_{j=0}^n N_{j,p}(t) w_j}$$

En plus des propriétés des courbes B-Spline non rationnelles, qui restent vérifiées, les courbes B-Spline rationnelles présentent un degré supplémentaire en termes de souplesses de modélisation grâce à l'influence des poids  $w_i$  sur la forme de la courbe. En effet les  $w_i$  permettent une modification locale de la courbe. Le changement de la valeur d'un poids  $w_i$  n'affectera la courbe que pour  $t \in [t_i, t_{i+p+1}]$ , et cela de façon prévisible et quantifiable [Léon'91], [Bensalah'90].

### 1.2.2.5 Surfaces B-spline rationnelles

Une surface bi-paramétrique B-spline rationnelles  $S(u, v)$  de degré  $p$  en  $u$ , et  $q$  en  $v$ , dont les  $P_{ij}$  représentent le réseau de contrôle, est définie par:

$$S(u, v) = H\{S^h(u, v)\} = H\left\{\sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) P_{ij}\right\}$$

$$S(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n N_{i,p}(u) N_{j,q}(v) w_{ij} P_{ij}}{\sum_{i=0}^m \sum_{j=0}^n N_{i,p}(u) N_{j,q}(v) w_{ij}}$$

Les propriétés géométriques des surfaces B-spline rationnelles sont analogues à celles des surfaces B-spline non rationnelles.

### 1.2.2.6 Courbes et surfaces NURBS

Le modèle NURBS, pour « Non Uniform Rational B-Spline », n'est autre que le modèle B-Spline rationnel à séquence nodale non uniforme (Cf. 1.2.2.1), il représente de se fait une généralisation des modèles précédents. En effet une courbe, respectivement une surface de Bézier ou B-Spline peut se représenter comme une courbe, respectivement une surface NURBS particulière [Leon '91].

## 1.3 Conclusion

Par rapport aux modèles B-splines [Léon '91], [Farin '93], [Boehm '81, '87], [Bensalah '90] dont les propriétés géométriques de modélisation sont très appréciées en raison de leur flexibilité dans les manipulations géométriques et leur capacité de représentation exacte des coniques, le modèle de Bézier, bien qu'implémenté dans de nombreux systèmes de CFAO et en dépit des très intéressantes propriétés de modélisation géométrique qu'il offre [Bézier '77, '85, '88], [Léon '91], [Farin '93], [Bensalah '90], [De Casteljaou '85], impose pour la modélisation de formes complexes l'utilisation de degré élevé. Ce qui engendre des problèmes numériques [Belaidi '98], [Nougier '93], [Léon '91] inhérent aux formes polynomiales à hauts degrés.

La mise en œuvre de techniques de sub-réduction [Bensalah '90, '93], [Hoček '88, '89, '90], [Eck '93, '95], [Belaidi '98], [Asma '98] représente une solution au problème (Cf. II.1.2) et permet de disposer à la fois de la souplesse de modélisation des B-Spline lors de la conceptions des formes, et de la simplicité et efficacité de mise en œuvre du modèle de Bézier à faibles degrés pour les phases de traitements et calculs, plus particulièrement lors de la phase génération des trajectoires d'outils.

## Chapitre I.2 Planification et calcul des trajectoires d'outil

### 2.1 Introduction

L'évolution des systèmes de FAO a été pendant longtemps et est jusqu'à présent sous-tendu par le souci de rentabilité maximale ; elle vise essentiellement l'automatisation totale de la génération des trajectoires et l'intégration CAO-FAO. Ces dernières années, beaucoup de travaux de recherche portant sur l'amélioration des performances des systèmes de génération des trajectoires d'outils ont été publiés [Prabhu et al. '90], [Lee et al. '91], [Lakkaraju et al. '92], [Whang '92], [Oliver et al. '93], [Held et al. '94], [Marshall et al. '94a, '94b], [Suresh et al. '94], [Sun et al. '94], [Cox et al. '94], [Kim et al. '95].

Cela a conduit à la mise au point de logiciels relativement spécifiques car leur utilisation impose des hypothèses de départ assez contraignantes à respecter tels que les limitations de degré, de modèle, des restrictions des formes géométriques, de la géométrie d'outils, etc., qui limite ainsi leur champ d'utilisation.

Outre l'automatisation de la génération des trajectoires d'outils, les systèmes de FAO doivent satisfaire d'autres critères non moins importants et conditionnant la productivité globale [Marshall et al. '94a, '94b], [Dahmane '93] à savoir qu'ils doivent :

- être applicables à diverses formes : pour n'imposer qu'un minimum de restrictions quant aux formes des pièces pouvant être traitées ;
- être applicables à divers modèles de surfaces : en vue de pouvoir traiter des modèles de pièces réalisés avec différents modèles mathématiques ;
- garantir le respect d'une qualité imposée et la validité de la pièce: ce qui implique la prise en compte de l'erreur de corde, de la hauteur de crête [Belaidi '98], ainsi que la gestion des interférences ;
- contrôler l'efficacité de l'usinage : contrôle des erreurs de trajectoires (chevauchement, écartement, discontinuité, recouvrement) ; usinage à la tolérance la plus large possible compatible avec la qualité imposée ; minimisation des temps improductifs ; recherche de chemins d'outils optimisés. [Sun et Lee '94], [Dahmane '93] [Kim et al. '95], [Prasad '90], [Ravi '92].

Ces critères de performances nous permettront par la suite d'analyser les différentes approches adoptées et de proposer une méthode au vu de cette analyse.

## 2.2 Planification des trajectoires

### 2.2.1 Classification

Les différentes topologies de trajectoires utilisées pour l'usinage des formes complexes peuvent se résumer en cinq types [Marshall et al. '94a, '94b], [Tan et al. '90]

#### 2.2.1.1 Trajectoires en spirales par décalage de contours

Ces trajectoires sont construites par décalage successif des contours (Figure 2.1a) pour former des chemins d'outil concentriques (Figure 2.1b). Ce type de trajectoires est surtout utilisé dans l'usinage des poches [Person '78], [Suh et Lee '90].

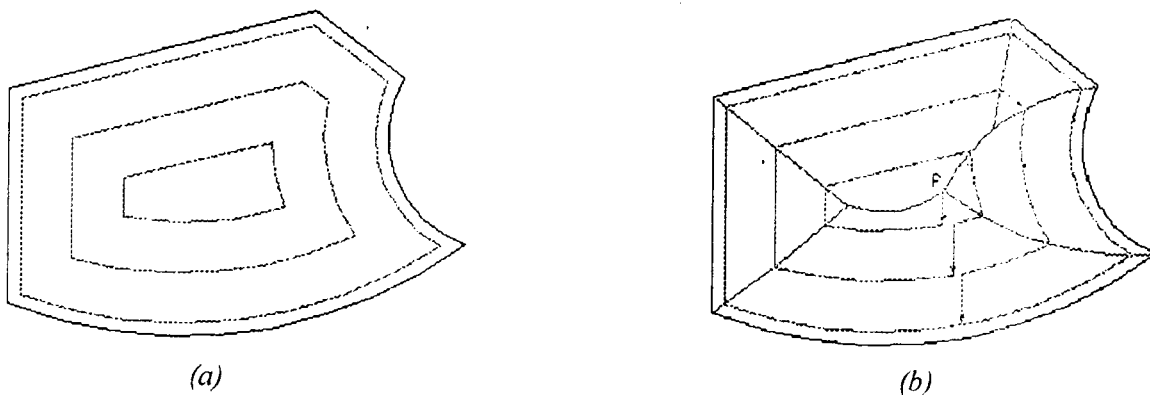


Figure 2.1 Trajectoires en spirales par décalage de contours

#### 2.2.1.2 Trajectoires en spirales par réduction de contour

Les trajectoires en spirale par réduction de contours sont construites à l'aide d'une transformation homothétique progressive des contours (Figure 2.2) en s'éloignant du contour final [Festenberg '88].

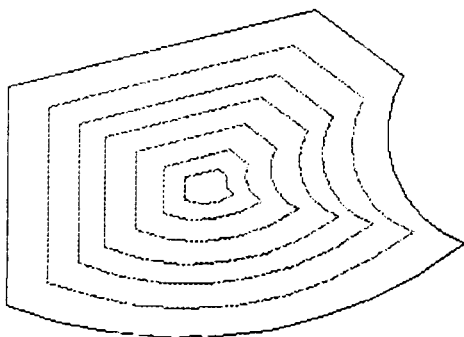


Figure 2.2  
Trajectoires en spirales par réduction de contour

### 2.2.1.3 Trajectoires en zigzag parallèles

Les chemins d'outil sont formés pour balayer la pièce suivant des trajectoires parallèles (figure 2.3). Ce type de trajectoires est applicable à des formes générales [Choi et al. '88], [Parabhu et al. '90].

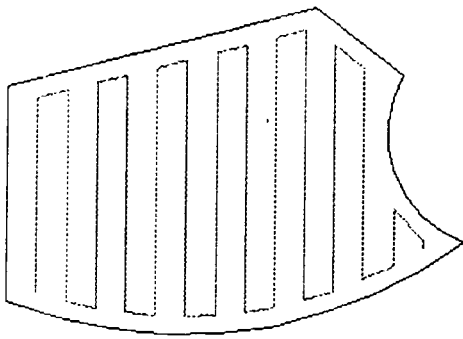


Figure 2.3  
Trajectoires en zigzag  
parallèles

### 2.2.1.4 Trajectoires en zigzag iso-paramétriques

Les trajectoires en zigzag iso-paramétriques sont obtenues en faisant évoluer l'outil le long de courbes iso-paramétriques sur les carreaux de surfaces paramétriques à usiner (figure 2.4). Ce type de trajectoires est applicable à des formes générales [Broumhead et Edkins '86], [Chou '89]

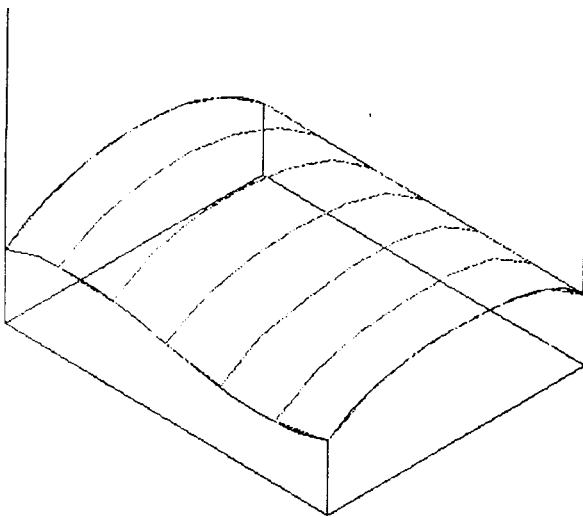


Figure 2.4  
Trajectoires en zigzag  
iso-paramétriques

### 2.2.1.5 Trajectoires unidirectionnelles iso-paramétriques

Ce type de trajectoires est similaire aux précédentes, sauf que dans le cas des trajectoires en zigzag iso-paramétrique, l'outil travaille tantôt en avalant tantôt en



opposition, alors que dans le cas des trajectoires unidirectionnelles, l'outil travaille toujours dans le mode choisi par l'utilisateur, en opposition ou en avalant ; permettant ainsi un usinage plus régulier [Broumhead et Edkins '86], [Chou '89].

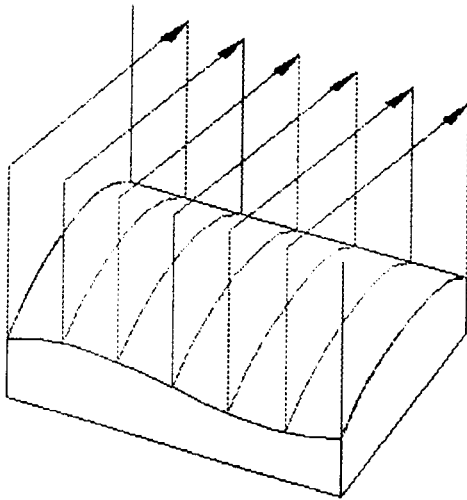


Figure 2.5  
Trajectoires unidirectionnelles  
iso-paramétriques

#### 2.2.1.6 Trajectoires unidirectionnelles parallèles

Ce type de trajectoires est similaire aux trajectoires unidirectionnelles iso-paramétriques, sur le plan des conditions d'usinage. Dans le premier cas les chemins d'outils sont parallèles dans l'espace paramétrique, alors que dans ce cas ils le sont dans l'espace euclidien (figure 2.6).

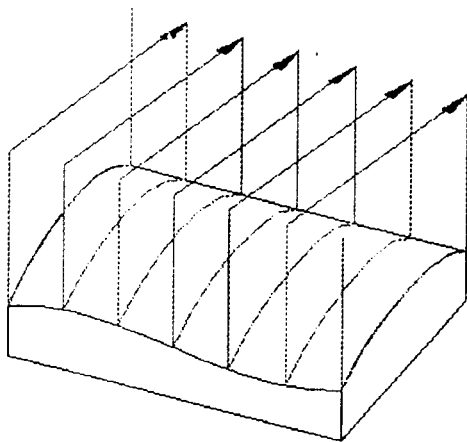


Figure 2.6  
Trajectoires unidirectionnelles  
parallèles

#### 2.2.1.7 Courbes de remplissage

Ce type de trajectoires est moins connu que les précédents, il a fait l'objet de recherches récentes [Marshall et al. '94a], [Cox et al. '94]. Les chemins d'outils successifs forment une courbe de remplissage de la surface à usiner (figure 2.7).

Ces trajectoires ont été développés en vue de résoudre des problèmes de chevauchement des trajectoires et d'usinage à tolérance trop serrée ; ce type de

trajectoires présente un caractère d'auto adaptabilité très intéressant et mérite des approfondissements.

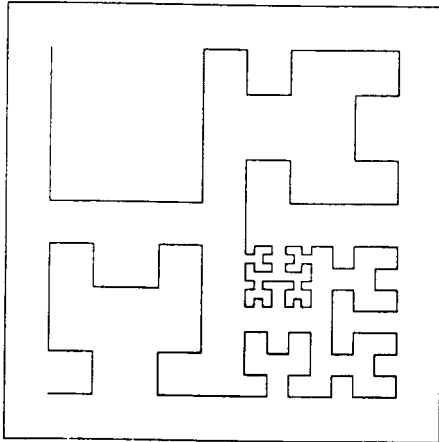


Figure 2.7  
Trajectoires par courbes de remplissage

## 2.2.2 Problèmes posés

### 2.2.2.1 La hauteur de crête

L'usinage de surfaces complexes sur MOCN est caractérisé par un surplus de matière laissé par l'outil entre les différents parcours de balayage et ayant une forme de crêtes entre deux sillons successifs taillés par l'arête de l'outil (figure 2.8). Ces crêtes sont généralement éliminées, lorsque le degré de finition imposé l'exige, soit par un polissage manuel, soit par une attaque chimique ou électrochimique. Cette phase prend beaucoup de temps et induit des coûts supplémentaires élevés. Il est donc primordial de contrôler la hauteur de crête en introduisant des méthodes de prédiction de cette dernière dès la phase de planification et de calcul des chemins d'outils.

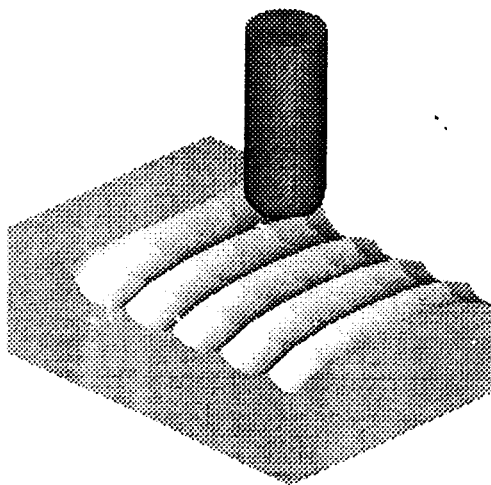


Figure 2.8  
La hauteur de crête et son aspect sur la surface usinée.

La mesure de la hauteur de crête  $h$  (figure 2.9) est une méthode conventionnelle pour déterminer le surplus de matière lors de l'usinage des surfaces gauches ; elle est fréquemment utilisée pour caractériser la qualité des surfaces gauches usinées sur MOCN car elle s'apparente à la hauteur des aspérités de rugosité. Bien que cette méthode soit devenue un standard, elle ne fournit pas toutes les informations nécessaires pour caractériser la qualité des surfaces [Ralph et Loftus '92], [Cho et al. '94] ; RALPH et LOFTUS [Ralph et Loftus '92] proposent une approche plus rigoureuse basée sur l'évaluation de l'aire et du volume des crêtes laissées après l'usinage.

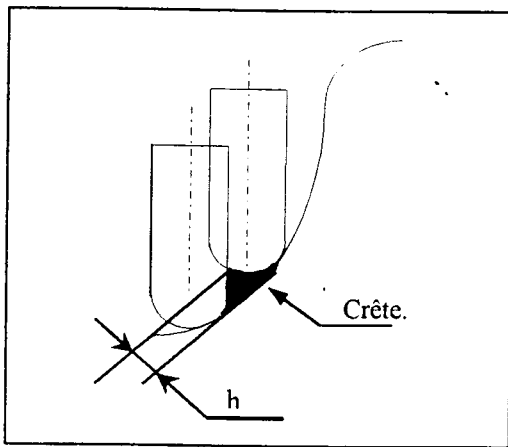


Figure 2.9  
Mesure de la hauteur de crête  $h$ .

## A- Cas de l'usinage 3 axes

### a- Fraise à bout hémisphérique

Une évaluation exacte de la valeur de la hauteur de crête est assez délicate du fait de la variation du rayon de courbure de la surface entre deux positions successives du centre de l'outil (figure 2.10) ; néanmoins, étant donné qu'en pratique la valeur du pas transversal en finition est assez faible ( $l < 2R$  outil), nous pouvons approcher localement la surface par une calotte sphérique [Rubio '93], [Kim et al. '93].

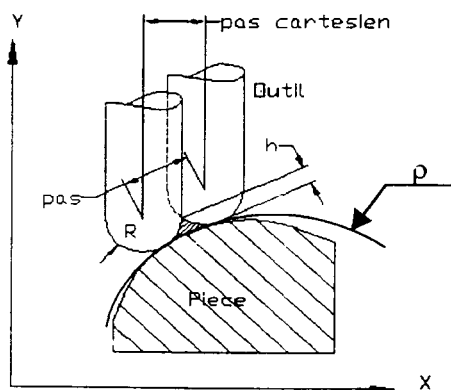


Figure 2.10  
Approximation locale de la surface par une sphère.

Dans ce cas, l'établissement des relations qui lient le pas transversal à la valeur de la hauteur de crête est simplifié. Les détails de calculs sont donnés en annexe I.

#### b- Fraise torique

Dans ce cas, il convient de signaler que dans la majorité des études publiées, les calculs sont établis en supposant que la crête laissée entre deux passes consécutives n'est générée que par la partie torique de l'outil, la partie cylindrique étant exclue. Cette hypothèse n'est pas toujours vérifiée surtout pour des outils dont le rayon  $r$  du tore est faible par rapport à celui de la partie cylindrique,  $R$ . Rubio [Rubio '92], [Rubio '93a], [Rubio '93b] et KIM et al. [Kim et al. '94] abordent l'étude de la hauteur de crête lors de l'usinage à l'outil torique, d'une façon beaucoup plus détaillée et proposent des relations qui tiennent compte de tous les cas de figures possibles. En effet la crête peut être générée par :

- a -les parties toriques des deux fraises,
- b -la partie cylindrique de la première fraise et la partie torique de la deuxième,
- c -la partie torique de la première fraise et la partie plane de la deuxième,
- d -la partie cylindrique de la première fraise et la partie plane de la deuxième,

Par la suite nous parlerons d'intersection sur la partie plane lorsque l'intersection aura lieu sur les parties planes ou cylindriques des fraises, et d'intersection circulaire lorsque l'intersection aura lieu sur la partie torique des fraises. L'annexe I donnent une étude détaillée des différents cas de figures et les relations correspondantes.

### B-Cas de l'usinage 5 axes

Le passage des points de contacts outil/pièce aux points centre d'outil, impose de fixer deux paramètres, puisque la machine outil est dotée de cinq degrés de liberté: Pour un Point de contact outil/pièce il existe une infinité de Point centre d'outils. Pour des considérations technologiques, il convient de garder l'axe de l'outil dans un plan contenant le segment liant le point de contact outil/pièce courant au Point de contact suivant sur la même trajectoire. Le deuxième paramètre que l'on se fixe sera l'angle de d'épînage (angle formé entre le vecteur normal à la surface au CC point considéré et l'axe de l'outil), afin de permettre un débit maximum de copeau. L'angle de dépinçage sera pris aussi faible que possible, sans qu'il y ait interférence avec la pièce à usiner ; en pratique il est de l'ordre de quelques degrés [Choi et al. '93], [Li et Jerard'94].

#### a- Fraise cylindrique

Les relations de calcul de la valeur de la hauteur de crête dans le cas de l'usinage 5 axes à l'outil cylindrique sont données dans les références suivantes : [Choi et al. '93], [Li et Jerard'94], [Ralph et Loftus '92].

### b- Fraise à bout hémisphérique

Dans le cas de l'usinage à l'outil à bout hémisphérique, les angles d'inclinaison relative outil/pièce n'ont pas d'influence sur la hauteur de crête, tant que cette dernière est générée par les parties sphériques de l'outil (Figure 2.11). Les relations développées pour le cas de l'usinage 3 axes sont donc toujours valables.

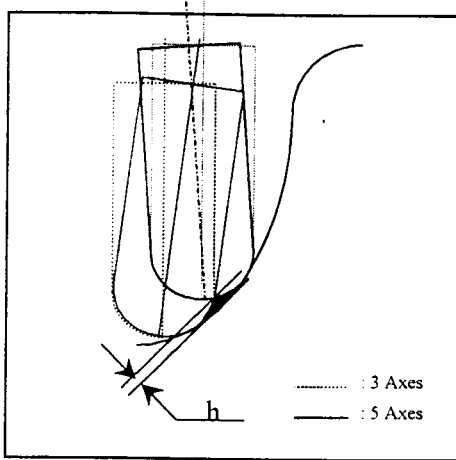


Figure 2.11  
 Invariance de la hauteur de crête entre usinage 3 axes et 5 axes à l'outil hémisphérique.

### c- Fraise à bout torique

Les fraises toriques permettent un débit de copeaux plus important que les fraises à bout hémisphériques et sont de ce fait plus performantes ; néanmoins elles imposent des degrés de liberté supplémentaires à la MOCN (MOCN à 5 axes). L'établissement des relations de calcul de la valeur de la hauteur de crête nécessite des développements qui sortent du cadre de cette étude.

#### 2.2.2.2 L'erreur de corde

Les trajectoires d'usinage des surfaces complexes sur MOCN, sont généralement réalisées grâce à une succession d'interpolations linéaires, plus rarement circulaires. Cette approximation de la courbe par des segments de droite induit une erreur dite erreur de corde  $TC$  (figure 2.12), définie comme étant la distance normale maximale entre les points  $P(u)$  de la portion de courbe à la corde qui la sous tend ( $M_i M_{i+1}$ ).

$$TC = \text{Max} [ \text{distance} (P(u), M_i M_{i+1}) ]$$

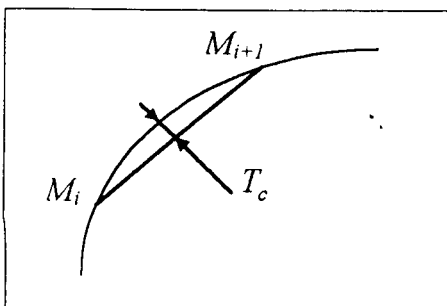


Figure 2.12  
 Erreur de corde et tolérance de conception  $TC$

L'erreur de corde  $TC$  est d'autant plus grande que la surface est courbée et que le pas d'interpolation linéaire est important. Plus le pas est faible, plus la précision est améliorée ; mais cela induit des temps de calcul et des tailles de fichiers plus importants ; les temps d'usinage augmentent aussi, ainsi que l'usure des outils. La maîtrise de cette erreur est donc primordiale pour un système de génération de trajectoires d'outils.

### Calcul de la valeur de $TC$ :

Soit  $c$  le vecteur corde joignant les deux points  $P(u_{min})$  et  $P(u_{max})$ , et  $d(u)$  le vecteur liant un point  $P(u)$  de la portion de courbe ( $P(u), u \in \{u_{min}, u_{max}\}$ ), à sa projection orthogonale sur  $c$  (figure 2.13).

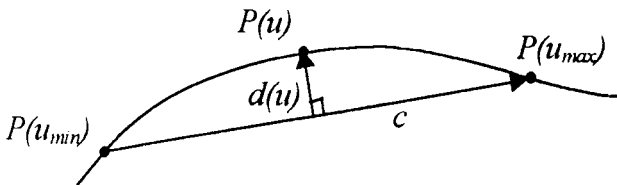


Figure 2.13  
 Calcul de l'erreur de corde.

Nous pouvons écrire :

$$P(u) = P(u_{min}) + d(u) + \lambda c \quad \Rightarrow \quad d(u) = P(u) - P(u_{min}) - \lambda c \quad [2.1]$$

Comme  $c$  est perpendiculaire à  $d(u)$  Alors

$$c \cdot d(u) = 0$$

donc

$$c \cdot d(u) = c \cdot [P(u) - P(u_{min}) - \lambda c] \Rightarrow c \cdot [P(u) - P(u_{min})] - \lambda |c|^2 = 0$$

ce qui nous permet d'avoir l'expression de  $\lambda$

$$\lambda = \frac{[P(u) - P(u_{min})] \cdot c}{|c|^2}$$

si nous remplaçons  $\lambda$  dans l'équation [2.1] nous obtiendrons

$$d(u) = P(u) - P(u_{min}) - \frac{[P(u) - P(u_{min})] \cdot c}{|c|^2} c$$

soit en notation matricielle

$$\{d(u)\} = [M_p] \{P(u) - P(u_{\min})\}$$

$[M_p]$  étant la matrice de projection orthogonale sur  $c$  définie par

$$[M_p] = [I] - \frac{\{c\}\{c\}^T}{\{c\}^T \{c\}}$$

Lorsque la distance  $|d(u)|$  est maximale, le vecteur  $d(u)$  est perpendiculaire au vecteur tangent à la courbe au point correspondant  $P(u)$ , ce qui nous permet d'écrire

$$\{\dot{P}(u)\}^T \{d(u)\} = \{\dot{P}(u)\}^T [M_p] \{P(u) - P(u_{\min})\} = 0$$

qui est une équation polynomiale de degré  $(2n-1)$  en  $u$ . Comme  $d(u)$  s'annule pour les deux valeurs particulières  $u=u_{\min}$  et  $u=u_{\max}$ , il faudra résoudre un polynôme de degré  $(2n-3)$ .

Dans le cas particulier d'une courbe de Bezier quadratique, la solution est unique et vaut ( $u=1/2$ ). Dans le cas général elle admet des solutions multiples, la solution pour laquelle la distance est maximale doit vérifier la condition supplémentaire

$$\{\ddot{P}(u)\}^T \{d(u)\} < 0$$

Les relations précédemment établies permettent de calculer la valeur de l'erreur de corde pour une portion de courbe donnée, approximée par la corde qui la sous tend.

### 2.2.2.3 Relations entre erreur de corde et hauteur de crête

Il est à signaler qu'il existe une relation entre la hauteur de crête et l'erreur de corde. En effet, Dahmane [Dahmane '93] a mis en évidence une interaction non négligeable entre le pas longitudinal  $l$  et la hauteur de crête  $h$ .

Lors du calcul du pas longitudinal, il est supposé, à tort, que deux passages de l'outil se font toujours au niveau nominal de la surface théorique. Or, une erreur dont la valeur augmente avec l'augmentation de la tolérance de forme est générée (figure 2.13).

Des exemples donnés [Dahmane '93] montrent en effet que pour un usinage réalisé avec une fraise de diamètre 10mm et une rugosité totale  $R_t$  prévue de  $20 \mu m$ , la variation de la valeur de la tolérance d'erreur de corde allant de  $20$  à  $80 \mu m$ , donnent des hauteurs de crête correspondants à des rugosités mesurées allant de  $31$  à  $80 \mu m$ . Soit jusqu'à 4 fois  $R_t$  prévue.

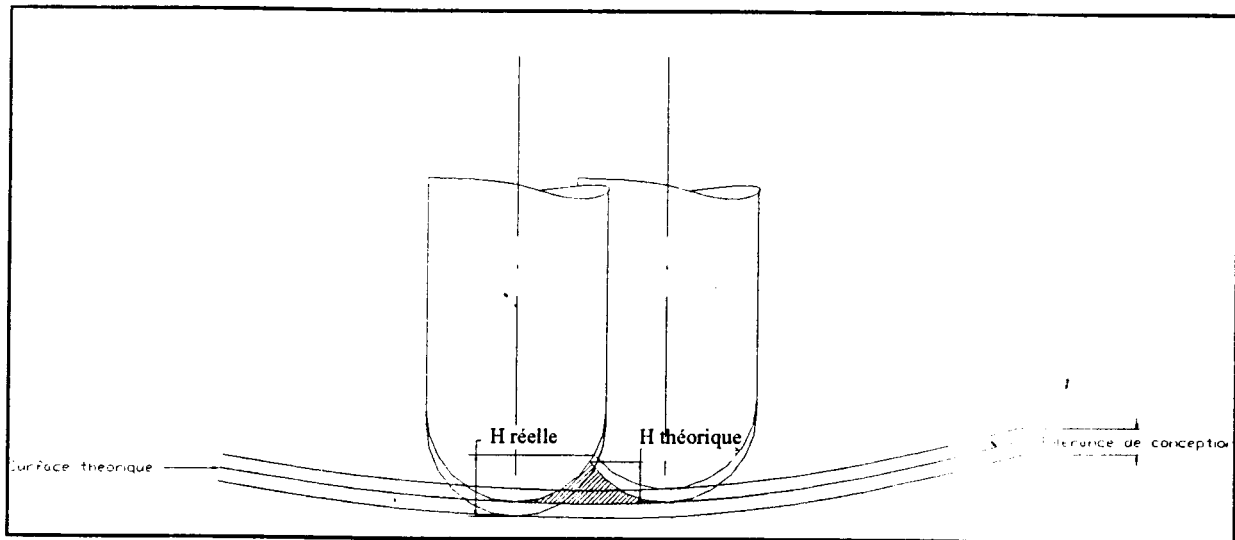


Figure 2.14 Relations entre erreur de corde et hauteur de crête

#### 2.2.2.4 Tolérances de CFAO

Les différentes phases du processus de traitement permettant le passage du modèle de l'objet à sa fabrication induisent des erreurs. Ces erreurs représentent les écarts entre le modèle CAO et l'objet réellement fabriqué. La maîtrise de ses différentes erreurs nous amène à introduire la notion de tolérances de CFAO, représentant les seuils admissibles pour ces différentes erreurs.

#### A) Tolérance imposée par la CAO

##### a) Définition

Par analogie à la définition de la tolérance  $TC$  imposée par la CAO, introduite dans le cas des courbes [Belaidi '98], nous la définirons dans le cas des surfaces comme étant l'écart maximal toléré entre la surface paramétrique  $S(u, v)$  théorique au sens mathématique (figure 2.15a) et sa représentation réelle au sens infographique, qui est une approximation de  $S(u, v)$  par un réseau de facettes polyédriques (figure 2.15b). Elle se calcule d'après les relations données dans la section §2.2.2.2.



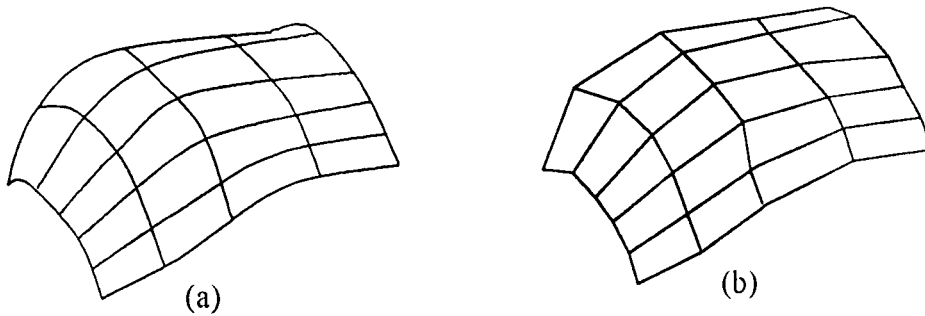


Figure 2.15 Modèle mathématique (a) et représentation infographique (b) d'une surface gauche.

### b) Tolérance de conception utile

La tolérance de conception utile  $TC_u$  sera définie comme étant un paramètre du constructeur représentant la valeur maximale de  $TC$  que l'on doit imposer pour construire une surface  $S(u, v)$  afin de garantir la tolérance de forme prescrite après usinage [Belaidi'98]. Le modèle de détermination de la valeur de la tolérance de conception utile est développé ci-dessous.

### B) Tolérance imposée par la FAO

#### Définition

La tolérance imposée par la fabrication est la tolérance prescrite par le constructeur pour le respect de l'erreur de crête  $h$  (Cf. Section 2.2.2.1) qui représente les défauts d'état de surface de 3<sup>ème</sup> et 4<sup>ème</sup> ordre, provoqués par les sillons imprimés par la forme de la partie active de l'outil sur la surface.

### C) Intervalle de tolérance de CFAO théorique

#### a) Définition

Par analogie à la tolérance de forme de courbe et de surface quelconques telles que définies par la norme *NF E 04-552* et en référence à la définition de la tolérance de conception et de fabrication introduites dans [Bensalah'90, '93], [Belaidi'98], nous définirons la tolérance théorique de CFAO ou  $IT(CF_{th})$  comme étant la distance de la zone délimitée par deux surfaces enveloppes distantes l'une de l'autre d'une distance normale représentant le diamètre  $\varnothing d$  d'une sphère dont le centre est le lieu géométrique qui décrit la surface nominale.

#### b) Algorithmes de calcul

La surface nominale est comprise entre deux surfaces parallèles séparées par la valeur de l'intervalle de tolérance  $IT(CF_{th})$  (Figure 2.16), qui dépend de  $TC$ , de  $TF$  du

sens des courbures (concave ou convexe) et du type de trajectoires (iso-paramétriques, en zigzag ou autres...). Ainsi la surface à usiner sera comprise entre  $S(u,v) + S_1(u,v)$  et  $S(u,v) - S_2(u,v)$  (Figure 2.16 et 2.17).

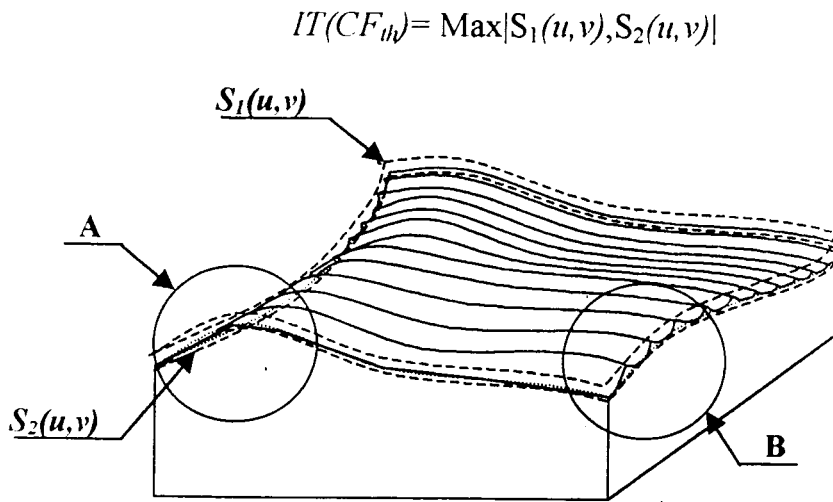


Figure 2.16 Définition de l'intervalle de tolérance de CFAO théorique.

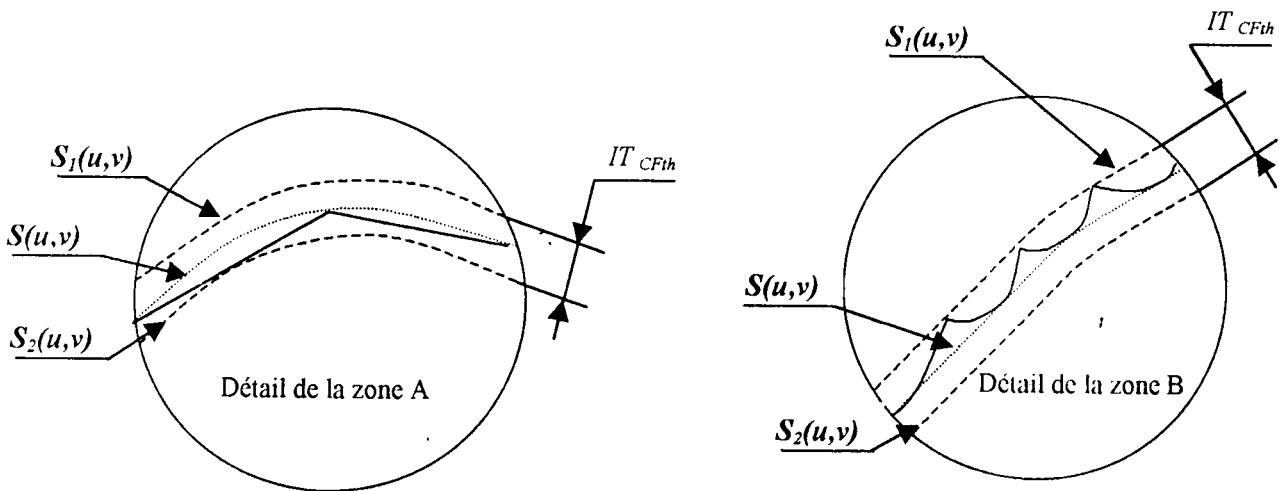


Figure 2.17 Influence de TC et de TF sur l'intervalle de tolérance de CFAO théorique.

A partir de l'algorithme introduit dans [Bensalah'90, '93] et généralisé dans [Belaidi'98], nous développerons un algorithme, appliqué au cas des surfaces usinées à l'outil hémisphérique ou torique en trajectoires iso-paramétriques ou par plans parallèles, à partir d'une discrétisation polyédrique de la surface. En contournage des surfaces de révolution, les valeurs de TC et de TF se cumulent alors que dans le cas de l'usinage de surfaces gauches à l'outil hémisphérique ou torique en trajectoires iso-paramétriques ou par plans parallèles, la composante relative à l'erreur de corde intervient principalement dans le sens longitudinal et celle relative à l'erreur de crête dans le sens transversal (Figure 2.17), les deux composantes étant tous de même

interdépendantes (Figure 2.18), tel qu'il a été mentionné précédemment (Cf. Section 2.2.2.3).

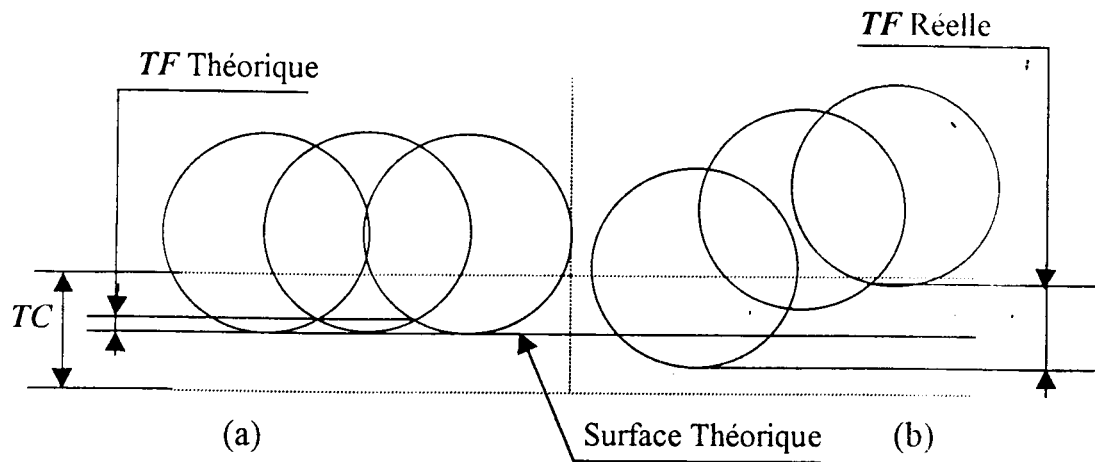


Figure 2.18 Relation entre TC et TF  
 (a) Configuration théorique optimiste,  
 (b) Configuration critique réelle.

La discrétisation polyédrique de la surface permet, de part la synchronisation implicite des trajectoires de balayages successifs de l'outil, une atténuation de l'influence de la tolérance de corde sur la valeur réelle de la hauteur de crête  $h$ .  $TC$  et  $TF$  étant évaluées (Cf. Sections 2.2.2.1 et 2.2.2.2), pour le cas convexe, la surface nominale enveloppe sera comprise entre deux surfaces parallèles

$$S_1(u, v) = S(u, v) + TC + TF \quad \text{et} \quad S_2(u, v) = S(u, v) + TC$$

alors que dans le cas concave, la surface nominale enveloppe sera comprise entre

$$S_1(u, v) = S(u, v) - TC \quad \text{et} \quad S_2(u, v) = S(u, v) - TC + TF.$$

Dans le cas mixte, du fait que  $TC > 0$  et  $TF > 0$ , nous aurons

$$S_1(u, v) = S(u, v) - TC \quad \text{et} \quad S_2(u, v) = S(u, v) + TC + TF$$

d'où l'expression de l'intervalle de tolérance de CFAO théorique  $IT(CF_{th})$

$$IT(CF_{th}) = \text{Max}[\text{Max}|S(u, v), S_1(u, v)|, \text{Max}|S(u, v), S_2(u, v)|] \quad [2.2]$$

$$IT(CF_{th}) = \text{Max}[\text{Max}(TC + TF, TC), \text{Max}(TC, TF - TC)]$$

$$TC + TF > TC \Rightarrow \text{Max}[TC + TF, TC] = TC + TF$$

$$\text{Max}[TC, TF - TC] = \begin{cases} TC & \text{si } TF < 2TC \\ TF - TC & \text{si non} \end{cases}$$

Comme  $TC + TF > TC$  et  $TC + TF > TF - TC$  donc

$$IT(CF_{th}) = TF + TC \quad [2.3]$$

## D) Tolérance de forme d'une surface gauche

### a) Erreur de profil lié au processus de CFAO

Les erreurs de profil d'une surface gauche usinée sur MOCN peuvent être décomposées en trois composantes essentielles [Belaidi '98] :

- Les erreurs de formes, principalement liées au comportement du système Machine-Outil-Pièce  $E_f(u, v)$ .
- Les erreurs paramétriques, liées aux propriétés mathématiques de la surface et caractérisant les écarts entre la surface moyenne et la surface réelle  $E_p(u, v)$ .
- Les erreurs de localisation de la pièce  $E_l(u, v)$ , composées des erreurs de position et d'orientation.

### b) Expression de l'intervalle de tolérance de forme

Une expression intuitive, permettant de palier à l'indisponibilité d'un modèle d'erreur analytique du processus d'usinage est proposée dans [Belaidi '98]

$$IT(F_p) < IT(P_{nr} \eta) + IT(E_f(u, v) + E_p(u, v) + E_l(u, v))$$

Où  $IT(P_{nr} \eta)$  représente la tolérance de forme admissible,  $P_m$  la tolérance nominale de la machine et  $\eta$  un coefficient de correction tenant compte de l'état de vétusté et d'usure de la machine.

Pour une machine donnée, la tolérance nominale machine est connue et son coefficient de vétusté  $\eta$  peut être déterminé expérimentalement. La tolérance sur l'erreur paramétrique  $E_p(u, v)$  étant exprimée par l'intervalle de tolérance de CFAO théorique il reste alors à déterminer les erreurs de forme  $E_f$  et de localisation  $E_l$  que nous désignerons par  $E_{fp}$ . La tolérance du processus de fabrication  $IT(E_{fp})$ , sera définie comme étant la tolérance sur les écarts probables du processus de fabrication et donnée par

$$IT(F_p) < IT(P_{nr} \eta) + IT(CF_{th}) + IT(E_{fp}).$$

soit en remplaçant  $IT(CF_{pv})$  par son expression

$$IT(F_p) < IT(P_m \eta) + (TC + TF) + IT(E_{fp}). \quad [2.4]$$

### 2.2.2.5 Modèle de prédiction de la Tolérance de CFAO

Le modèle de prédiction proposé par [Belaidi '98] nous permettra la prédétermination de la valeur de la tolérance de conception utile  $TC_u$  permettant de garantir une qualité géométrique conforme à une tolérance de forme prescrite et tenant compte des écarts du processus de CFAO, et cela dès la phase de construction.

#### A) Prédiction des écarts du processus de fabrication

Partant des travaux de Menq, Yai et Lai [Menq et al. '92] relatifs à la prédiction du nombre minimale  $N_c$  de points de contrôle requis sur une surface devant subir un contrôle suffisamment fiable d'une tolérance de profil spécifiée, basées sur les méthodes de probabilité-statistiques des tests d'hypothèse [Wonnacott '95] et sur la définition de l'indice de précision d'un processus [Juran '74], Belaidi [Belaidi '98] propose un modèle statistique de prédiction des écarts probables du processus de fabrication qui représentent en effet les écarts probables de forme et de localisation en considérant le raisonnement inverse à savoir, en partant d'un nombre de points de contrôles  $N_c$  fixes pour une tolérance de forme prescrite  $IT(F_p)$ . L'expression proposée est la suivante

$$E_{fp} = \sigma_m = \frac{IT(F_p) [(1-\gamma) + 2(2N_c)^{1/2}]}{6((2N_c)^{1/2} - \delta)} \quad [2.5]$$

Les coefficients  $\gamma$  et  $\delta$  étant spécifiés par l'utilisateur tel que mentionné dans [Belaidi '98].

#### B) Prédétermination de la tolérance de conception

Les écarts probables du processus de CFAO  $E_{fp}$  et les caractéristiques de la machine étant connues, l'expression [2.6], obtenue à partir de la relation [2.4] nous permet de calculer, pour un nombre de points de contrôle  $N_c$  donné, la valeur de la tolérance de conception utile  $TC_u$  à imposer afin de garantir la construction et l'usinage de la surface en conformité avec la tolérance de profil prescrite  $IT(F_p)$

$$TC_u = TC = IT(F_p) \left[ 1 - \frac{(1-\gamma) + 2(2N_c)^{1/2}}{6((2N_c)^{1/2} - \delta)} \right] - IT(P_m \eta) - IT \quad [2.6]$$

### 2.2.2.6 Les interférences et collisions

Afin de garantir le respect de la qualité imposée et la validité de la pièce, la prise en compte des erreurs d'usinage et la maîtrise de la tolérance s'impose lors du calcul des trajectoires d'outils, ainsi que la gestion des interférences éventuelles qui peuvent avoir lieu lors de la réalisation de la pièce sur MOCN, à cause d'une incompatibilité locale entre la géométrie de l'outil et celle de la pièce, et que nous désignerons par le terme « interférence d'usinage » (*figure 2.19*). Un autre type d'interférence peut aussi avoir lieu entre l'outil et une partie de la pièce autre que celle pour laquelle les trajectoires d'outils sont générées, comme dans le cas de l'usinage de surfaces composites carreau par carreau par exemple ou bien, entre l'outil et l'environnement de travail, telle que les brides ou tout autre partie de la machine ; ce type d'interférences sera désigné par le terme « collisions ».

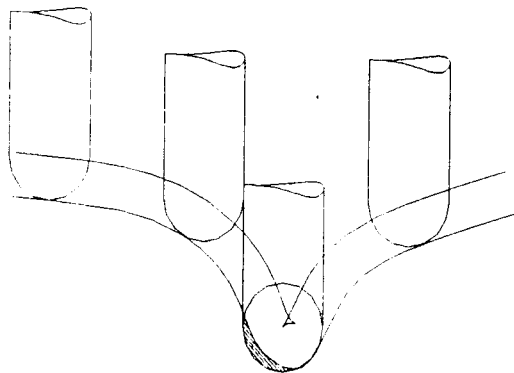


Figure 2.19  
Interférence outil/pièce  
locale ( $R > \rho$ ).

En pratique, si les collisions sont plus ou moins facile à éviter, grâce à une attention supplémentaire de l'opérateur lors de la préparation de l'usinage, les interférences ne peuvent être décelées et gérées que par le système de génération des trajectoires d'outils. Sur ce plan, les systèmes de génération de trajectoires d'outils peuvent être classés en cinq catégories. Les systèmes de première catégorie, sont les moins évolués et ne permettent aucune gestion des interférences (*figure 2.19*), le soin est alors laissé à l'utilisateur pour choisir une géométrie d'outil compatible avec la forme à usiner (*figure 2.20*). La seconde catégorie de systèmes, permet uniquement de déceler les interférences éventuelles lors de la phase de calculs et informe donc l'utilisateur de la nécessité de changer de géométrie d'outil sans plus d'informations ; les systèmes de troisième catégorie, permettent en plus de la détections des interférences, de fournir à l'utilisateur le rayon d'outils maximum pouvant être utilisé (uniquement dans le cas de l'usinage à l'outil à bout hémisphérique) ; les systèmes de quatrième catégorie laissent le libre choix à l'utilisateur de choisir une géométrie d'outils, puis génèrent les trajectoires d'outils avec éliminations des interférences éventuelles ; un surplus de matière est laissé dans ce cas et la pièce n'est pas conforme au modèle qui a servi pour la génération des trajectoires d'usinages (*figure 2.21*) ; les systèmes de cinquième catégorie, qui sont les plus efficaces en terme de gestion des

interférences, permettent en plus de l'élimination automatique des interférences, de prendre en considération les contraintes de productivité. En effet, de tels systèmes offrent un compromis entre la géométrie de l'outil et le débit de copeau, en se sens qu'ils permettent de choisir des dimensions d'outils permettant de réaliser un maximum de débit de copeau, quitte à laisser des surplus de matières au niveau des zones où les interférences ont été éliminés (figure 2.21), puis de générer les informations nécessaires pour une reprise locale d'usinage par la suite, avec des outils de géométries compatibles avec les régions en question (figure 2.22).

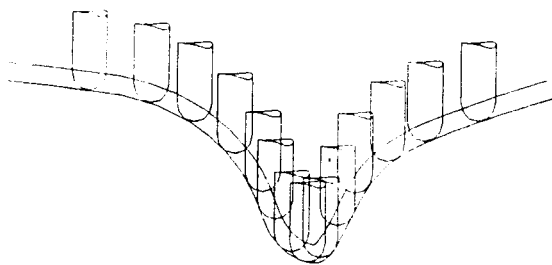


Figure 2.20  
Passe de finition ( $R \leq \min(\rho)$ )

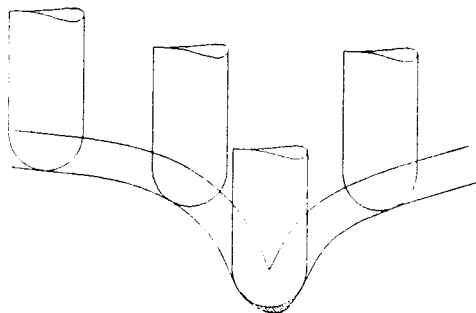


Figure 2.21  
Passe de finition ( $R > \min(\rho)$ )  
nécessitant une reprise locale  
d'usinage

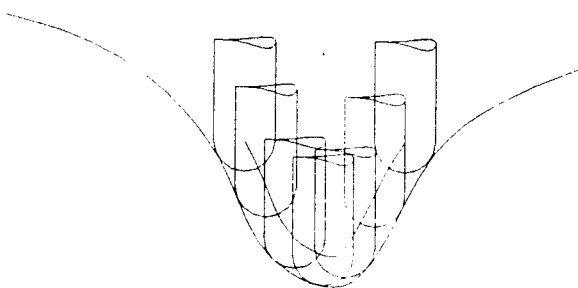


Figure 2.22  
Reprise locale d'usinage  
( $R < \min(\rho)$ )

### 2.2.3 Etude comparative et problématique

L'analyse précédente des différentes techniques de planification des trajectoires d'outil nous permet d'établir un tableau de synthèse donnant un comparatif des

différentes topologies. Pour cela nous avons proposé des critères d'appréciation qualitatifs, établis à partir des propriétés intrinsèques de chaque type de trajectoires. Une analyse quantitative nécessiterait des investigations plus poussées et des conditions expérimentales précises puisque les résultats sont conditionnés par les conditions particulières d'implémentation.



**Tableau de synthèse**

Critères	Complexité géométrique	Traitement informatique		Déplacements à vide	Volume fichier CN	Optimisation des parcours <sup>1</sup>	Observations
		Temps	Mémoire				
Topologie des trajectoires							
Spirales par décalage de contours	**	**	***	Augmente avec la complexité des zones à usiner	***	Non	
Spirales par réduction de contours	**	**	**	Augmente avec la complexité des zones à usiner	***	Non	
En zigzag parallèle	*	*	*	Très important en présence d'îlots	*	Oui	
En zigzag parallèle avec optimisation des déplacements à vide	*	**	***	Optimisés	*	Oui	
En zigzag iso-paramétriques	*	*	*	Très important en phase d'ébauche avec présence d'îlots	**	Non	Pose problème en usinage multi-carreaux
Unidirectionnelles iso-paramétriques	*	*	*	Très important en phase d'ébauche avec présence d'îlots	**	Non	Pose problème en usinage multi-carreaux
Unidirectionnelles parallèles	*	**	**	Très important en phase d'ébauche avec présence d'îlots	**	Oui	

1 : Pour les trajectoires parallèles, le choix de la direction de balayage influence la longueur totale des parcours d'outils ; d'où une possibilité d'optimisation.

## 2.3 Méthodes de calcul des trajectoires d'outils

Au niveau de la génération des trajectoires d'outil, il convient de faire la distinction entre l'aspect planification des trajectoires, qui porte sur la topologie et l'agencement des différents parcours d'outils [Marshall et al. '94b], de celui de calcul proprement dit et qui porte sur la détermination par calcul exact des positions de l'outil planifiées. La première se rapportant à la stratégie d'usinage à adopter, alors que la seconde correspond à la mise en œuvre de cette stratégie.

Les différentes méthodes de calcul des trajectoires d'outil, peuvent être classées selon le type de modélisation utilisé, à savoir : filaire, surfacique ou volumique [Tan et al. '90].

Concernant les méthodes traitant des modèles surfaciques, auxquelles nous nous intéressons dans le cadre de ce travail, elles peuvent être classées selon trois approches.

La première approche, implémentée sur les premiers systèmes de génération automatique de trajectoires d'outils [Bobrow '85], [Loney et Ozoy '87], consiste à calculer les points de contact outil/pièce à partir du modèle CAO, conformément à la planification adoptée, puis de calculer les points que doit décrire le point contrôle de l'outil qui constituent la trajectoire de l'outil (figure 2.23). Cette approche est basée sur la notion de distance normale à une surface. Sa mise en œuvre est généralement implémentée grâce à des algorithmes de calculs itératifs sujets à des situations singulières pour lesquelles la convergence n'est pas garantie [Faux et Pratt '87], [Choi et June '89] et donnant lieu à une instabilité numérique.

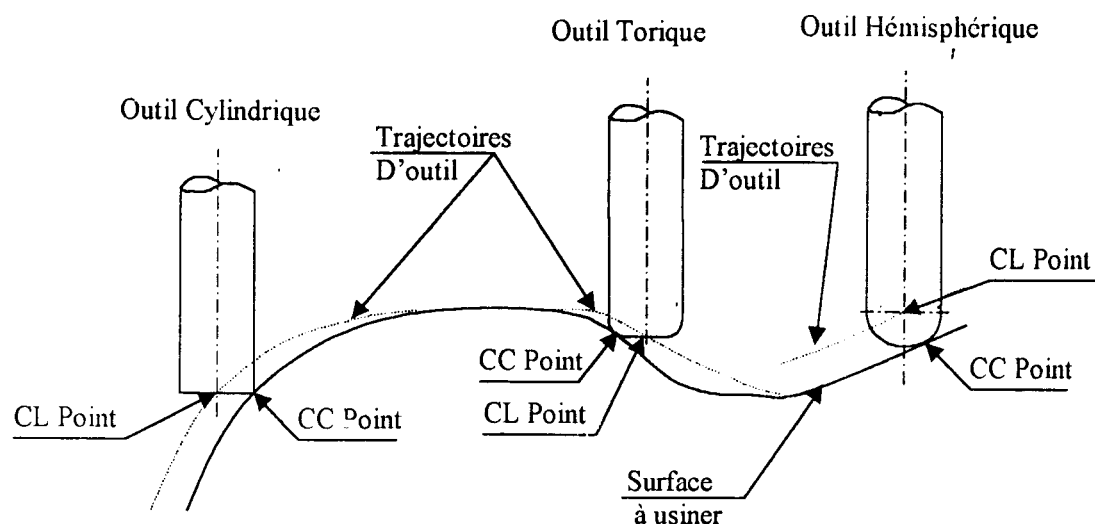


Figure 2.23 Point de contact outil/pièce et point de contrôle outil.

La seconde approche, permet d'effectuer les calculs à partir du modèle surfacique de l'objet, en effectuant une discrétisation polyédrique de ce dernier (figure 2.24). En plus de la très grande stabilité numérique de cette approche, elle permet aussi une gestion totale des interférences et collisions [Choi et June '89], [Hwang '92], [Kim et Ko '94], comme nous le verrons dans les chapitres qui suivent.

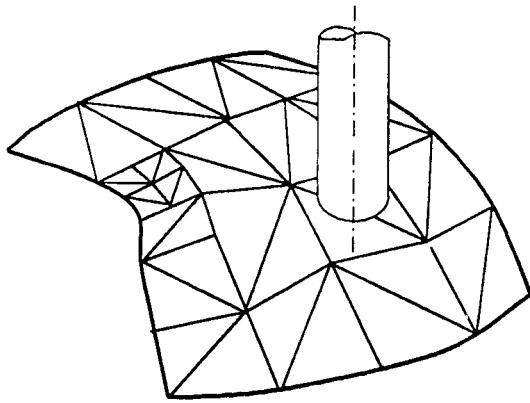


Figure 2.24  
Discretisation polyédrique  
de la surface.

La troisième approche est basée sur le concept de surfaces offset. En effet dans le cas de l'usinage à l'outil hémisphérique, les points centre d'outil appartiennent à la surface décalée vers le coté extérieur de la valeur du rayon  $R$  de l'outil (figure 2.25). De par son fondement, cette approche est limitée au cas des outils hémisphériques puisque la notion de surface offset n'a plus aucun sens dans le cas des outils cylindriques et toriques. De plus, les techniques de calculs et de traitement des surfaces parallèles sont assez délicates à mettre en œuvre et ne sont efficaces que sous certaines contraintes quant aux modèles mathématiques utilisés, aux degrés des surfaces ainsi qu'aux formes traitées [Choi et June '89], [Farouki '86], [Pham '92] ce qui ne répond pas aux critères de performances précédemment établis. En outre, le traitement des interférences pose des problèmes, tel que le calculs de courbes d'auto intersections de surface offset [Aomura '90],.

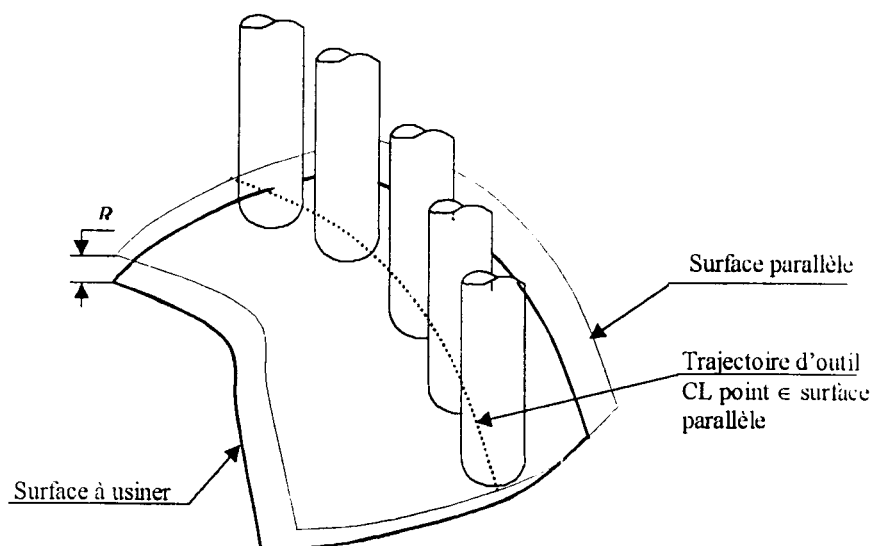


Figure 2.25 Calcul des CL points par construction  
de surface parallèle.

## 2.4 Synthèse bibliographique des techniques d'usinage d'ébauche:

La phase d'ébauche conditionne en grande partie la productivité de l'usinage des formes complexes, car en général, il se fait dans la masse ; d'où un volume de matière à enlever beaucoup plus important qu'en finition. De plus, les possibilités de choix des outils et des trajectoires sont plus grandes qu'en finition.

### 2.4.1 Usinage 2 axes ½:

Cette technique qui consiste en un usinage par couches parallèles généralement perpendiculaires à l'axe de l'outil, est très utilisée [Kuragano '92], [Herman '84], [Lee et Chang '91], l'usinage étant réalisé sur des MOCN 2 axes ½, dont la conception très rigide autorise des débits de copeaux importants.

Dans cette technique, la pièce à usiner est découpée en tranches limitées par des plans parallèles, généralement perpendiculaires à l'axe de l'outil, la distance entre deux plans consécutifs étant égale à la profondeur de passe. Les courbes d'intersection entre le plan de coupe et la surface à usiner délimitent les zones à usiner de celles à préserver. Le problème est donc ramené à un évidement de poches.

Dahmane [Dahmane '93] cite comme inconvénient de cette technique la génération de crêtes importantes en forme de marches d'escalier, l'outil utilisé étant dans la majorité des cas une fraise 2 tailles à bout cylindrique, imposant une passe de demi-finition à l'outil à bout hémisphérique [Lee et Chang '91], [Lee et al. '92], [Dahmane '93]. Néanmoins, beaucoup de travaux de recherches portant sur l'optimisation des techniques d'évidement de poches ont été réalisés ces dernières années [Suh et Lee '90], [Held et al. '94], [Lakaradju et al. '92], [Lee et al. '92], [Marshal et al. '94a, '94b], [Sun et Tsai '94], les résultats obtenus pourront être directement exploités dans ce type d'usinage.

Une autre approche, dans laquelle les plans de coupes sont parallèles à l'axe de l'outil, fut la technique la plus utilisée dans les premiers systèmes de FAO, notamment dans la méthode de Loney et Ozoy [Loney et Ozoy '87], elle est moins utilisée dans les systèmes récents du fait des déplacements à vides importants.

#### 2.4.1.1 Méthode d'Herman

##### *Planification des trajectoires*

La surépaisseur de matière à enlever est répartie [Herman '84] en couches parallèles, perpendiculaires à l'axe de l'outil (figure 2.26). Pour chaque couche, les trajectoires d'outil sont générées dans un plan perpendiculaire à l'axe de l'outil.

### Calcul des trajectoires:

Pour chaque plan de coupe, les points de la courbe d'intersection entre la surface à usiner et le plan de coupe sont calculés, puis un diamètre d'outil est choisi en tenant compte des risques de collision, une fois le pas transversal de balayage déterminé, la courbe d'intersection précédemment trouvée est approximée par une poly ligne respectant la tolérance dimensionnelle imposée par l'utilisateur. Les points de contact outil/pièce obtenus seront sauvegardés dans un fichier de points de contacts "CC-File", en chaque point de contact le vecteur normal à la surface est évalué puis les points centre d'outil seront calculés à partir des points de contacts et des normales à la surface en ces points ; le résultat est sauvegardé dans le fichier des points centres d'outil "CL-File".

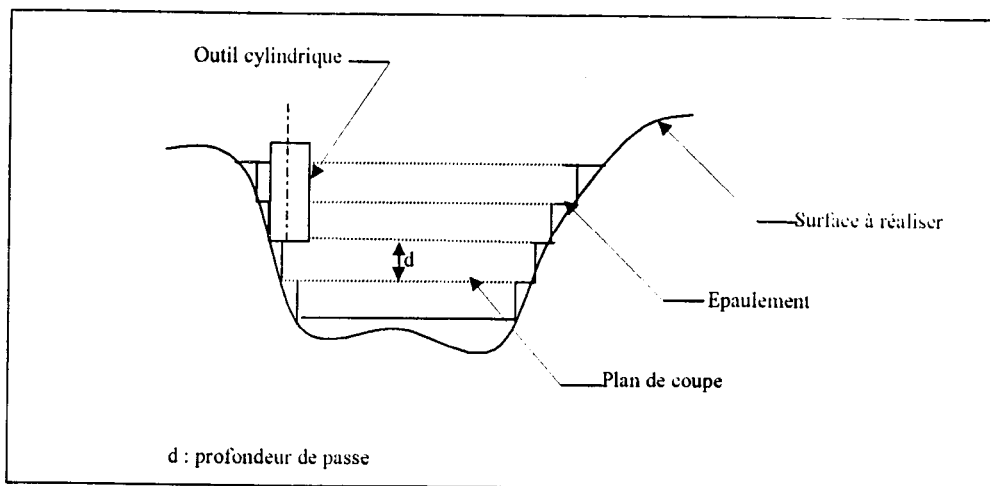


Figure 2.26 Planification des trajectoires d'Herman.

#### 2.4.1.2 Méthode de Loney et Ozoy

##### Planification des trajectoires

La surépaisseur de matière à enlever est répartie en couches parallèles entres elles et perpendiculaires à l'axe de l'outil, par contre les trajectoires d'outil sont générées dans des plans parallèles à l'axe de l'outil. Loney et Ozoy [Loney/Ozoy'87] adoptent un balayage en zigzag ou unidirectionnel.

##### Calcul des trajectoires

La surépaisseur de matière à enlever est répartie en couches parallèles entres elles et perpendiculaires à l'axe de l'outil ; pour une couche donnée (couche courante), ils définissent, comme indiqué sur la figure 2.27, les niveaux suivants (figure 2.28) :

- Niveau de référence,

- Niveau de dégagement,
- Niveau d'usinage courant.

Pour un diamètre d'outil donné, le pas transversal est calculé puis des trajectoires dites "type" et correspondant à une passe de finition sont générées ; ces dernières seront ensuite projetées sur les couches de matière au-dessus de la surface finale et n'ayant pas d'intersection avec cette dernière. Pour les couches ayant une intersection avec la surface finale, une correction des trajectoires est réalisée comme le montrent les figures 2.27 et 2.28. L'outil suit la trajectoire type jusqu'à atteindre la surface, puis remonte vers le niveau supérieur adjacent en restant tangent à la surface finale, ou décalé de la valeur de la surépaisseur de finition, si une passe de finition est prévue ; un dégagement vertical ramène ensuite, en vitesse rapide, l'outil jusqu'au niveau de dégagement, un autre déplacement en vitesse rapide, dans le niveau de dégagement permet ensuite de positionner l'outil à la verticale du point de plongée précédant et l'usinage est repris à vitesse de travail avec un suivi de la surface finale (ou décalé comme précédemment) jusqu'à atteindre le niveau de coupe courant, et ainsi de suite.

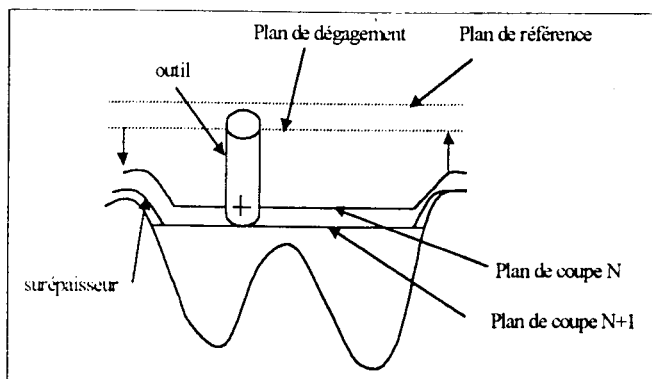


Figure 2.27  
Planification des trajectoires de  
Loney et Ozoy.

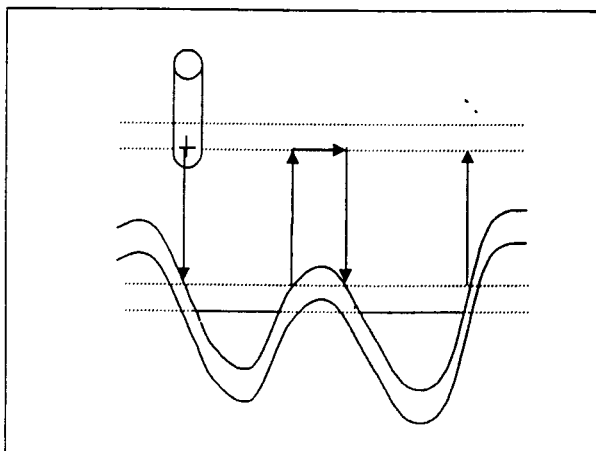


Figure 2.28  
Trajectoires d'outil de  
Loney et Ozoy.

### 2.4.1.3 Méthode de Lee et Chang

#### *Planification des trajectoires*

Lee et Chang [Lee et Chang'91] utilisent des trajectoires parallèles en zigzag. Lorsque, pour un plan de coupe donné, plusieurs poches sont obtenues, un algorithme d'optimisation permet de minimiser les passes à vides de transition.

#### *Calcul des trajectoires*

Cette méthode est en fait une combinaison des deux précédentes. Dans un premier temps une évaluation du volume de matière à enlever est réalisée, en calculant pour chaque niveau de coupe le volume de matière à usiner. Si le volume total est important, la technique d'Herman est d'abord mise en œuvre avec un outil cylindrique suivi d'une demi-finition à l'outil hémisphérique. Sinon, la méthode de Loney et Ozoy est utilisée avec un outil hémisphérique. Dans le premier cas, les courbes d'intersection obtenues pour chaque plan de coupe sont approximées par des polygones ; si les poches obtenues contiennent des îlots, elles sont subdivisées en sous-poches sans îlots. Dans le deuxième cas, la méthode de Loney et Ozoy est utilisée.

### 2.4.1.4 Techniques d'évidement de poches planes

#### 2.4.1.4.1 Méthode de Persson

##### *Planification des trajectoires*

La méthode de Person [Persson'78] utilise des trajectoires par décalages successifs des contours, permettant un balayage en spirale de la poche.

##### *Calcul des trajectoires*

Cette méthode traite des poches sans îlots, avec un contour composé uniquement de segments de droites et d'arcs de cercles. Elle repose sur l'idée de partager l'aire de la poche en plusieurs sous-poches indépendantes, ayant chacune une entité du contour de la poche comme côté ; cette subdivision de la poche est réalisée grâce à l'établissement des bissectrices à chaque jonctions de deux entités du contour, en utilisant le diagramme de Voronoi. [Persson'78], [Held et al. '94], [Akam'96].

Pour la construction des trajectoires d'outil, à partir du point le plus éloigné de la poche (Figure 2.29a), un décalage de la moitié du diamètre de coupe effectif est effectué vers l'un des cotés du contour de la poche (Figure 2.29b), ensuite l'outil est déplacé parallèlement au coté considéré, jusqu'à atteindre une bissectrice (Figure 2.29c), l'outil continue son parcours parallèlement au nouveau coté de la poche correspondant au deuxième élément de définition de la bissectrice atteinte (Figure 2.29d et 2.29e), les segments suivants de la trajectoire sont calculés de manière

similaire ; une fois la passe finie, un nouveau décalage est réalisé (Figure 2.29f). La construction des passes suivante est réalisée d'une manière similaire jusqu'à atteindre le contour de la poche, en laissant éventuellement une surépaisseur pour la passe de finition (Figure 2.29g).

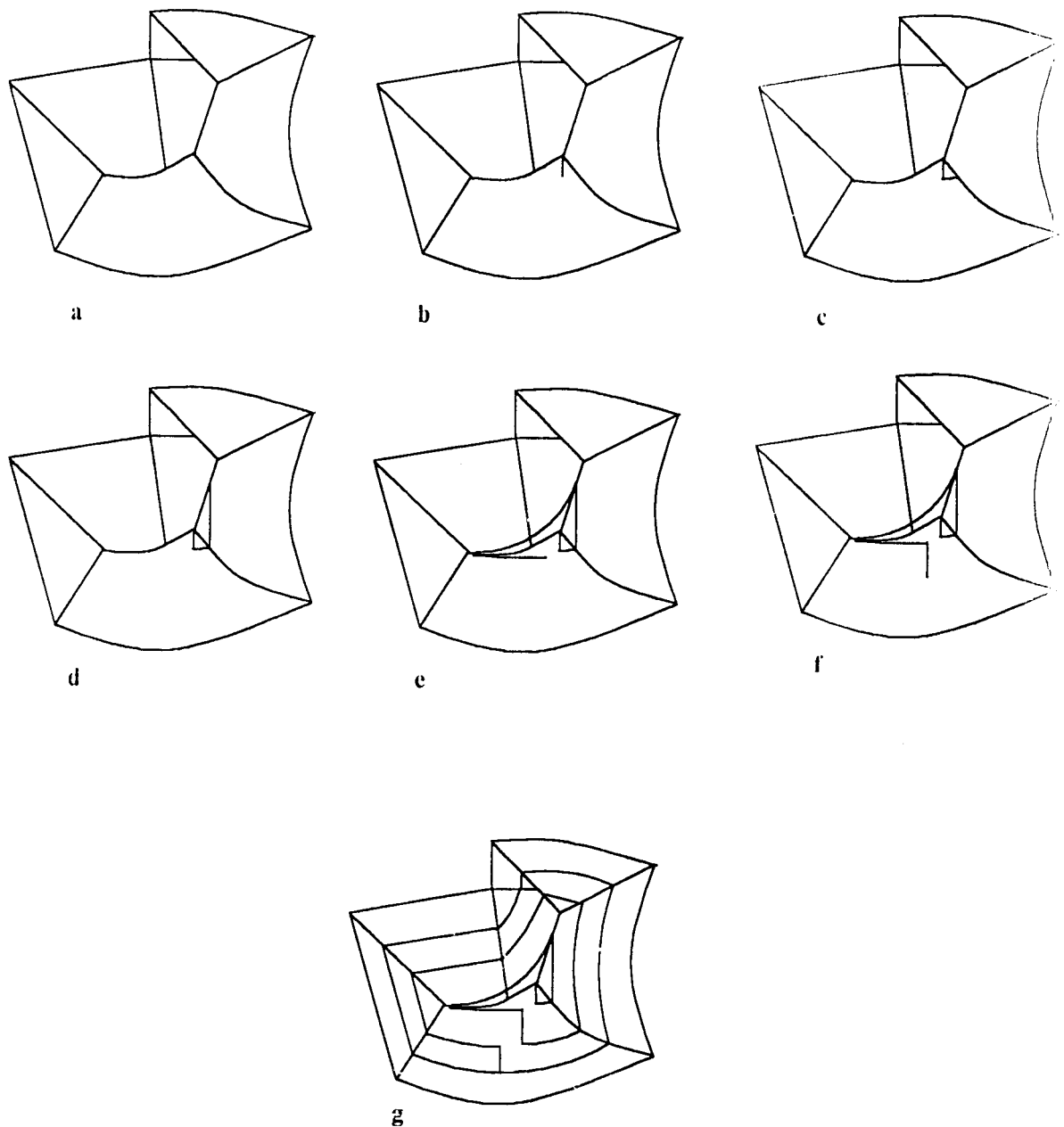


Figure 2.29 Trajectoires d'évidement de poche de Persson.



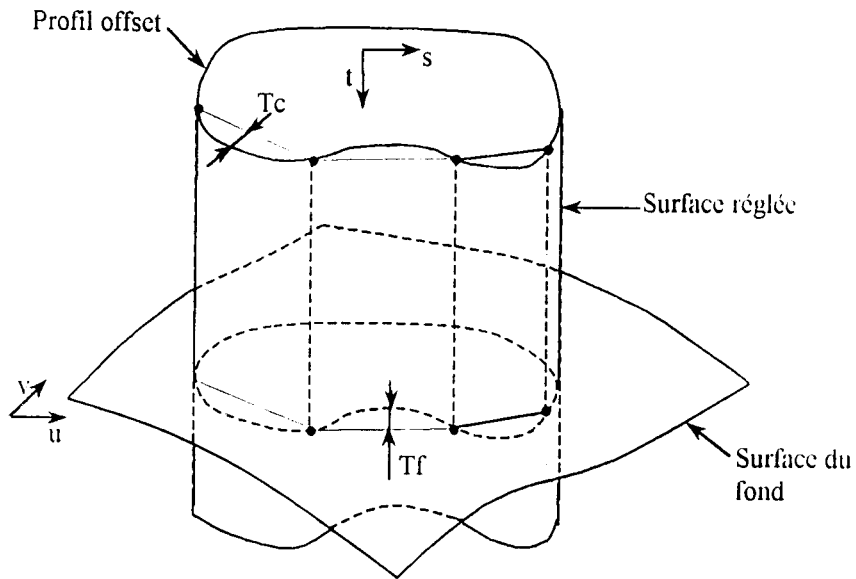


Figure 2.31 Projection du contour sur la surface du fond avec respect de la tolérance de corde.

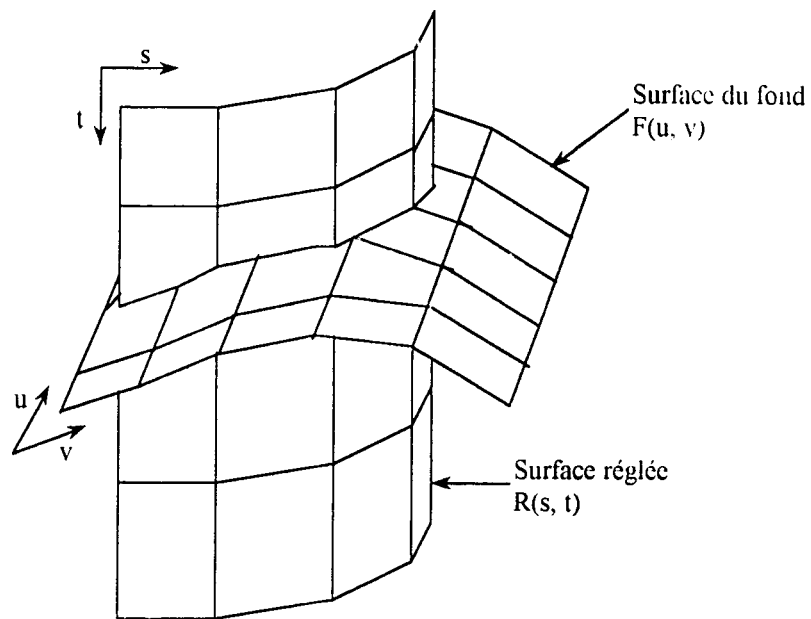


Figure 2.32 Discrétisation des surfaces pour les calculs d'intersections.

#### 2.4.1.4.3 Méthode de Held, Lukacs et Andor

##### *Planification des trajectoires*

La méthode proposée par Held, Lukacs et Andor [Held et al. '94] est en réalité une amélioration de la méthode de Persson, par son extension aux cas de l'usinage de poches à contour gauche avec îlots. Nous retrouvons donc des trajectoires en spirale.

##### *Calcul des trajectoires*

Nous retrouvons le diagramme de Voronoï au cœur de la méthode, et la prise en compte des îlots repose sur le principe de subdivision de la poche initiale en sous-poches monotones.

La présence d'îlots impose une manipulation géométrique supplémentaire qui consiste à relier ces derniers entre eux, puis au contour extérieur à l'aide de segments de droites appelés « ponts » (Figure 2.33). Nous aurons donc un nouveau contour constitué du contour extérieur de la poche, des ponts, et des éléments des contours des îlots. Le nouveau diagramme de Voronoï du contour de poche sans îlots ainsi obtenue devra être déterminé.

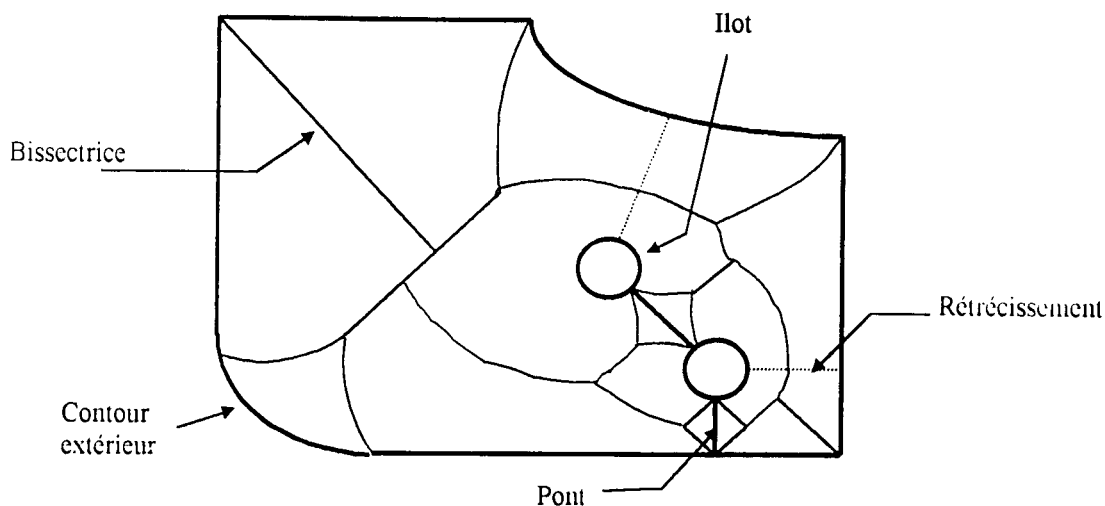


Figure 2.33 Utilisation des ponts dans le cas de présence d'îlots.

Pour la construction des trajectoires d'outils, une technique « graphe des sous-poches monotones » est proposée. Au niveau de chaque rétrécissement, un segment de droite qui divisera la poche en plusieurs sous-poches appelées « sous-poches monotones », où chacune peut contenir un seul ou un ensemble de points les plus éloignés des éléments qui la constituent (Figure 2.34).

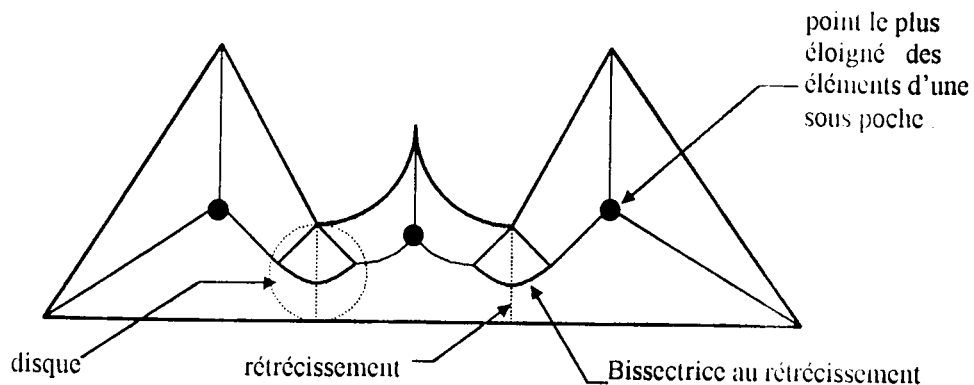


Figure 2.34 Construction des trajectoires d'outils  
et « graphe des sous poches monotones »

## 2.4.2 Problèmes posés et étude critique

Le critère le plus important pour la phase d'ébauche étant la productivité, il apparaît de l'analyse précédente que les techniques les plus appropriées sont celles d'un usinage en 2axes  $\frac{1}{2}$  par plans parallèles perpendiculaires à l'axe de l'outil, qui présentent les avantages suivants par rapport aux autres techniques :

- Utilisation de machines outils robustes permettant des débits de matière important,
- Coût horaire machine bas,
- Abondance de résultats de recherche (techniques d'optimisation, techniques de calcul d'intersection, etc.)
- Ne présente pas de restrictions quant au modèle, ni aux formes de pièces pouvant être traitées.

## 2.5 Synthèse bibliographique des techniques d'usinage de finition:

### 2.5.1 Classification

Contrairement à la phase d'ébauche pour laquelle le critère prépondérant était la productivité, la phase de finition est conditionnée en plus du critère de productivité par des contraintes de respect des tolérances dimensionnelles et géométriques ainsi que des états de surface de la pièce finie.

De même que pour la phase d'ébauche, au niveau de la génération des trajectoires de finition il convient de faire la distinction entre l'aspect planification des trajectoires, qui porte sur la topologie et l'agencement des différents parcours d'outils [Marshal et al. '94b], de celui de calcul proprement dit et qui porte sur la détermination exacte des positions planifiées de l'outil.

### 2.5.1.1 Méthode Automatic Programing Tools « APT »

#### Planification des trajectoires

La méthode A.P.T [Bobrow '85], [Faux/Pratt '87], [Marty et al. '92], développée par le 'Massachussets Institut of Technology' à la fin des années soixante, est l'une des toutes premières méthodes de génération automatique de trajectoires d'outil pour l'usinage de surfaces complexes. Cette méthode nécessite un degré élevé d'interactivité avec l'utilisateur et son efficacité est donc largement conditionnée par l'expérience et l'habileté de celui-ci ; la planification des trajectoires est confiée à ce dernier qui doit définir pour chaque parcours d'outil une surface guide que l'outil doit suivre au cours de chaque passe d'usinage le long du modèle.

#### Calcul des trajectoires

Les parcours d'outils sont définis par une série de surfaces guides puis d'un parcours final le long des courbes frontières du carreau à usiner. A partir d'un point de départ initial défini par l'utilisateur, l'outil est déplacé de façon à parcourir la courbe d'intersection entre la surface à usiner et la surface guide ; les points de passage de l'outil sont calculés à l'aide d'un algorithme itératif. Pour les surfaces à variation de courbure prononcée il arrive que l'algorithme diverge, tel qu'il est détaillé dans [Faux et Pratt '87].

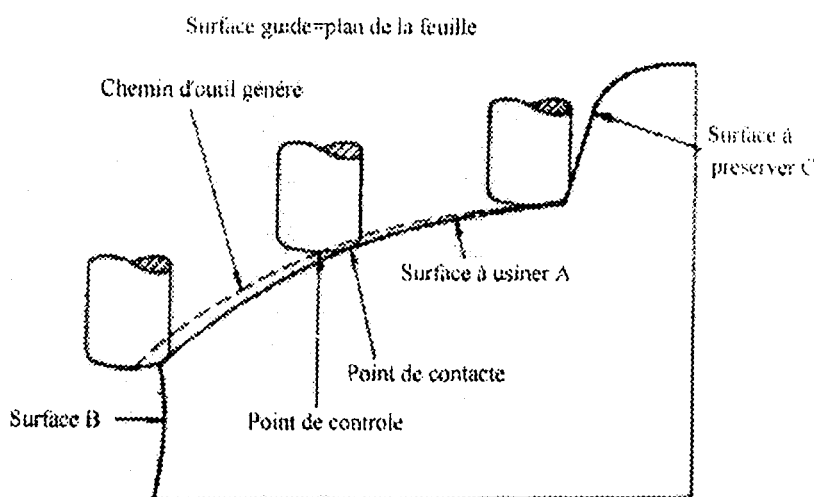


Figure 2.35 Méthode A.P.T.

Pour l'évaluation de l'erreur de corde, la courbe est localement approximée par son cercle osculateur, la valeur de l'écart maximum se situe donc au milieu du segment.

Pour la valeur de la hauteur de crête, aucune évaluation n'est prise en charge par le système et il appartient à l'utilisateur de définir les surfaces guide d'outils successifs, en fonction des variations de courbure locale de la surface à usiner.

La méthode A.P.T ne gère les interférences outil/pièce qu'au niveau des transitions d'un carreau au suivant, le long d'une même trajectoire.

### 2.5.1.2 Méthode de Bobrow

#### *Planification des trajectoires*

BOBROW [Bobrow '85] propose un algorithme de génération de trajectoire d'outil par plans parallèles, les chemins d'outils ainsi obtenus permettent soit un balayage unidirectionnel, soit un balayage en zigzag.

#### *Calcul des trajectoires*

L'approche adoptée pour le calcul des trajectoires consiste à construire des courbes de contact outil/pièce, à partir des courbes d'intersections entre les surfaces composant le modèle et un réseau de plans de coupe parallèles (figure 2.36).

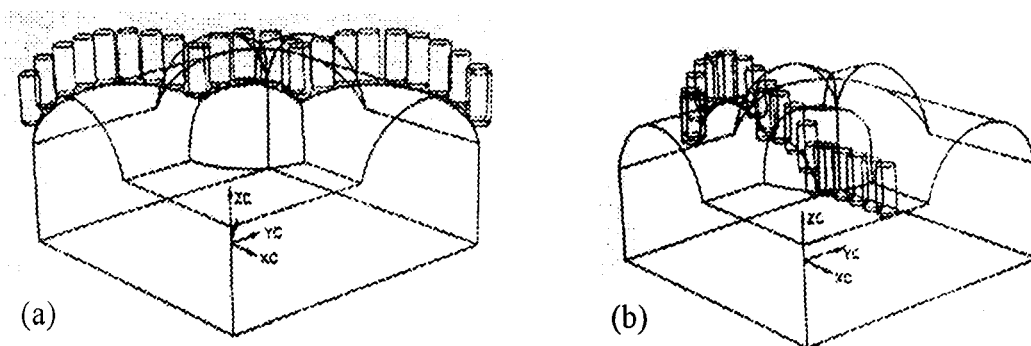


Figure 2.36 Passe d'usinage par plans de coupe quelconques (a), et parallèles à  $\{ox,oz\}$  (b).

La technique de Bobrow étant inspirée de la méthode A T P, l'utilisateur doit au préalable définir les surfaces à usiner et celles à préserver. L'algorithme génère dans un premier temps les courbes d'intersection entre le plan de coupe infini et les différents carreaux infinis composant l'objet (carreaux complets sans restriction), et cela grâce à une routine standard du système de CAO utilisé « P.A.D.L.2 », ensuite pour chaque courbe, les portions appartenant à l'objet sont identifiées ; puis mémorisées. Pour la génération du chemin d'outil l'utilisateur doit donner le point de départ de l'usinage, l'algorithme recherche ensuite dans la liste des portions de courbes mémorisées celles dont l'extrémité coïncide avec l'entrée de l'utilisateur, le point est ensuite translaté à l'autre extrémité de la courbe et l'algorithme de recherche est

réutilisé et ainsi de suite jusqu'à épuisement de toutes les portions de courbes constituant le parcours. Lors de la transition d'une surface à une autre un algorithme d'élimination des interférences similaire à celui de A.T.P est mis en œuvre.

Etant donné que les routines standard de P.A.D.L2 sont utilisées pour la détermination des courbes d'intersection, aucun détail concernant la prise en charge ou non des tolérances d'erreur de corde n'est donné.

Il en est de même pour le contrôle de l'erreur de crête.

Par contre les interférences outil/pièce sont gérées de la même façon que pour la méthode A.P.T, donc uniquement au niveau des transitions de l'outil d'un carreau à un autre, le long de la trajectoire.

### 2.5.1.3 Méthode de Choi, Lee, Hwang et June

#### *Planification des trajectoires*

Les travaux de Choi et al. [Choi et al. '88] représentent une référence dans le domaine de l'usinage des formes complexes. La méthode qu'ils proposent est basée sur la technique des points de contact cartésiens, elle permet l'utilisation des trajectoires unidirectionnelles ou en zigzag.

#### *Calcul des trajectoires*

A partir d'un quadrillage rectangulaire du domaine de la projection du modèle dans le plan xy (figure 2.38) des droites parallèles à l'axe z (figure 2.37 et 2.38) sont menées et les points d'intersection de ces droites avec les primitives du modèle sont déterminés afin de constituer les points de contact outil/pièce. Dans le cas où la droite coupe plusieurs primitives, l'arbre C.S.G est utilisé pour déterminer le point à retenir.

Les points centre d'outil sont ensuite calculés à partir des points de contact outil/pièce et des vecteurs normaux en ces points.

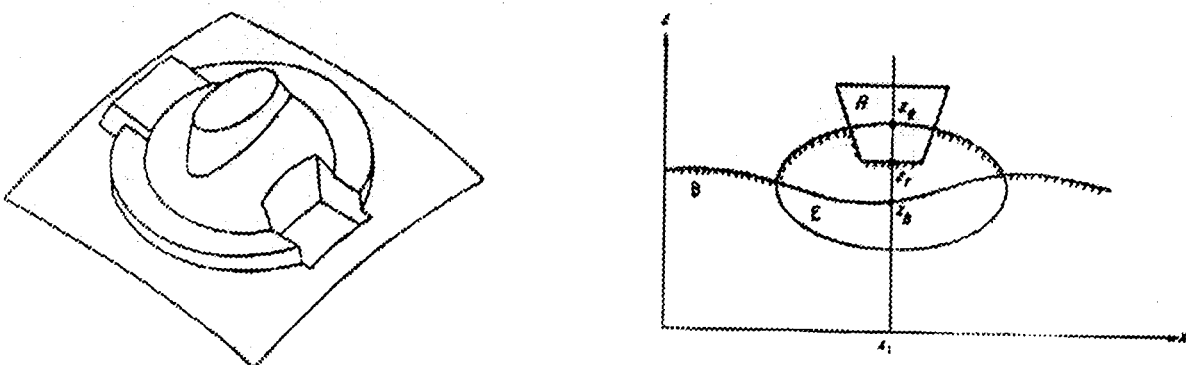


Figure 2.37 Utilisation du modèle C.S.G pour le calcul des trajectoires d'outil

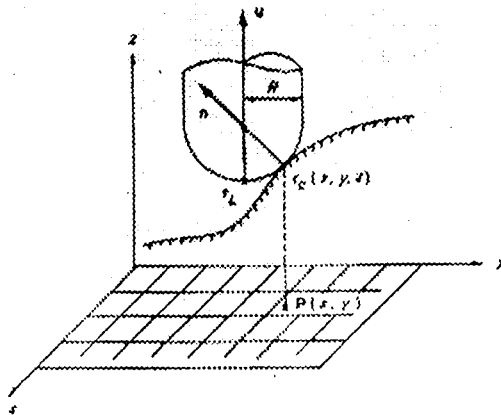


Figure 2.38 Quadrillage du domaine.

Pour l'évaluation de l'erreur de corde, une vérification de l'écart entre les chemins d'outils produits par les  $C L$  points initiaux et la surface réelle à usiner est effectuée. Si cette déviation est supérieure à la tolérance imposée par l'utilisateur un point de contact outil/pièce intermédiaire est inséré et le point centre d'outil correspondant est calculé, et ainsi de suite.

Quant à la tolérance de hauteur de crête, elle est prise en compte lors de la détermination du pas transversal  $l$ . Pour chaque point de contact outil/pièce le rayon de courbure de la surface est évalué puis le pas  $l$  est calculé par la relation suivante :

$$l = \frac{|\rho| \sqrt{4(R + \rho)^2 (h + \rho)^2 - (\rho^2 + 2R\rho + (h + \rho)^2)^2}}{(R + \rho)(h + \rho)}$$

A partir du pas curviligne  $l$  ainsi obtenu, le calcul du pas transversal cartésien  $l_p$  (Figure 2.39) sera donné par :

$$l_p = l |e - (e \cdot k)k| \quad \text{avec} \quad \begin{cases} k = (0, 0, 1) \\ e = (n \times t) / |(n \times t)| \\ t = n \times (tp \times k) / |n \times (tp \times k)| \\ n : \text{vecteur normale unitaire} \\ tp : \text{vecteur unitaire dans le plan } ox, oy \end{cases}$$

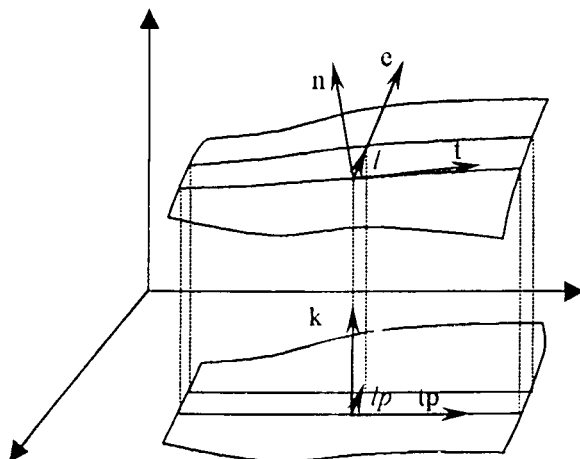


Figure 2.39 Passage du pas curviligne  $l$  au pas cartésien  $l_p$

La valeur du pas transversal retenu sera le minimum des valeurs évaluées de  $lp$ .

Cette méthode ne permet d'éliminer que les interférences longitudinales et seulement pour le cas de l'outil à bout hémisphérique.

#### 2.5.1.4 Méthode de Hwang

##### *Planification des trajectoires*

Les trajectoires d'outil obtenues par cette technique sont des trajectoires cartésiennes pures (C L data cartésiens) et permettent soit un balayage unidirectionnel soit un balayage en zigzag.

##### *Calcul des trajectoires*

La méthode de Hwang [Hwang'92] est basée sur la technique de discrétisation de la surface en éléments géométriques simples. La discrétisation est réalisée à l'aide de facettes triangulaires construites à partir d'un maillage de points de la surfaces obtenues à partir des points d'intersection entre la surface et un réseau de plans de coupes parallèles, sous contrainte des tolérances d'erreur de corde et de hauteur de crête.

Les points obtenus seront pris comme points de contact outil/pièce, puis pour chaque point de contact un point centre d'outil initial est calculé par la relation :

$$CLP_i = CCP_i + Rn_i$$

Pour chaque point centre d'outil initial ainsi obtenu, une vérification aux interférences outil/pièce est réalisée en calculant la distance de chaque facette de la zone de contact outil/pièce (figure 2.40), si cette distance est inférieure à la valeur du rayon de l'outil, la valeur de la cote  $z$  qui permet de positionner l'outil de sorte qu'il soit juste tangent à la facette en question est calculée, la valeur maximale de toutes les cotes  $z$  précédemment calculées (figure 2.41) sera affectée à la cote  $z$  de l'outil.

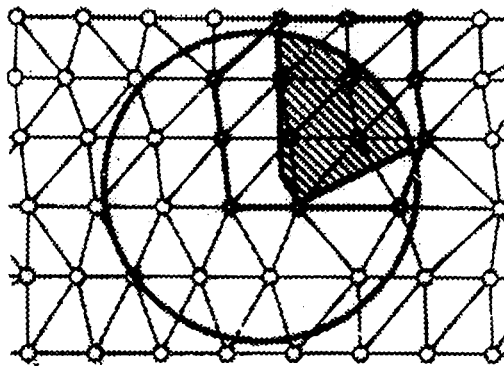


figure 2.40 Zone de contact outil/pièce.



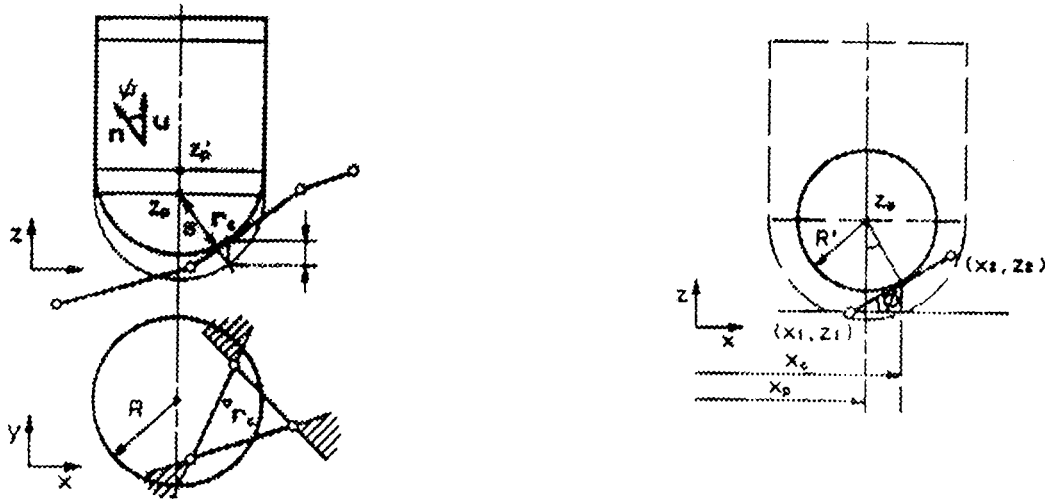


figure 2.41 Elimination des interférences outil/pièce.

La tolérance d'erreur de corde est prise en compte lors de la phase de discrétisation de la surface.

Lors du choix des plans de coupe, la tolérance de hauteur de crête est prise en compte

Cette méthode permet une gestion rigoureuse des interférences outils/pièce aussi bien longitudinales que transversales. La détection et l'élimination des collisions peuvent aussi être prises en compte. De plus une reprise d'usinage peut être implémentée.

### 2.5.1.5 Méthode de Choi et June

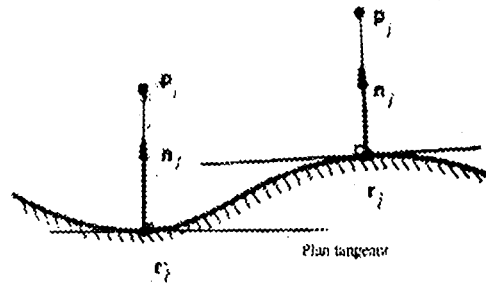
#### *Planification des trajectoires*

Les trajectoires obtenues par la méthode de Choi et June [Choi et June '89], sont similaires à celles de Hwang (Cf. Section 2.5.1.4), à savoir des trajectoires cartésiennes pures, car les points de contact outil/pièce sont calculés de la même façon ; elles permettent donc soit un balayage unidirectionnelle, soit en zigzag.

#### *Calcul des trajectoires*

La phase de calcul des points de contact outil/pièce est réalisée à partir de la génération d'un maillage de points issus de l'intersection de la surface avec des plans de coupe parallèles sous contrainte d'une erreur de crête. Le passage des points de contact aux points centre d'outil avec élimination des interférences est différent de celle de Hwang. Choi et June établissent des conditions d'interférences basées sur les

relations topologiques liants les points de contact voisins (figure 2.42). Les relations nécessaires et suffisantes d'interférence sont établies en imposant à la surfaces des restrictions de formes qui limitent le domaine d'utilisation de la méthode. Ils supposent en effet que la surface est complètement visible, vue dans la direction de l'axe de l'outil, ce qui interdit tout recouvrement dans cette direction, cette condition est en réalité une condition intrinsèque à l'usinage de finition en 3axes, par contre la condition la plus contraignante et celle qui impose au contour extérieur de pouvoir être usiné en contournage sans interférences outils/pièce (figure 2.43): ce qui élimine la possibilité de l'usinage de reprise pour ce type de surfaces.



- Si  $(r_j - r_i) \cdot n_j < 0$  Alors CC point  $r_i$  convexe/ CC point  $r_j$
- Si  $(r_j - r_i) \cdot n_j > 0$  Alors CC point  $r_i$  convexe/ CC point  $r_j$
- Si  $(r_j - r_i) \cdot n_j = 0$  Alors CC point  $r_i$  parallèle/ CC point  $r_j$

Figure 2.42 Relations topologiques entre CC points voisins.

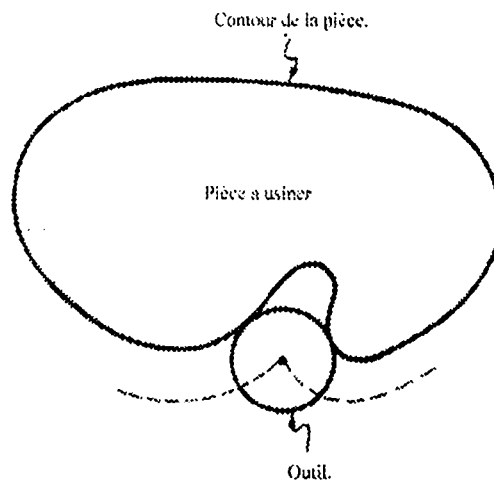


Figure 2.43 Restriction de la forme du contour extérieur de la pièce.

La phase de calcul des points de contact outil/pièce est réalisée en respectant une tolérance d'erreur de corde.

Bien que les auteurs n'aient pas implémenté une prise en compte de la tolérance de la hauteur de crête, une technique similaire à celle utilisée par Hwang peut être implémentée sans difficultés.

L'algorithme utilisé dans cette méthode permet l'élimination des interférences outils/pièce longitudinales et transversales. De plus, l'implémentation d'une reprise d'usinage est envisageable.

### 2.5.1.6 Méthode de Lee Choi et Chang

#### *Planification des trajectoires*

Lee et Chang [Lee et Chang'91], [Lee et al.'92] utilisent des trajectoires iso-paramétriques, ce qui permet soit un balayage unidirectionnel, soit en zigzag.

#### *Calcul des trajectoires*

Dans un premier temps un parcours d'outil iso-paramétrique initial est calculé avec respect d'une tolérance d'erreur de corde ; ensuite pour chaque Point de contact outil/pièce  $p_i$  sur le parcours initial sont identifiés tous les points  $p_j$  situés sur les parcours voisins et appartenant à la zone de contact outil/pièce, correspondant à la projection de l'outil sur la surface; la valeur de la hauteur de crête entre  $p_i$  et  $p_j$  est ensuite évaluée, si elle est supérieure à la hauteur de crête maximum tolérée, alors le pas transversal est diminué. Si la valeur de l'erreur de corde est supérieure à la valeur tolérée, le pas longitudinal est réduit et un nouveau point est calculé. Dans le cas où une interférence outil/pièce est détectée, les points d'interférences sont éliminés ; de nouveaux points de contact outil/pièces sont rajoutés, si nécessaire, pour éviter une interférence de convexité. Les calculs sont répétés pour un nouveau parcours d'outil jusqu'à générer toutes les trajectoires d'outil.

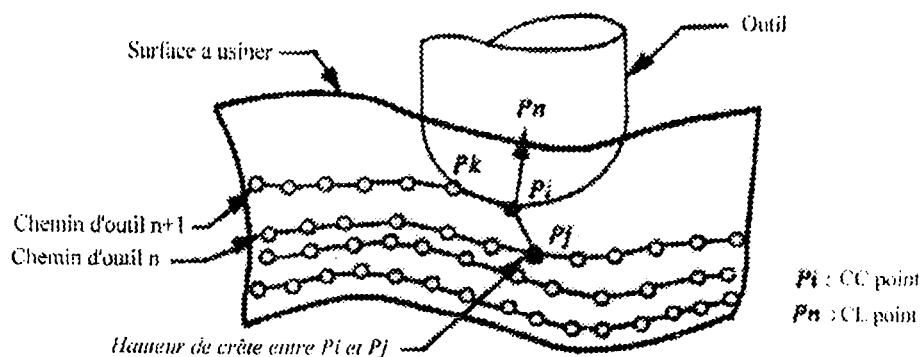


figure 2.44 Elimination des interférences outil/pièce.

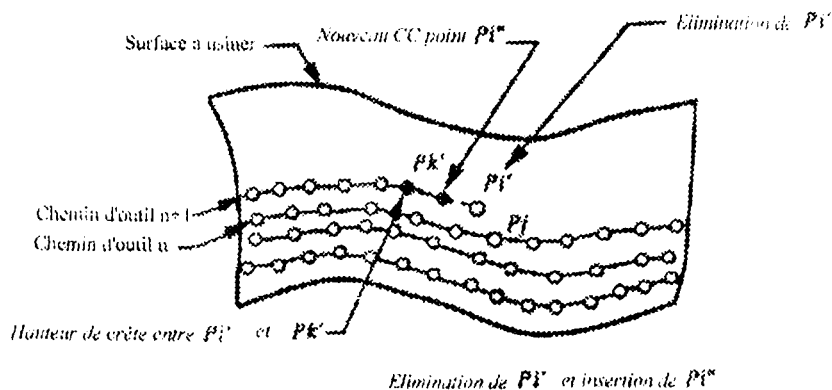


Figure 2. 45 Élimination des interférences outil/pièce.

Le calcul des Points de contact outil/pièces est réalisé en respectant une tolérance d'erreur de corde.

Le calcul des Points centre d'outil est réalisé avec respect de la tolérance d'erreur de crête.

L'algorithme présenté permet l'élimination des interférences outils/pièce longitudinales et transversales.

### 2.5.1.7 Méthode de Oliver, Wysocki et Goodman

#### *Planification des trajectoires*

Les trajectoires utilisées sont de type iso-paramétriques, ce qui permet un balayage soit unidirectionnel, soit en zigzag.

#### *Calcul des trajectoires*

Oliver et al. [Oliver et al. '93] proposent une méthode de calcul de trajectoires de type iso-paramétrique. La détermination des points de contact outil/pièces le long d'une courbe iso-paramétrique est réalisée sur la base d'un algorithme d'évaluation précise de la valeur de l'erreur de corde. Les Points centre d'outil sont calculés à partir des points de contact outil/pièces grâce à la relation :

$$CLP_i = CCP_i + Rn_i$$

La détection et l'élimination des interférences outil/pièce longitudinales sont réalisées à partir de l'analyse des chemins liant les Points centre d'outils d'un même parcours d'outils après détection des intersections segments/segments. La

détermination du pas transversal est réalisée sur la base de l'évaluation de la valeur de la hauteur de crête, le pas minimum est alors retenu.

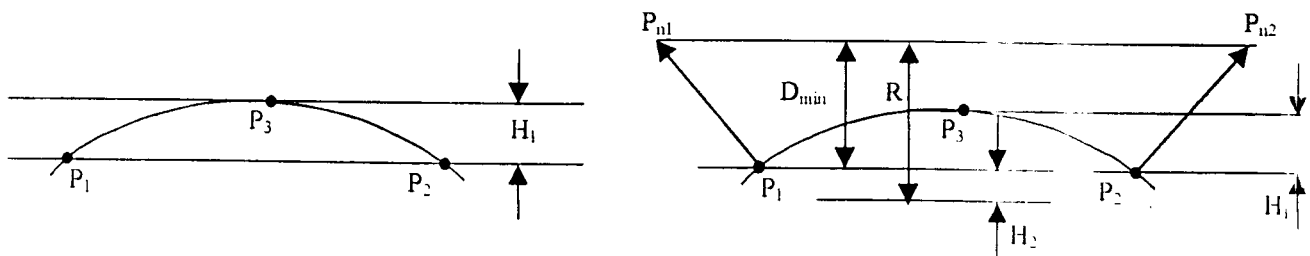


figure 2.46 Détails de calcul de l'erreur de corde

Oliver et al. proposent une méthode de calcul de la valeur de l'erreur de corde plus rigoureuse et plus précise que celles implémentées dans les autres méthodes étudiées. Ils attirent en effet l'attention sur le fait que la méthode communément utilisée et qui évalue la valeur de l'erreur de corde à la valeur moyenne des paramètres des deux points successifs du tronçon de courbe approximée localement par le segment de droite, ne donne un résultat acceptable que dans le cas d'un paramétrage uniforme et des valeurs de tolérance d'erreur de corde assez faibles (l'figure 2.46) ; dans le cas contraire l'erreur peut être importante ; Loney et al. proposent une méthode pour ces cas de figure [Loney et al. '85]. Un autre point très important, est mis en évidence ; l'erreur de corde ne représente pas réellement une tolérance de fabrication puisque l'outil n'usine pas exactement le contour discretisé tel qu'il est supposé dans la majorité des méthodes étudiées, comme illustré par la figure 2.46.

La détection et l'élimination des interférences outil/pièce ne sont assurées que dans le sens longitudinal, et cela sur la base de l'analyse de la courbe des points centre d'outil; un algorithme itératif permet de déterminer les valeurs des paramètres des points de la courbe iso-paramétrique de début et de fin de la zone d'interférences.

### 2.5.1.8 Méthode de Suresch et Yang

#### *Planification des trajectoires*

Suresch et Yang [Suresh/Yang'94] proposent une méthode d'usinage de surfaces gauche en finition utilisant un balayage en zigzag.

#### *Calcul des trajectoires*

La méthode de Suresch et Yang est originale puisqu'elle introduit une nouvelle approche pour le calcul des chemins d'outil. Les méthodes traditionnelles reposent toutes sur le concept des trajectoires parallèles soit dans l'espace euclidien (plans parallèles), ou bien dans l'espace paramétrique (courbes iso-paramétriques), alors que celle de Suresch et Yang repose sur le critère de hauteur de crête constante, ce qui

implique une réduction des parcours d'outils d'où une réduction des temps d'usinage ainsi que des volumes des fichiers de commande numérique (figure 2.47). De plus l'élimination du problème de chevauchement des trajectoires permet d'avoir une meilleure qualité d'état de surface.

Le calcul des Points de contact *outil/pièces* se fait dans le sens longitudinal en fonction de la tolérance d'erreur de corde. Par contre dans le sens transversal, c'est le critère de hauteur de crête constante qui est utilisé.

Partant d'un chemin d'outil initial, construit généralement à partir d'une courbe frontière sous contrainte d'une tolérance de corde, les points du parcours d'outil adjacent seront déterminés en calculant, pour chaque point de contact *outil pièce*, les valeurs des paramètres  $u$  et  $v$  qui garantissent la tolérance de corde et respectivement une hauteur de crête prescrites.

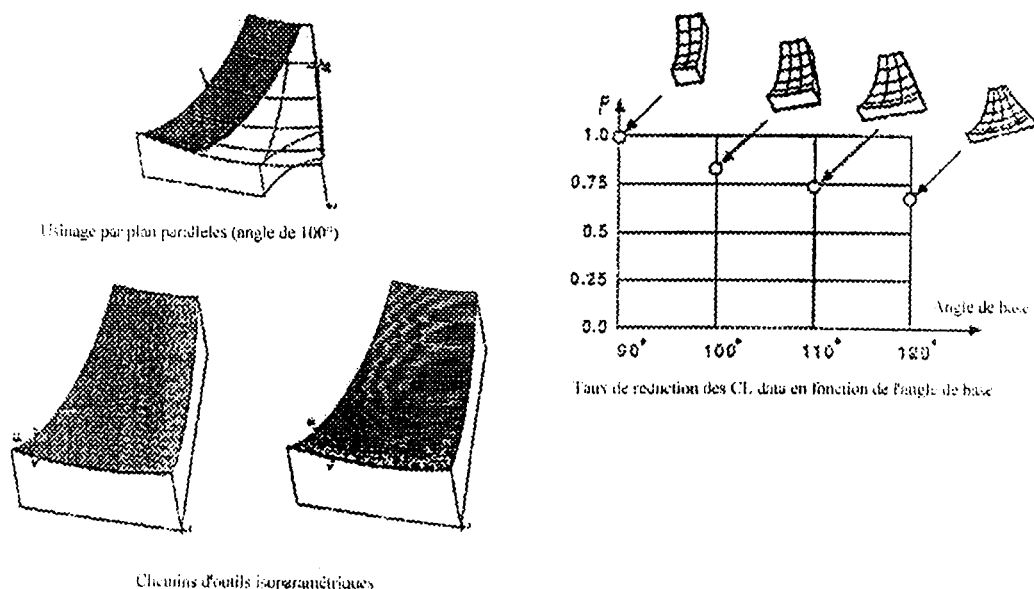


figure 2.47 Taux de réduction de la taille du CL  
File.

La méthode utilisée pour l'évaluation de l'erreur de corde est celle de l'approximation locale d'une courbe par son cercle osculateur (Cf. Section 1.2.2.2.2).

La méthode proposée permet un usinage à hauteur de crête constante, égal à la valeur prescrite par l'utilisateur mais ne gère pas les interférences.

### 2.5.1.9 Méthode de Marschal et Griffiths

#### *Planification des trajectoires*

Marshall et Griffiths [Marshall et Griffiths'94b] proposent une méthode originale de planification des trajectoires d'usinage qui combine deux types de

trajectoires. Les trajectoires en zigzag par plans parallèles et les trajectoires par lignes de niveau.

### *Calcul des trajectoires*

Cette méthode ne traite que de l'aspect planification des trajectoires d'outil. L'implémentation des calculs des points centre d'outil n'est pas abordée dans l'article

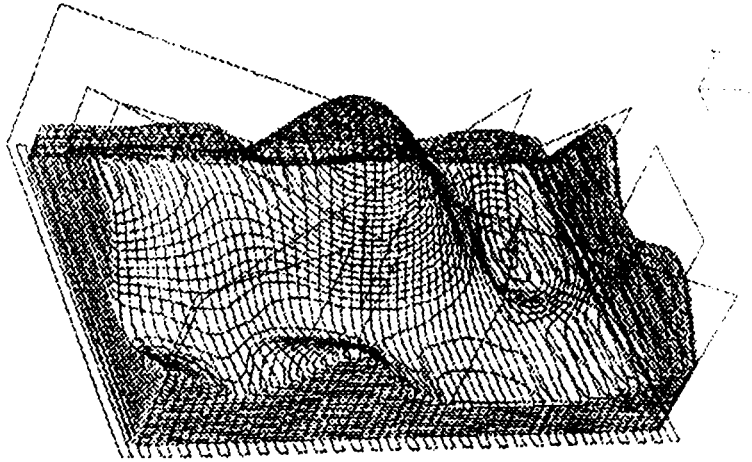


Figure 2.48 Trajectoires d'usinage de finition.

#### **2.5.1.10 Méthode de Kim et Ko**

##### *Planification des trajectoires*

Kim et Ko [Kim et Ko'94] proposent un algorithme de génération de trajectoires d'outils par plans parallèles ; ce type de trajectoires permet un balayage soit unidirectionnel soit en zigzag.

##### *Calcul des trajectoires*

Le calcul des trajectoires se fait en deux étapes. Dans un premier temps ils appliquent à la surface à usiner un algorithme de subdivision itérative auto-adaptative permettant de réaliser une approximation par des polyèdres de la surface vérifiant une tolérance imposée par l'utilisateur (figure 2.49). Dans le cas d'une surface de Bézier l'algorithme de redéfinition de l'intervalle utile d'une surface est utilisé, pour une surface B-spline Kim et Ko proposent de réaliser une conversion B-spline → BEZIER, ensuite d'appliquer l'algorithme de subdivision précédemment cité. La seconde étape consiste à calculer les points de contact outil/pièce. Pour cela, les polyèdres issus de la subdivision de la surface à usiner seront coupés par une série de plans de coupe parallèles à l'axe de l'outil, les segments d'intersection plan/facette représentent les trajectoires de balayage. L'auto-adaptabilité de l'algorithme de subdivision permet une optimisation du nombre de points interpolés.

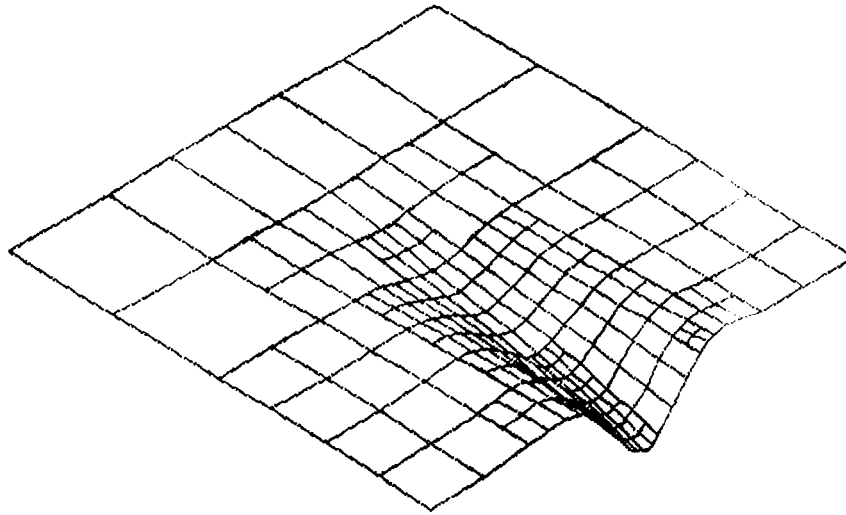


Figure 2.49 Discretisation de la surface.

En plus de la tolérance d'approximation fixée par l'utilisateur qui est prise en compte lors de la discrétisation de la surface, les auteurs incorporent dans l'algorithme une procédure de vérification de la déviation entre la courbe réelle d'intersection plan/surface et celle entre les facettes et le plan. Dans le cas où cette déviation serait supérieure à la tolérance imposée, le carreau en question est encore subdivisé.

Les auteurs ne préconisent pas dans leur article une prise en compte de l'erreur de crête.

Puisque l'algorithme n'effectue aucun contrôle d'interférence outil/pièce, il appartient donc à l'utilisateur de choisir un outil compatible avec la forme de la surface à usiner.

#### 2.5.1.11 Méthode de Kim et Kim

##### *Planification des trajectoires*

Kim et Kim [Kim et Kim '94] proposent une technique originale de planification des trajectoires d'outils. En effet, ils utilisent des trajectoires iso-paramétrique, mais à la différence des techniques de planification des trajectoires d'outils iso-paramétriques classiques, qui sont générées directement à partir de la définition paramétrique de la surface à usiner, Kim et Kim construisent d'abord une surface offset de la surface à usiner, par un algorithme basé sur les techniques de conversions de surface permettant de convertir la surface initiale en un réseau de carreaux de Coons, puis génèrent les trajectoires d'outils à partir de la définition paramétrique des surfaces offset ainsi obtenues.



### **Calcul des trajectoires**

La détermination des chemins d'outils est réalisée grâce à un algorithme de calcul des pas paramétriques en u et en v, sous contrainte du respect de la tolérance d'erreur de corde et de hauteur de crête, de façon concise. En effet, la valeur de l'erreur de corde est calculée sur la surface offset et non pas sur la surface à usiner, ce qui permet un contrôle plus rigoureux de l'erreur, cette technique est similaire à celle de Oliver et Al. [Oliver et Al. '93] ; le pas transversal est calculé sous contrainte du respect de la tolérance de hauteur de crête maximale.

L'algorithme de calculs des trajectoires d'outils proposé, n'effectue aucun contrôle d'interférence outil/pièce. Par contre il permet d'informer l'utilisateur si le rayon d'outil choisi est compatible ou non avec la géométrie de la pièce à usiner.

### **2.5.2 Etude comparative et problématique**

L'analyse précédente nous permet de mettre en relief trois niveaux de problèmes lors de la phase de génération des trajectoires d'outil.

Le premier cercle de problème concerne le niveau d'efficacité et de validité des relations de calculs mise en œuvre pour la prise en charge des erreurs et la maîtrise des tolérances.

Le second niveau porte sur la planification des trajectoires et ses répercussions sur l'efficacité des chemins d'outils correspondants ainsi que la qualité finale de la surface usinée qui en résulte.

Le dernier cercle de problèmes est relatif à la gestion des interférences et collisions.

Le tableau qui suit reprend une synthèse des méthodes étudiées par rapport à ces trois niveaux d'évaluation.

**Tableau de synthèse**

Méthode	Modèles supportés	Type de trajectoires		Technique de calcul des trajectoires	Err. Corde	Err. crête	Interférences		Contrôle Collisions.
		Trajectoires	Balayage				Long.	Trans.	
A.P.T (1967)	Surfaces paramétriques.	Parallèles à une surface données.	Unidirectionnel ou en zigzag	CC points par courbes d'intersection, CL points.	oui	non	Oui <sup>1</sup>	Non <sup>2</sup>	Oui <sup>2</sup>
Bobrow (1985)	Surfaces paramétriques.	Plans parallèles à l'axe de l'outil	Zigzag ou unidirectionnel	CC points par courbes d'intersection, CL points.	oui	oui	Oui <sup>1</sup>	Non <sup>2</sup>	Oui <sup>2</sup>
Choi Lee Hwang et June (1988)	Primitives CSG et surfaces paramétriques	Plans parallèles à l'axe de l'outil	Zigzag ou unidirectionnel	CC points cartésiens par quadrillage, CL points sous contrainte du respect de TC et h.	Oui	oui	oui	non	non
Choi et June (1989)	Surfaces paramétriques.	Plans parallèles à l'axe de l'outil	Zigzag ou unidirectionnel	CC points cartésiens par quadrillage, CL points avec élimination des interférences.	Oui	Oui	Oui	Oui	Oui <sup>3</sup>
Hwang (1992)	Surfaces paramétriques.	Plans parallèles à l'axe de l'outil	Zigzag ou unidirectionnel	CC points cartésiens par discrétisation, CL points avec élimination des interférences.	Oui	Oui	Oui	Oui	Oui <sup>3</sup>
Lec. Choi et Chang (1991)	Surfaces paramétriques.	Iso-paramétriques.	Zigzag ou unidirectionnel	CC points iso-paramétriques, CL points avec élimination des interférences.	Oui	Oui	Oui	Oui	Oui <sup>3</sup>
Oliver, Wysocki et Goodman (1993)	Surfaces paramétriques.	Iso-paramétriques	Zigzag ou unidirectionnel	CC points iso-paramétriques, CL points avec élimination des interférences.	Oui	Oui	Oui	Non	Non
Suresch et Yang (1994)	Surfaces paramétriques.	Hauteur de crête constante.	Zigzag	CC points sous contrainte de h constant, CL points.	Oui	Oui	Non	Non	non
Marshal et grifiths (1994)	Surfaces paramétriques.	Plans parallèles et lignes de niveau	Zigzag et contournage	néant					
Kim et KO (1994)	Surfaces paramétriques.	Plans parallèles à l'axe de l'outil	Zigzag ou unidirectionnel	CC points cartésiens par discrétisation, CL points.	Oui	Non	Non	Non	non
Kim et Kim (1995)	Surfaces paramétriques.	Iso-paramétriques	Zigzag ou unidirectionnel	CL points iso-paramétrique, calculés sur surface offset.	Oui	Oui	Non <sup>4</sup>	Non <sup>4</sup>	non

1- uniquement au niveau des transitions entre surfaces.  
 3- possibles mais non implémentés.

2- uniquement au niveau des frontières de surfaces.

4- Informe l'utilisateur d'une incompatibilité local entre l'outil et la surface.

*2<sup>ème</sup> PARTIE*

*Usinage de formes gauches  
à  
trajectoires optimisées.*

## Chapitre II.1 Système d'usinage élaboré

### 1.1 Aperçu général

Conformément aux critères de performances d'un système de génération automatique de trajectoires d'outils que nous avons établis (Cf. Chapitre 2), nous proposons un système d'usinage 3 axes. Ce dernier permet de générer des trajectoires d'outils pour l'usinage d'ébauche et de finition des formes complexes de degrés et de modèles quelconques, faisant intervenir d'une part le concept de construction flexible [Belaidi '98] basée sur les principes de réduction de degré des surfaces de Bézier, respectivement de conversion exacte des surfaces NURBS et d'autre part une optimisation des trajectoires d'ébauches et de finition. L'introduction d'une surépaisseur maximale de finition et la combinaison simultanée de trajectoires de topologies différentes [Marschall et al. '94a, 94b] représente une nouvelle opportunité d'élimination de passes de demi-finition [Hassaid et Belaidi '98]. Par ailleurs, ce système permet de respecter au mieux les tolérances de corde et de crête imposées pour les opérations de finition, d'éliminer systématiquement les interférences et collisions et de générer les données nécessaires pour une reprise locale d'usinage. Les différents modules constituant le système proposé, dont l'organigramme général est donné en figure 1.1, sont représentés par la figure 1.2, alors que la figure 1.3 donne le détail des fonctions assurées par chaque module.

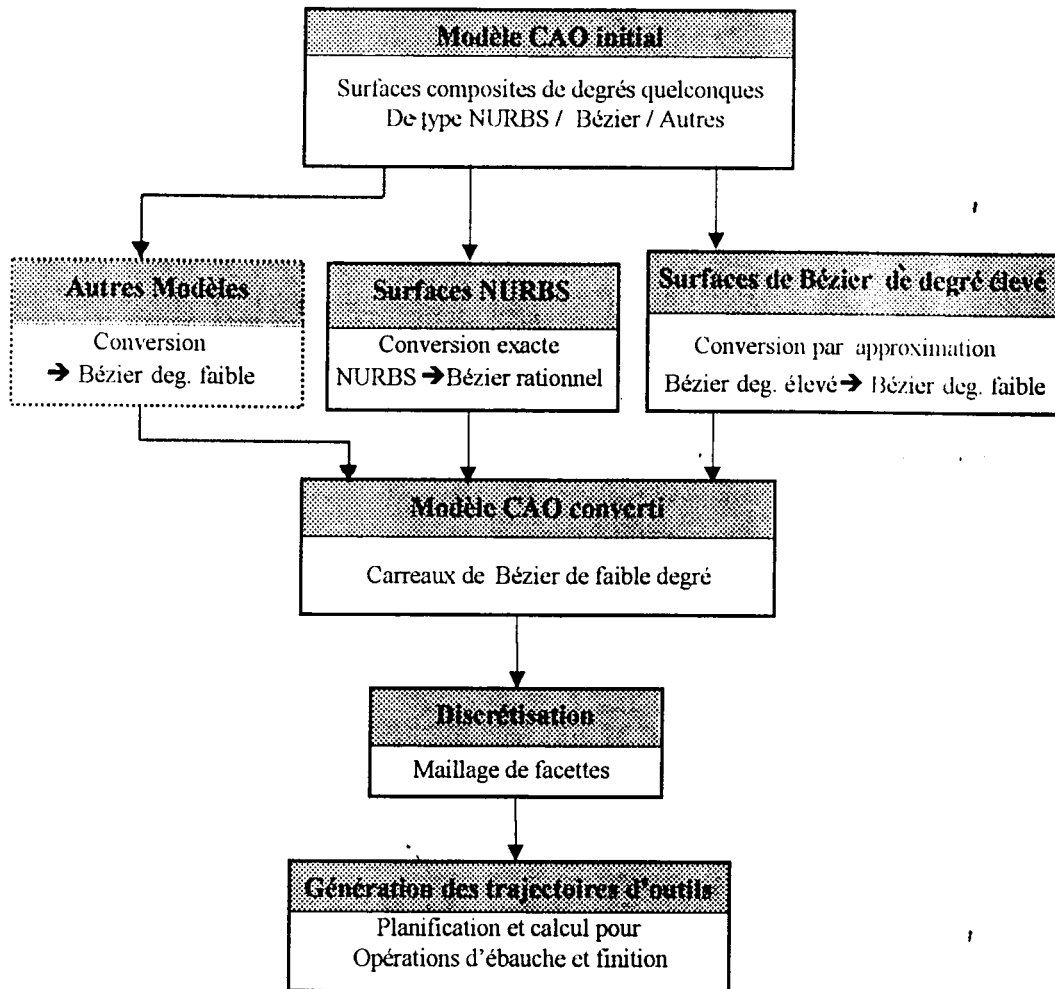


Figure 1.1 Organigramme général du système de CFAO proposé

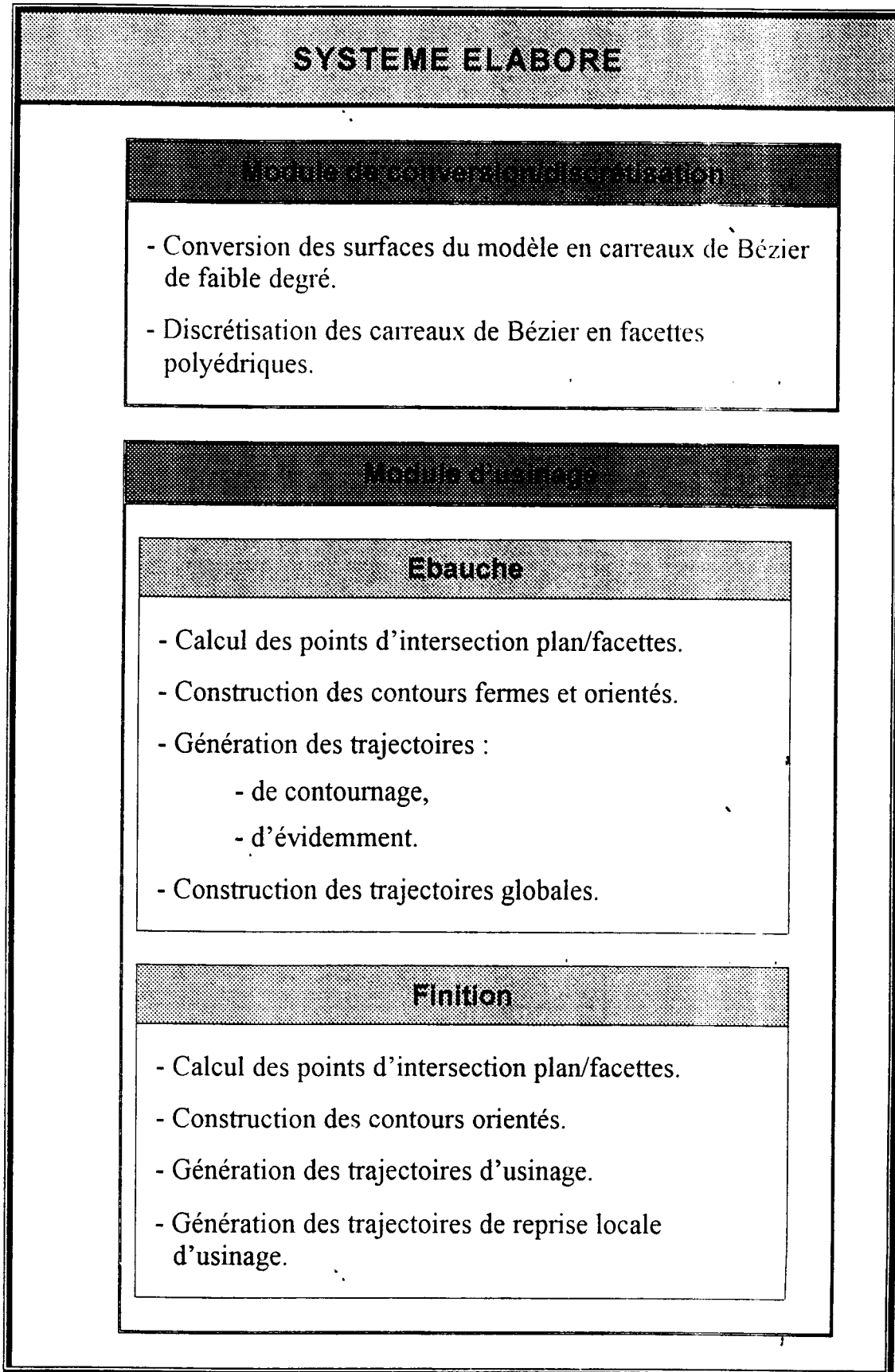


Figure 1.2 Structure du système de CFAO proposé.

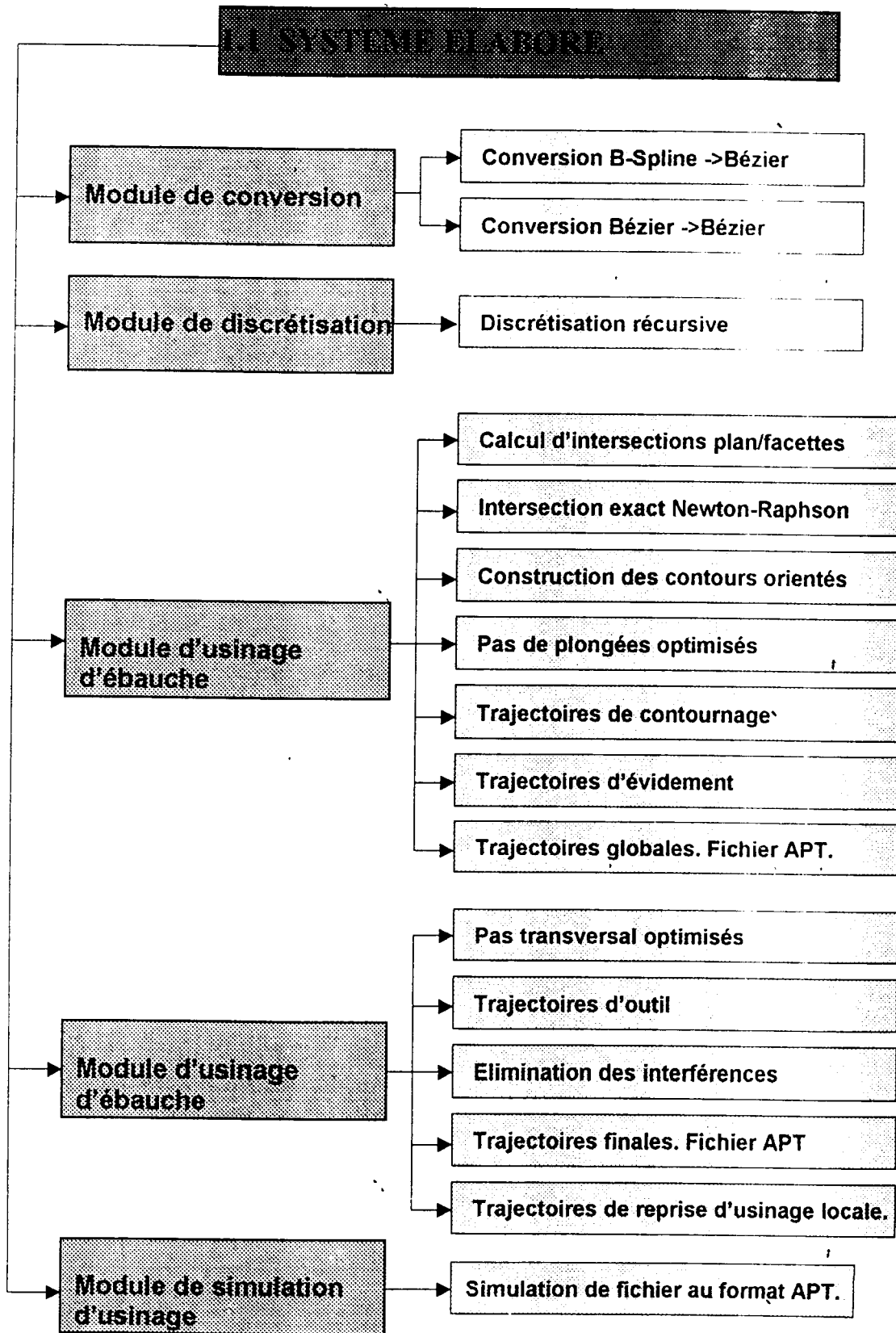


Figure 1.3 Détails des modules du système de CFAO proposé.

## 1.2 Conception flexible de formes complexes

### 1.2.1 Problème posé

La modélisation des formes gauches est basée sur des modèles mathématiques polynomiaux. Outre les modèles B-splines [Léon'91], [Farin'93], [Boehm'81, '87], dont les propriétés géométriques de modélisation sont très appréciées en raison de leur flexibilité dans les manipulations géométriques et leur capacité de représentation exacte des coniques, nous retrouvons aussi le modèle de Bézier implémenté dans de nombreux systèmes de CFAO en raison des nombreuses et très intéressantes propriétés de modélisation géométrique qu'il offre [Bézier'77, '85, '88], [Léon'91], [Farin'93], [Bensalah'90], [De Casteljaou'85]. Toutefois, l'emploi du modèle de Bézier intégral pour la modélisation de formes complexes impose l'utilisation de degré élevé afin de permettre une flexibilité de conception dont l'inconvénient majeur est le nombre élevé d'opérations de calcul qu'il faut effectuer, source d'oscillations parasites non décelables visuellement, [Nougier'93], [Léon'91] qui risquent

- a) de réduire la précision numérique des calculs et engendrer des restrictions importantes au niveau des manipulations graphiques (zoom, ombrage, transformations géométriques diverses...);
- b) d'augmenter les temps de traitement ;
- c) d'altérer la qualité et l'esthétique des surfaces , ce qui nécessite l'intégration de modules d'inspection et de visualisation en vue de corriger les imperfections et alourdit davantage la procédure de construction.

Pour remédier à cet inconvénient, nous pouvons faire appel à deux possibilités de solutions [Belaidi'98] :

### 1.2.2 Solutions conventionnelles

Pour contourner les problèmes relatifs au degré élevé, les concepteurs de systèmes de CFAO recourent généralement à des modeleurs surfaciques utilisant des entités:

1. de type *B-Spline non rationnels* ou *NURBS* en raison des multiples avantages qu'on leur connaît [Léon'91], [Farin'93], [Boehm'84], [Bartels et al.'88]; en prenant en compte la complexité qui en découle ;



2. composites de types Bézier rationnelles ou non rationnelles de degré faible, le plus souvent de degré trois, cinq ou sept, dépassant toutefois rarement le degré dix. Cette deuxième solution implique certains inconvénients tels que :

- les opérations de raccordement et de lissage aux frontières qui sont délicates et consommatrices de temps et pouvant influencer sur la qualité d'usinage pour certains types de trajectoires d'outils

- la nécessité d'intégration d'un outil d'analyse de surfaces incluant une ou plusieurs étapes supplémentaires d'inspection et de visualisation des raccordements, de corrections interactives rendant la démarche fastidieuse et consommatrice de temps [Farin'93], [Léon'91].

### 1.2.3 Solution par la sub-réduction de degré

Les solutions conventionnelles étant à écarter du fait des problèmes cités ci dessus, la solution que nous avons retenue est basée sur l'idée de dissocier les contraintes de souplesse maximum lors de la phase de conception de celles de robustesse maximum et volumes de calculs minimums lors des étapes de calculs des trajectoires d'outils ou toutes autres phases de calculs, pour lesquelles les faibles degrés s'imposent. Cela implique une phase de traitement supplémentaire qui permet de convertir le modèle de degrés élevés en modèle de plus faibles degrés. En effet le processus de sub-réduction [Bensalah'90, '93], [Hoček'88, '89, '90], [Eck'93, '95], [Belaidi'98], [Asma'98], représente une solution qu'il est possible d'exploiter en CFAO, grâce à la flexibilité dans la construction des courbes et surfaces de Bézier de forme complexe que procure le degré élevé, à la possibilité de prescription de tolérances dès l'étape de construction et à une certaine aisance dans les manipulations graphiques et l'exploitation des données géométriques, tout en évitant les inconvénients liés au degré élevé.

Les données géométriques d'un modèle de Bézier de degré faible, issues d'une sub-réduction d'un modèle initial de degré élevé, pourront en effet être facilement exploitées par les modules de transformations et de manipulations graphiques (translation, rotation, couper, limiter, allonger etc.) et d'affichage (ombrage, zoom, loupe etc.) ou par le module d'usinage (calcul d'offset, calcul des

points centre d'outil, calcul d'intersections, d'interférences locales etc.) au sein d'un même système ou par un système externe. L'organigramme général du principe de sub-réduction est donné dans la figure (figure 1.4).

La méthode adaptée dans le contexte du système élaboré repose sur les fondements de calcul d'approximation pour la réduction de degré exposé dans [Asma '98].

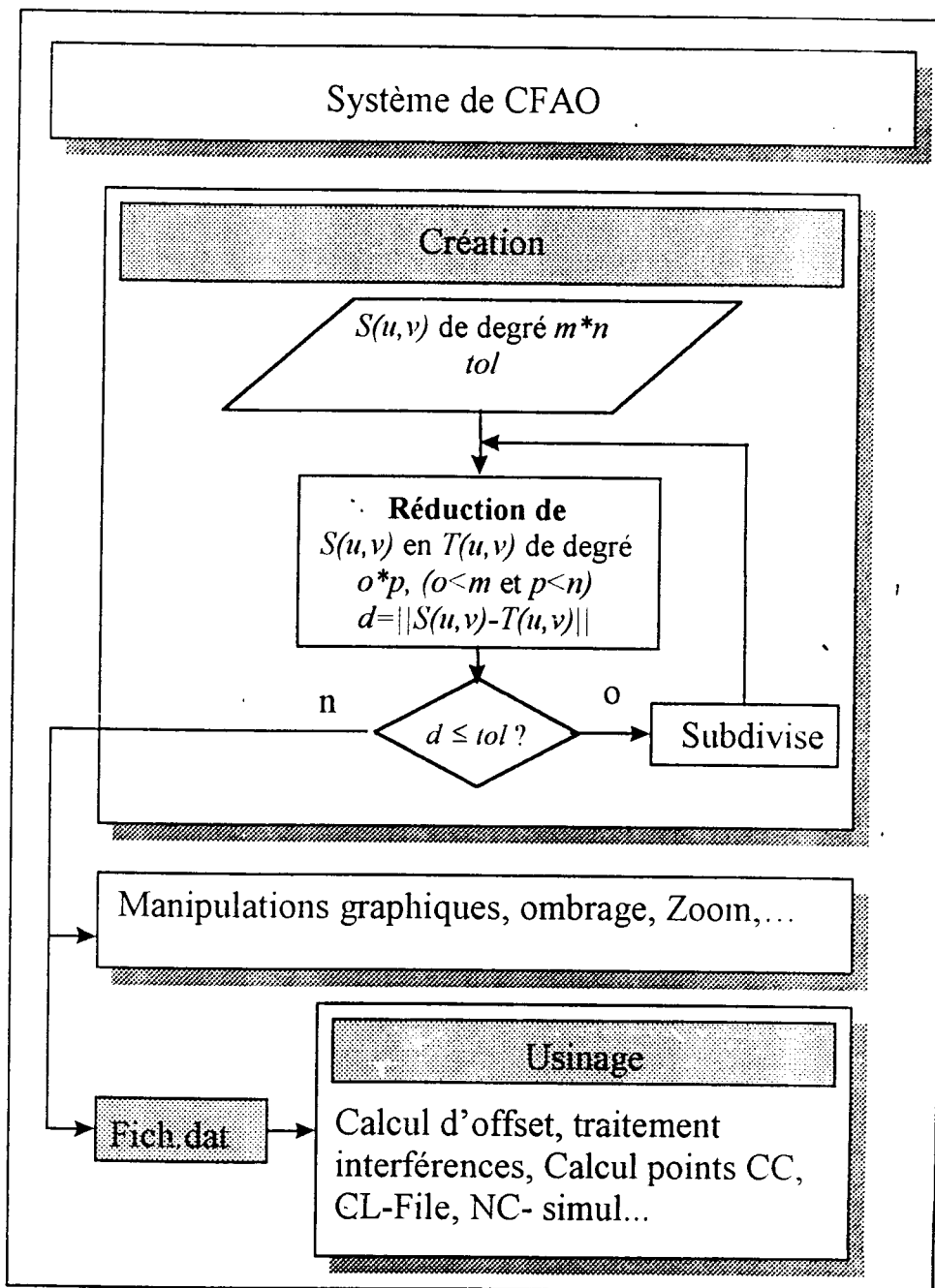


Figure 1.4  
 Principe et applications de la sub-réduction

## 1.3 Méthode d'usinage d'ébauche

La méthode proposée présente l'avantage d'être robuste et la performance de générer des trajectoires d'outils en usinage 2 axes ½ de surfaces gauches composites et d'optimiser ces dernières par rapport au temps de coupe et à la taille de fichiers de points-outil. Sa robustesse est assurée par un algorithme de calcul des courbes d'intersections surface/plan implémenté, basé sur une technique de discrétisation auto-adaptative de la surface en facettes, sous contraintes du respect de tolérances imposées par l'utilisateur. L'optimisation des trajectoires s'effectue dans un premier temps de façon implicite pendant la phase de discrétisation, grâce à son caractère auto-adaptatif et dans une deuxième étape au niveau de la mise en œuvre de la méthode d'évidement de poche introduite ci-dessous. Cette dernière utilise des trajectoires d'outils parallèles en zigzag prenant en compte l'optimisation des déplacements à vide de l'outil.

### 1.3.1 Présentation générale

La première phase du processus de traitement consiste à convertir les carreaux de surfaces du modèle surfacique de l'objet en carreaux de Bézier de faible degré. Les carreaux de surfaces NURBS sont dans un premier temps convertis en carreaux de Bezier, puis la discrétisation s'effectue par subdivision récursive de ces derniers. Le test d'arrêt dépend du respect des tolérances imposées par l'utilisateur et le résultat final est ensuite mémorisé dans une structure dynamique de type arbre quaternaire.

L'algorithme de calcul des points d'intersection surface/plan de coupe est de type hybride, combinant les techniques de recherche d'intersection par subdivision à celles de raffinement des solutions par des méthodes non linéaires. A partir de l'intersection plan de coupe/facettes, des points initiaux qui définissent les segments des portions de courbes d'intersections sont calculés. Une approximation des paramètres  $u$  et  $v$  est ensuite effectuée sur la base de l'hypothèse d'une répartition uniforme de ces derniers. Le résultat est ensuite affiné grâce à un algorithme non linéaire de type Newton- Raphson [Nougier '93], [Gourdin et Boumahrat '91].

Pour chaque plan de coupe, les contours extérieurs et les îlots sont alors

construits à partir de l'analyse des relations topologiques qui lient les différents segments. Enfin, une nouvelle approche d'évidement des poches ainsi obtenues, qui élimine la passe de demi-finition et optimise les déplacements à vide, est proposée. La figure 1.5 présente l'organigramme général de cette méthode.

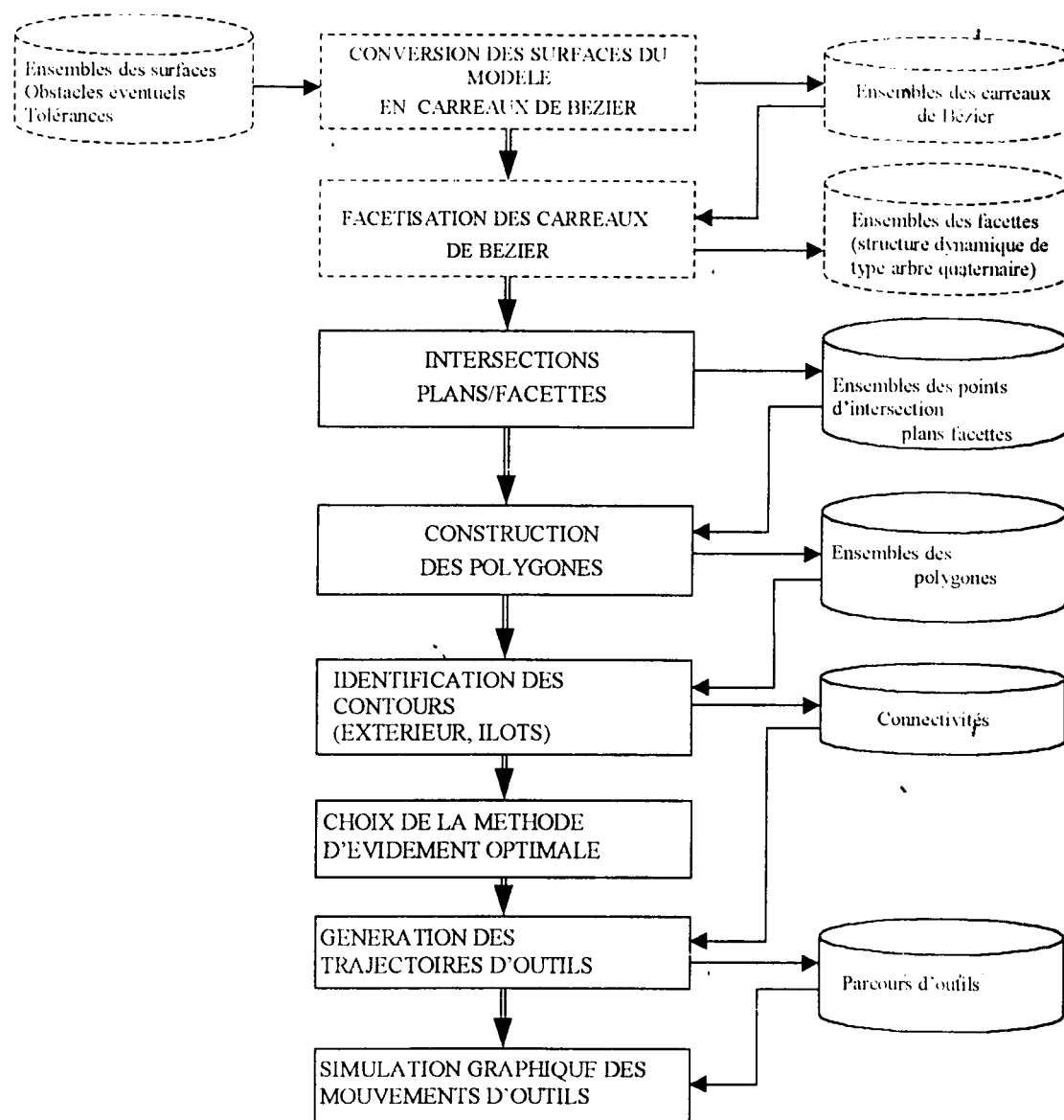


Figure 1.5 Organigramme général de la méthode d'ébauche proposée.

### 1.3.2 Choix d'un type de trajectoires

Pour la phase d'ébauche, les critères les plus importants sont le débit maximum de copeau, la réduction des temps improductifs et un minimum de restriction quant aux formes des surfaces pouvant être traitées. Il ressort de l'analyse précédente (Cf. 1<sup>er</sup> Partie, chapitre 2) que les techniques d'ébauche en 2 axes  $\frac{1}{2}$  sont les plus appropriées. Il reste néanmoins à choisir, parmi les différentes stratégies de planification des trajectoires, celles qui peuvent être utilisées avec ce genre d'usinage. Il est possible en effet d'utiliser tous les types de trajectoires planes citées précédemment. Si on élimine les trajectoires par *plans parallèles*, parallèles à l'axe de l'outil, similaires à celles utilisées par Loney et Ozoy [Loney et Ozoy'87] (figure 1.6), du fait des passes à vide importantes, il reste les trajectoires en *spirales par décalage* ou par *réduction de contours*, les trajectoires *parallèles en zigzag* ou *unidirectionnelles* et les *courbes de remplissages*.

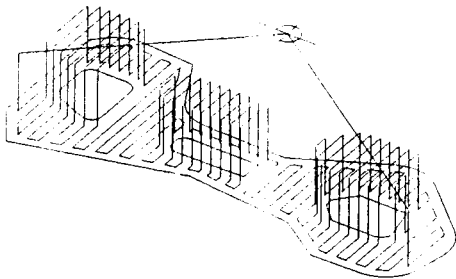


Figure 1.6  
Déplacements à vide  
excessifs des trajectoires en  
zigzag parallèles.

Les trajectoires parallèles *unidirectionnelles* ont été écartées en raison des déplacements à vide importants qu'elles imposent et qui ne se justifient pas en ébauche. Les courbes de remplissage sont préconisées en finition du fait de leur auto-adaptativité à la forme locale de la surface. En ébauche, elles se ramènent pratiquement à des trajectoires parallèles avec des complications supplémentaires au niveau des calculs. Les trajectoires en spirales par décalage ou par réduction de contours sont intéressantes mais posent un problème majeur : le volume de calculs est en effet important lors de la construction des trajectoires à partir des contours décalés. Ces calculs sont d'autant plus importants que les contours sont complexes et que le nombre d'îlots augmente. De plus, une optimisation du nombre de

plongées ainsi que les parcours à vide qu'elles imposent pour les transitions entre plongées est nécessaire (figure 1.7).

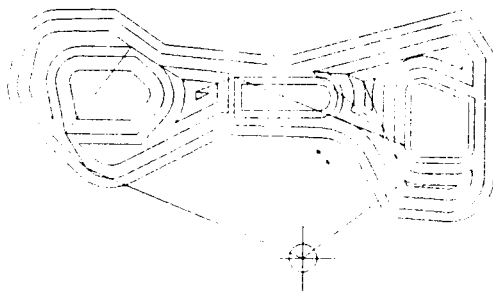


Figure 1.7  
Déplacements à vide pour  
la transition entre sous  
poches, trajectoires par  
décalage de contours.

Les trajectoires parallèles en zigzag sont les plus robustes, néanmoins leur construction augmente en complexité avec la présence d'ilots et impose une optimisation des parcours à vide pour les transitions entre plongées (figure 1.8a et 1.8b). Ce type de trajectoires possède, par rapport aux précédentes, une caractéristique très intéressante, à savoir la possibilité du choix de la direction de balayage permettant d'optimiser le parcours total de l'outil [Prabhu et al.'90], [Lakkaraju et al.'92], [Sun et Tsai'94]. Par conséquent, notre choix s'est porté sur ce dernier type de trajectoires.

Une fois la topologie par plans parallèles, perpendiculaires à l'axe de l'outil est choisie, le calcul des trajectoires se ramène essentiellement à la détermination des courbes d'intersection des carreaux du modèle avec les plans de coupe successifs. La méthode de calcul proposée est une méthode hybride combinant les techniques de subdivision [Peng'84], [Leon'91] et les méthodes non linéaires de type Newton-Raphson [Leon'91], [Barnhill et al.'87], [Barnhill et al'90], [Rubio'93], [Grandine et Klein'97].

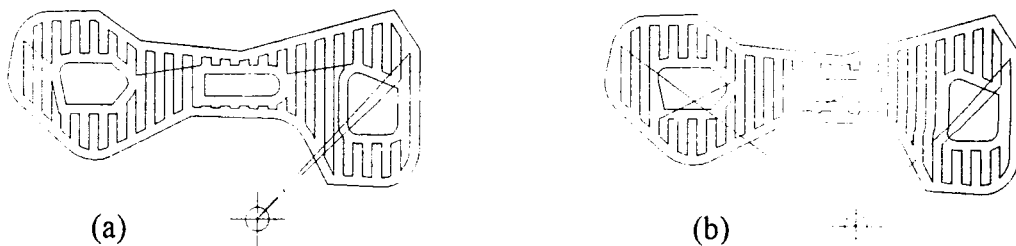


Figure 1.8 Optimisation des déplacements  
à vide des trajectoires en zigzag parallèles.

### 1.3.3 Discrétisation des surfaces du modèle

#### 1.3.3.1 Principe de discrétisation

Afin de garantir la robustesse de l'algorithme de calcul des points d'intersection, les carreaux composant le modèle sont discrétisés en facettes. Le processus de discrétisation consiste en une subdivision récursive de chaque carreau en quatre sous carreaux jusqu'à satisfaction de la tolérance imposée par l'utilisateur ( Cf. Section 1.3.3.3).

Cette technique présente l'avantage d'être auto-adaptative (Figure 1.9) et garantit ainsi le respect des tolérances imposées en optimisant au mieux le nombre de facettes. Le calcul des trajectoires de finition et de reprises locales d'usinage éventuelles se basera aussi sur ce nombre minimal de facettes ainsi obtenues.

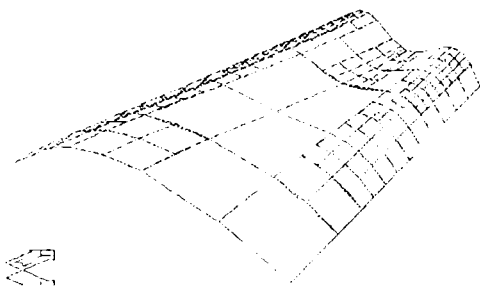


Figure 1.9  
Discretisation auto-  
adaptative d'un carreau de  
Bézier

#### 1.3.3.2 Subdivision d'un carreau de Bézier

##### A- Subdivision d'une courbe de Bézier rationnelle

L'algorithme de récurrence de De Casteljeau [De Casteljeau '86], [Bensalah '90], [Leon '91], permet de subdiviser une courbe de Bézier rationnelle  $C(t)$  de degré  $n$  définie par ses  $(n+1)$  pôles  $\{P_0, \dots, P_n\}$  chargés respectivement par les poids  $(w_0, \dots, w_n)$  au paramètre  $t_0$  en deux courbes de Bézier rationnelles  $R(t)$  et  $Q(t)$ , toutes les deux de degré  $n$  et ayant le point commun  $R(1) = Q(0) = C(t_0)$ .

Les pôles des polygones caractéristiques,  $\{R_0, \dots, R_n\}$  et ses charges  $(\alpha_0, \dots, \alpha_n)$  et  $\{Q_0, \dots, Q_n\}$  et ses charges  $(\beta_0, \dots, \beta_n)$  sont données par les relations de récurrence [1.1] et [1.2].

$$S_i^j(u^*) = (1 - u^*) \frac{w_i^{j-1}(u^*)}{w_i^j(u^*)} S_i^{j-1}(u^*) + u^* \frac{w_{i+1}^{j-1}(u^*)}{w_i^j(u^*)} S_{i+1}^{j-1}(u^*) \quad [1.1]$$

$$\varphi_i^j(u^*) = (1 - u^*) w_i^{j-1}(u^*) + u^* w_{i+1}^{j-1}(u^*) \quad [1.2]$$

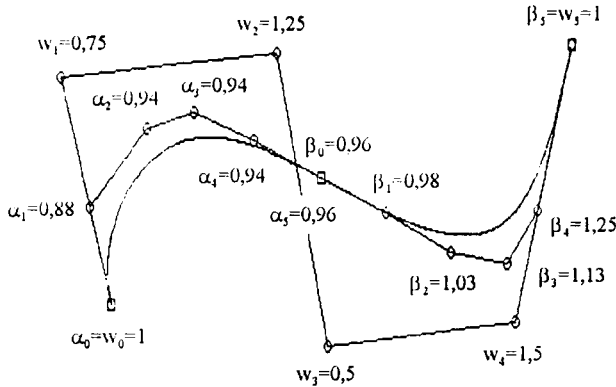


Figure 1.10  
 Courbe de Bézier rationnelle  
 deg.5 subdivisée en deux  
 courbes rationnelles deg.5  
 [Asma'98].

## B- Subdivision d'une surface de Bézier rationnelle

La subdivision d'un carreau de Bézier  $S(u,v)$ , défini par son réseau caractéristique  $P_{m,n}$  pondéré par les poids  $w_{m,n}$ , aux paramètres  $(u^*, v^*)$ , consiste à appliquer l'algorithme de subdivision des courbes de Bézier successivement aux  $(m+1)$  polygones caractéristiques dans la direction  $u$ , au paramètre  $u^*$ , puis aux  $2*(n+1)$  polygones caractéristiques ainsi obtenus, dans la direction  $v$ , au paramètre  $v^*$ . L'algorithme élaboré à cet effet est présenté en détail au chapitre 11.2.

## C- Subdivision d'une surface NURBS

Dans ce cas, il est nécessaire de procéder à une conversion de la surface NURBS en carreaux de Bézier, afin de pouvoir exploiter les résultats de la subdivision précédemment établis. Sur le plan mathématique, toute surface B\_Spline rationnelle de degré  $(m,n)$ , ayant  $k$  nœuds internes de multiplicité  $S_i$  en  $u$  et  $l$  nœuds internes de multiplicité  $P_i$  en  $v$ , est le raccordement de  $(k+1)*(l+1)$  surfaces rationnelles de Bézier de degré  $(m,n)$  avec des continuités  $(m-P_i)$  suivant les frontières en  $u$ , et des continuités  $(n-P_i)$  suivant les frontières en  $v$  [Bensalah'90].



### 1.3.3.3 Tolérances de discrétisation

Le test d'arrêt de la discrétisation consiste à évaluer l'erreur de discrétisation représentant la déviation maximale entre la surface du modèle, converti éventuellement, et les facettes. Plusieurs méthodes d'évaluation plus ou moins précises sont proposées [Barmhill et al.'87], [Houghton et al.'85], [Hegron'85], [Bret'88], [Kim et Ko'94]. Dans tous les cas, la meilleure approximation de cette erreur nécessite des temps de traitement considérables. Nous retenons entre autres deux méthodes. La méthode de calcul par la *boite mini-maxi* de la facette gauche qui consiste à comparer sa hauteur à la tolérance imposée (figure 1.11), mais le problème du choix de l'orientation des faces se pose. Généralement on adopte des faces parallèles aux plans principaux d'un repère cartésien local, dont l'axe  $\bar{z}$  est parallèle au vecteur normal calculé au centre du carreau/facette ( $u=0.5, v=0.5$ ).

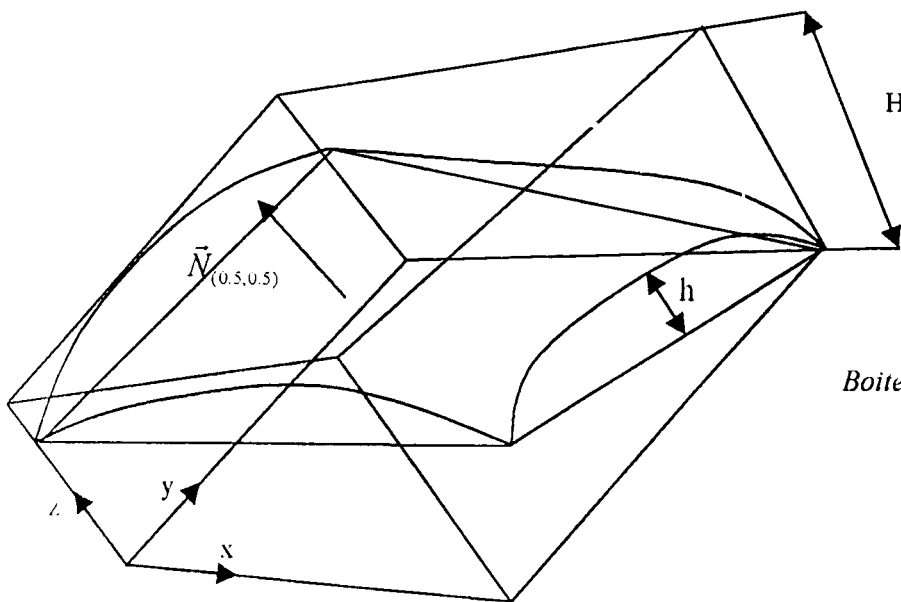
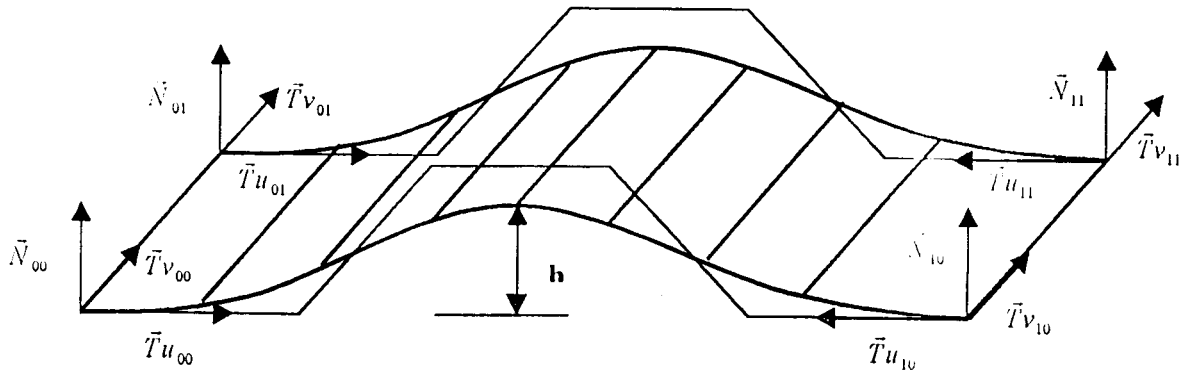


Figure 1.11  
 Boite mini-maxi d'une facette gauche

Pour un carreau fortement courbé, cette méthode donne une estimation majorant fortement l'erreur réelle ( $H \gg h$ ). La deuxième méthode, très utilisée [Bernhill et Kersey'90], [Leon'91], [Rubio'93], [Houghton et al.'85] consiste à évaluer l'erreur de linéarité des bords et de planéité du carreau/facette. Cette méthode est certes adaptée à l'infographie, mais n'est pas conforme au langage du mécanicien, puisque

l'erreur est exprimée en terme de variation d'angle. De plus, pour les carreaux de haut degré, elle peut donner des résultats erronés (figure 1.12).



$$[1 - (\bar{T}u_{ij} \cdot \bar{T}u_{ij})] = 0 \quad \text{et} \quad [1 - (\bar{N}_{ij} \cdot \bar{N}_{ij})] = 0$$

Figure 1.12 Exemple de singularité de la méthode d'évaluation de linéarité des bords et planéité de la facette

Son implémentation nécessite l'évaluation des vecteurs tangents et normaux aux quatre coins de la facette gauche (figure 1.13). On vérifie l'erreur de linéarité des bords et l'erreur de planéité de la facette en appliquant les relations

$$\max[1 - (T_{ij} \cdot T_{kl})] < Tol\_Lin \quad \max[1 - (N_{ij} \cdot N_{kl})] < Tol\_Plan$$

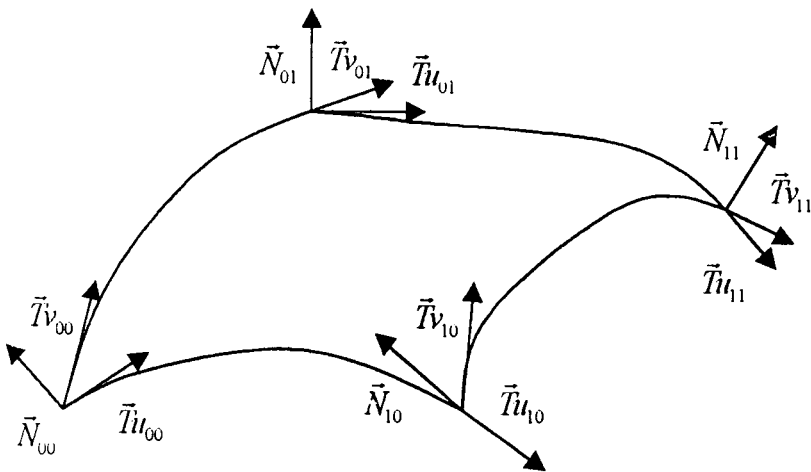


Figure 1.13 Planéité et linéarité des bords à l'aide des vecteurs normaux et tangents aux coins

La méthode proposée est une combinaison des deux précédentes. Dans une première phase de pré-discrétisation, cette méthode permet d'appliquer le critère de linéarité/planéité avec une tolérance large pour aplanir les carreaux/facettes. Ensuite, l'ensemble des carreaux ainsi obtenus sera discrétisé en conformité avec le critère de la hauteur de la boîte mini-maxi. La cohérence des résultats est garantie dans la mesure où seules les surfaces à degré faible sont traitées dans notre cas.

Les figures 1.14a et 1.14b montrent un exemple de subdivision récursive auto-adaptative d'un carreau de Bézier de degré (6x6) avec des tolérances de discrétisation respectivement de 0.25mm et 0.025mm.

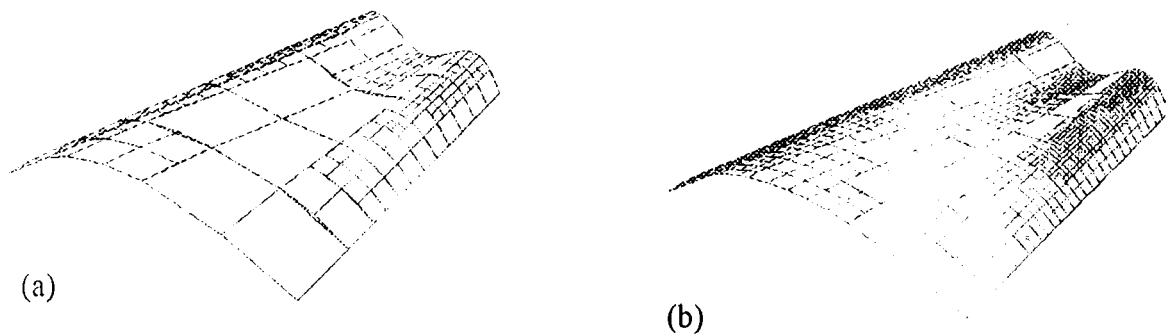


Figure 1.14 Discretisation auto adaptative d'un carreau de Bézier

### 1.3.4 Intersection plan de coupe /modèle

La méthode proposée repose fondamentalement sur l'algorithme de calcul des points d'intersection, car la robustesse globale du système en dépend en grande partie. Cet algorithme doit à cet effet jouir des propriétés telles que mentionnées dans [Rubio '93], [Kim et Kim '95], [Barmhill et al. '87], [Barmhill et al. '90], [Hougion et al. '85], à savoir :

- Garantir un résultat numériquement précis ;
- Etre Fiable et robuste, c'est-à-dire insensible à des configurations particulières présentant des singularités ;

- Capable d'obtenir toutes les solutions du problème d'intersections et d'identifier les singularités ;
- Etre suffisamment rapide pour ne pas compromettre la performance du système.

Pour remplir toutes ces conditions, nous avons mis au point un algorithme qui calcule, à partir du modèle discrétisé, les points d'intersection surface/plan en deux phases de traitement. La première phase permet de déterminer les points approchés à partir de l'intersection facettes/plan. Ces derniers définissent les extrémités des segments d'intersection pour lesquels une approximation des paramètres  $u$  et  $v$  est effectuée sur la base d'une répartition uniforme. Un changement de repère, tel que le plan de coupe soit parallèle au plan  $(xy)$  du nouveau repère, permet de réduire considérablement le volume de calcul [Rubio '93]. Cette première phase de traitement permet de localiser avec rapidité et précision toutes les solutions initiales du problème d'intersection et d'identifier des singularités éventuelles. Dans la deuxième phase, les résultats précédents seront ensuite affinés grâce à un algorithme itératif de résolution d'équation non linéaire de type Newton- Raphson [Nougier '93] qui permet d'avoir la solution exacte en quelques itérations grâce à la bonne capacité de localisation des solutions initiales, à l'instar des résultats rapportés dans [Faux et Pratt '87], [Nougier '93], [Rubio '93], [Hatna '95].

#### 1.3.4.1 Transformation de coordonnées

Etant donné que tous les plans de coupe sont parallèles, il convient de procéder à un changement de repère tels que ces derniers soient parallèles à l'un des plans principaux du nouveau repère. Dans ce cas, la détermination des facettes susceptibles d'être coupées par l'un des plans se ramène à une simple comparaison de l'une des trois abscisses des sommets extrêmes du carreau, ce qui permet de réduire considérablement les temps de calcul. On adoptera un nouveau repère  $(O,xy,z)$  tel que l'axe  $z$  soit parallèle à la normale au plans de coupes  $N(a,b,c)$ , (Figure 1.15).

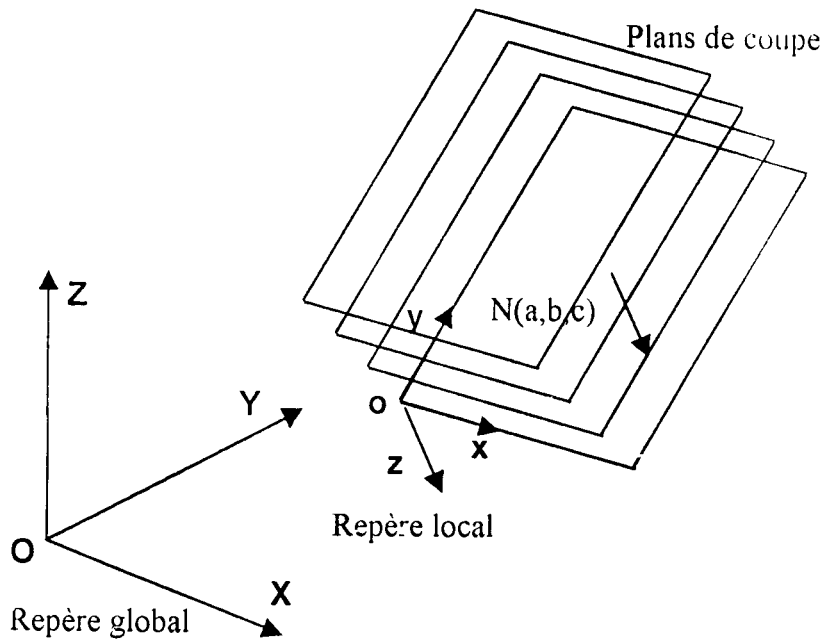


Figure 1.15 Repère local lié aux plans de coupe

Les relations de passage du repère (O,X ,Y ,Z) au repère (O,xy,z) sont alors exprimées par

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a & -b & -c \\ b & 1 - \frac{b^2}{1+a} & -\frac{bc}{1+a} \\ c & -\frac{bc}{1+a} & 1 - \frac{c^2}{1+a} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$\vec{N}(a, b, c)$  vecteur normal unitaire.

$$a^2 + b^2 + c^2 = 1$$

Dans le cas particulier où les plans sont parallèles au plan (X,Y), la matrice de passage devient la matrice identité.

#### 1.3.4.2 Calcul des points d'intersection *plan de coupe/modèle facettisé*

L'algorithme de calcul des points d'intersection *plan de coupe/modèle facettisé* doit permettre de calculer tous les points d'intersections entre le plan de coupe et l'ensemble des facettes du modèle. De plus, il doit fournir des valeurs

approchées des paramètres  $(u, v)$  de chaque point d'intersection et identifier les singularités éventuelles tel que points doubles, coté de facette confondue avec le plan de coupe, etc. La rapidité de calcul est garantie grâce à une phase d'identifications des facettes pouvant éventuellement avoir une intersection avec le plan. Après une transformation de repère automatique permettant de ramener le plan de coupe parallèle au plan  $(x, y)$  local de la facette, cette identification revient à une simple comparaison des cotes  $z$  des quatre sommets avec la cote  $z_p$  du plan de coupe. Une fois l'identification des facettes effectuée, les tests de détection des singularités permettent de calculer dans tous les cas les points. Ensuite, le coté de la facette correspondant à l'intersection concernée est identifié afin d'approximer les paramètres  $(u^*, v^*)$  sur la base d'une répartition uniforme des valeurs paramétriques. L'exemple de la figure 1.13. montre deux points d'intersections plan/facette  $P_1^*$  et  $P_2^*$ , qui représentent les solutions approchées correspondant aux points d'intersection exactes plan/surface  $P_1$  et  $P_2$ , pour les valeurs paramétriques  $(u_1, v_1)$  et  $(u_2, v_2)$ .

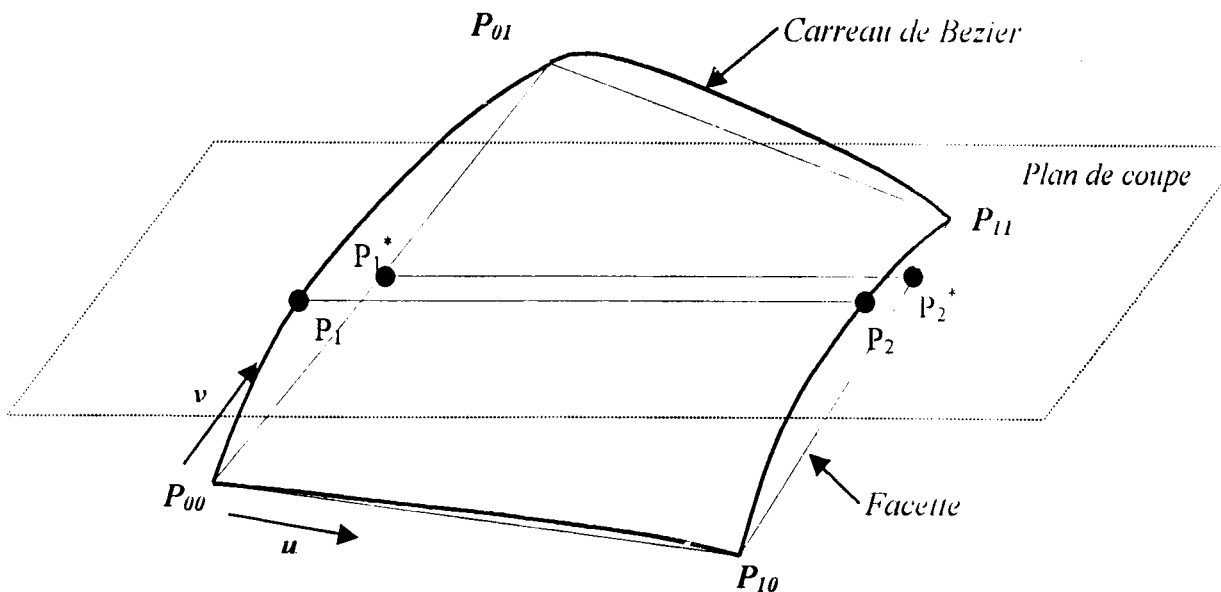


Figure 1.16 Déterminations des paramètres approchés  $(u^*, v^*)$  des points d'intersection  $P_1$  et  $P_2$ .

L'identification des cotés nous permet de connaître déjà les valeurs exactes de  $u_1$  et  $u_2$ , puisque les points sont sur les courbes frontières, donc  $u_1 = 0$  et  $u_2 = 1$ . Les

autres paramètres  $v_1$  et  $v_2$  seront approximés respectivement par  $v_1^*$  et  $v_2^*$  en supposant une répartition uniforme, en appliquant les relations ci-dessous.

Ainsi pour  $P_1$ , nous avons :

$$\frac{v_1^* - v_0}{v_1 - v_0} = \frac{|\bar{P}_1^* - \bar{P}_{00}|}{|\bar{P}_{01} - \bar{P}_{00}|}$$

avec  $v_0=0$  et  $v_1=1$ , cette dernière se ramène à :

$$v_1^* = \frac{|\bar{P}_1^* - \bar{P}_{00}|}{|\bar{P}_{01} - \bar{P}_{00}|}$$

et pour  $P_2$ , nous avons :

$$\frac{v_2^* - v_0}{v_1 - v_0} = \frac{|\bar{P}_2^* - \bar{P}_{10}|}{|\bar{P}_{11} - \bar{P}_{10}|}$$

de même avec  $v_0=0$  et  $v_1=1$ , nous obtenons :

$$v_2^* = \frac{|\bar{P}_2^* - \bar{P}_{10}|}{|\bar{P}_{11} - \bar{P}_{10}|}$$

### 1.3.4.3 Calcul des points d'intersection exacts

Pour le calcul des points d'intersection exacts, nous devons déterminer les valeurs des paramètres  $u$  et  $v$  correspondants au point d'intersection entre le plan de coupe  $P(s, t)$  et le carreau de Bézier  $S(u, v)$ . Comme les points d'intersections appartiennent toujours aux courbes frontières du carreau/facette, l'un des deux paramètres est déjà connu au moment de la localisation des solutions et prend la valeur zéro ou un. Le calcul du point d'intersection se ramène alors à la résolution de l'équation

$$S(u, v) - P(s, t) = 0 \Leftrightarrow \begin{cases} X(u, v, s, t) = X_s(u, v) - X_p(s, t) = 0 \\ Y(u, v, s, t) = Y_s(u, v) - Y_p(s, t) = 0 \\ Z(u, v, s, t) = Z_s(u, v) - Z_p(s, t) = 0 \end{cases} \quad [1.3]$$

Dans le repère local lié au plan de coupe définie précédemment. l'équation du plan de coupe s'écrit

$$\begin{cases} x_p(s, t) \\ y_p(s, t) \\ z_p \end{cases} \quad [1.4]$$

Avec [1.3], L'équation [1.4] devient ainsi

$$S(u, v) - P(s, t) = 0 \Leftrightarrow \begin{cases} X(u, v, s, t) = X_s(u, v) - X_p(s, t) = 0 \\ Y(u, v, s, t) = Y_s(u, v) - Y_p(s, t) = 0 \\ Z(u, v) = Z_s(u, v) - Z_p = 0 \end{cases}$$

L'un des deux paramètres étant connu, l'équation unique

$$Z(u, v) = Z_s(u, v) - Z_p = 0$$

permet de calculer le second paramètre en appliquant la méthode de Newton-Raphson. Le développement de Taylor d'ordre un donne alors

$$Z_s(u, v) - Z_p + \delta u \frac{\partial Z_s(u, v)}{\partial u} + \delta v \frac{\partial Z_s(u, v)}{\partial v} = 0. \quad [1.5]$$

Si le point d'intersection appartient à une courbe frontière iso-paramétrique en  $u$

alors  $u = cste.$  et  $\frac{\partial Z_s(u, v)}{\partial u} = 0$ , l'équation [1.5] se simplifie et devient

$$Z_s(u_{cste}, v) - Z_p + \delta v \frac{\partial Z_s(u_{cste}, v)}{\partial v} = 0 \Leftrightarrow \delta v = \frac{Z_p - Z_s(u_{cste}, v)}{\frac{\partial Z_s(u_{cste}, v)}{\partial v}}$$

Cette expression permet de calculer itérativement  $v$  en posant :

$$v = v^*$$

Tant que  $|Z_s(u_{cste}, v) - Z_p| > Tolerance$  Faire :

$$v = v + \delta v$$



Lorsque le point d'intersection appartient à une courbe frontière iso-paramétrique en  $v$ , le calcul de la valeur du paramètre  $u$  se fera de la même façon. Dans ce cas, l'expression [1.5] devient alors

$$Z_s(u, v_{cste}) - Z_p + \delta u \frac{\partial Z_s(u, v_{cste})}{\partial u} = 0 \Leftrightarrow \delta u = \frac{Z_p - Z_s(u, v_{cste})}{\frac{\partial Z_s(u, v_{cste})}{\partial u}}$$

et permet de calculer itérativement  $u$  en posant également :

$$u = u^*$$

Tant que  $|Z_s(u, v_{cste}) - Z_p| > Tolerance$  Faire :

$$u = u + \delta u$$

### 1.3.5 Construction des contours

#### 1.3.5.1 Etablissements des contours

Après le calcul des points d'intersections, il faut ensuite construire les courbes d'intersection à partir d'une analyse des relations topologiques qui lient les différents segments d'intersection. L'algorithme que nous avons développé à cet effet permet d'orienter systématiquement le contour dans le sens du premier segment.

#### 1.3.5.2 Orientation des contours

Le calcul des trajectoires d'outils à partir des contours d'intersections construits précédemment impose que ces derniers soit fermés et orientés. Le problème d'orientation des contours fermés est assez complexe et a fait l'objet de nombreux travaux de recherches [Filip'90], [Milenkovic'93], [Hatna'95]. L'orientation des contours est nécessaire pour connaître le sens de décalage lors de la construction des trajectoires de contournage. Parmi les différentes approches développées nous citerons en particulier :

- Celles basée sur l'évaluation de l'aire délimitée par un contour fermé. En effet, conformément au théorème de Green [Hatna'95], si cette aire est positive, alors le contour est orienté dans le sens horaire, si non il est orienté dans le sens trigonométrique.

- Celles basées sur le calcul des angles de variation des tangentes dans chacune des courbes du contour et la variation des tangentes aux points de jonction entre ces courbes, en effet, telle que le démontre le théorème des tangentes tournantes [Hatna '95], si la somme de ses angles est égale à  $2\pi$ , alors le contour est orienté dans le sens trigonométrique, si non la somme est alors égale à  $-2\pi$ , et le contour sera orienté dans le sens horaire.

- Celles qui utilisent le théorème de Jordan énoncé comme suit : « Etant donné une demi-droite  $R$  d'origine  $P$ , dirigée vers l'infini, et un contour fermé  $C$ ; Si  $R$  coupe le contour  $C$  en un nombre paire d'intersections, alors le point  $P$  est à l'extérieur du contour  $C$ . Si non il est à l'intérieur de ce dernier ». Tel qu'énoncé, le théorème de Jordan trouve beaucoup d'application en infographie [Peroche et al. '90], [Martin et al. '89]; Par Contre, pour pouvoir l'utiliser afin d'orienter des contours fermés, il sera nécessaire de formuler une conséquence du théorème comme suit : « étant donnée une demi-droite  $R$  dont le point origine  $P$  est un point de l'une et une seule des courbes du contour et formant un angle positif avec la tangente à cette courbe en ce point, si la droite  $R$  coupe le contour  $C$  en un nombre paire d'intersections, alors le contour  $C$  est orienté dans le sens trigonométrique. Si non il est orienté dans le sens horaire ».

### Méthode proposée

La méthode que nous proposons s'inspire du théorème de Jordan et permet simultanément la construction du contour fermé et son orientation, grâce à l'algorithme de construction des contours d'intersections (Cf. Chapitre 2). Ce dernier permet d'orienter automatiquement le contour total dans le sens du premier segment retenu sans calculs supplémentaires, pour peu que celui-ci soit déjà orienté dans le sens désiré. Par conséquent, une convention d'usinage doit être adoptée par l'utilisateur pour définir un sens de parcours de l'outil par rapport au contour usiné. L'outil peut ainsi être positionné à droite (figure 1.17a) ou à gauche du contour (figure 1.17b).

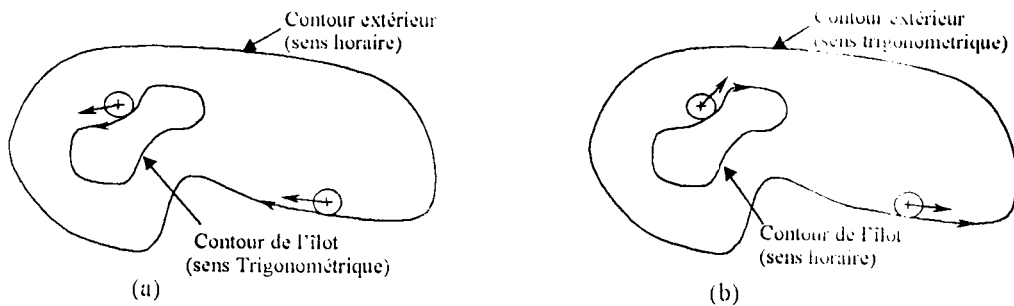


Figure 1.17 Orientation des contours. (a) convention de l'outil à droite et (b) convention de l'outil à gauche.

Dans le cas particulier d'usinage d'ébauche par plans parallèles, il convient d'orienter les contours extérieurs et ceux des îlots, obtenus précédemment, en sens opposés. La convention de l'outil à droite du contour impose une orientation des contours extérieurs dans le sens horaire et ceux des îlots dans le sens trigonométrique (figure 1.17a), dans le cas contraire, les contours extérieurs seront orientés dans le sens trigonométrique et ceux des îlots dans le sens horaire (figure 1.17b).

### 1.3.6 Génération des trajectoires d'outil

Lors de cette phase, les points de passage exacts que doit décrire le point de contrôle de l'outil sont calculés pour permettre de générer les trajectoires d'usinage sous forme d'ordres de mouvements qui seront ensuite transmis au Directeur de Commande Numérique « DCN » de la MOCN.

Pour palier aux problèmes relatifs aux limitations des systèmes classiques d'usinage d'ébauche  $2^{1/2}$  axes (Cf. 1<sup>er</sup> Partie, Chapitre 2), nous proposons une nouvelle technique de planification des trajectoires d'outils qui combine les trajectoires parallèles en zigzag avec optimisation des déplacements à vide et les trajectoires par décalage de contours (Figure 1.18a et 1.18b) répondant aux critères de performances requis (Cf. 1<sup>er</sup> Partie, Chapitre 2). En effet, cette technique d'usinage repose sur une discrimination entre les portions de trajectoires d'outil où ce dernier contourne la forme de la surface finale et celle où l'outil ne fait qu'un évidement de régions massives de matière brute. Cette discrimination nous a permis l'application simultanée de critères d'optimisation antagoniste, à savoir un débit maximum de copeau pour les zones d'évidement et le contrôle des régularités des

stries laissées par l'outil au niveau des zones de contournage. De plus, la méthode proposée donne la possibilité d'élimination de la passe de demi-finition.

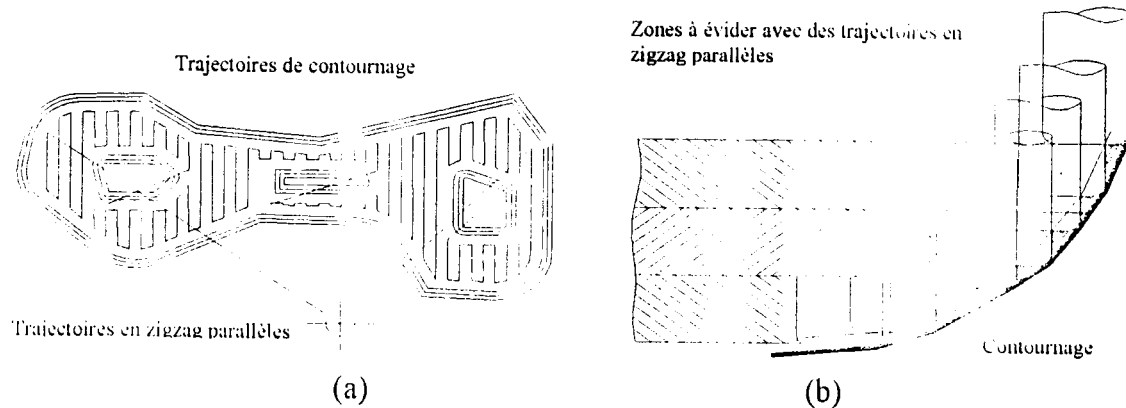


Figure 1.18 : Méthode de planification des trajectoires proposée.

Pour le calcul des trajectoires de contournage, nous avons introduit un nouveau concept, à savoir la *Tolérance de Surépaisseur de Finition ou TSF* qui représente la surépaisseur de finition maximale tolérée (figure 1.19) et qui servira de critère d'optimisation lors du calcul du pas de plongée. Par contre, dans le cas des trajectoires d'évidement en zigzag parallèles, il est possible d'implémenter une stratégie d'optimisation des conditions de coupe en conformité avec les critères d'optimisation souhaitée. Dans notre cas c'est la puissance de broche disponible qui sera retenu comme critère d'optimisation de la profondeur de passe. Les différentes passes de contournage sont ensuite combinées à celles en zigzag parallèles, en vue de construire les trajectoires d'outils finales.

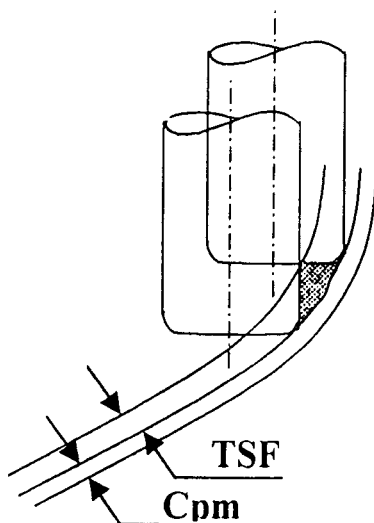


Figure 1.19  
Tolérance de Surépaisseur  
de Finition (TSF).

Il faut noter que la *TSF* fourni à l'utilisateur une souplesse supplémentaire lors de la planification des trajectoires d'outil. En effet, pour de larges valeurs de cette dernière, le système proposée génère des trajectoires similaires à celles des méthodes classiques du point de vue géométrique tout en garantissant le respect du critère de la puissance maximale disponible et en simplifiant considérablement les calculs.

### 1.3.6.1 trajectoires de contournage

La première étape consiste à déterminer les différents plans de coupe en fonction du critère *TSF*. Les points de contacts outil/pièce relatifs à chaque plan de coupe seront déterminés par le calcul des points d'intersections plan/modèle et la construction des contours (Cf. Section 1.3.4 et 1.3.5). Les points de localisation d'outils seront ensuite calculés en fonction du type d'outils utilisé (figure 1.20) que nous détaillerons dans ce qui suit.

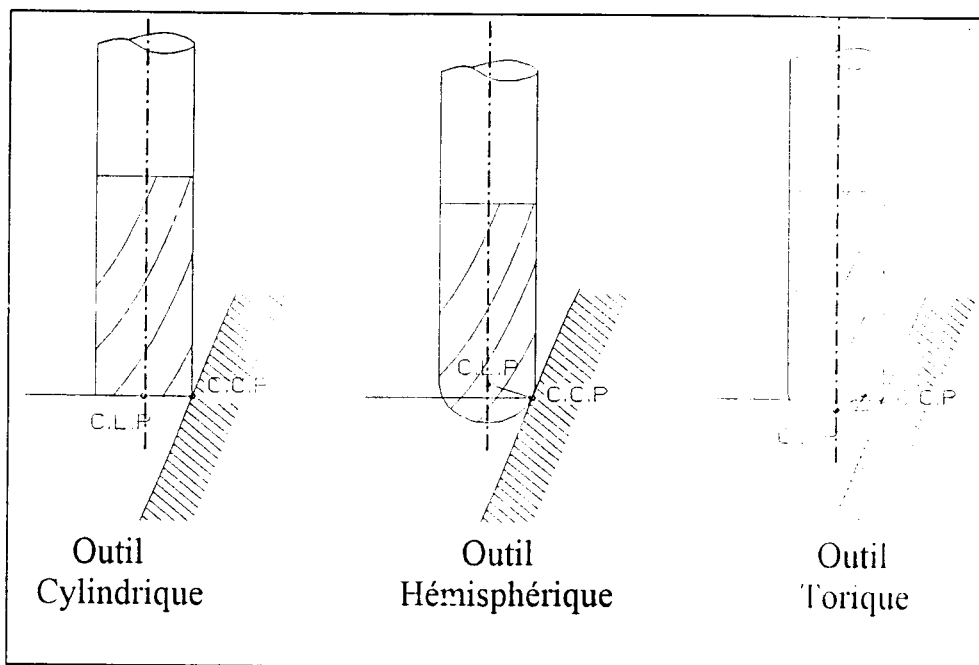


Figure 1.20 Calcul des points de contrôles de l'outil.

## A) Cas de l'outil cylindrique

### a) Calcul des points de localisation de l'outil

Ces points sont calculés en construisant les offset planes dans le plan de coupe courant des contours par un décalage de la valeur du rayon d'outil  $R$ , vers l'intérieur pour le contour extérieur et vers l'extérieur pour les contour des îlots (figure 1.21 et 1.22).

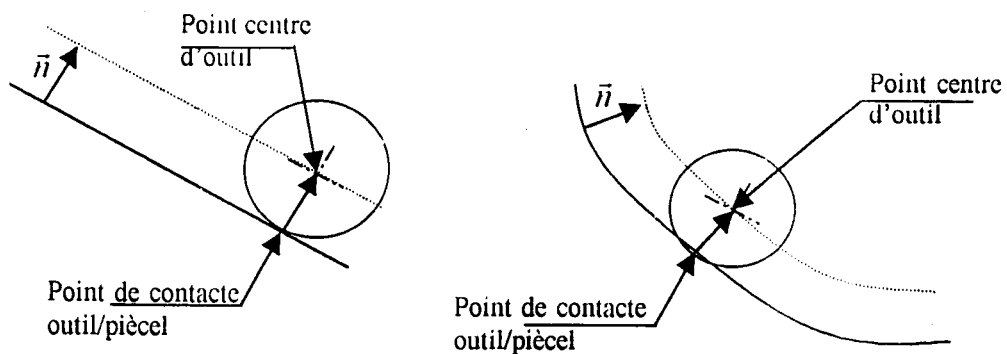


Figure 1.21 Passage du point de contact outil/pièce au point de localisation de l'outil dans le cas de l'outil cylindrique.

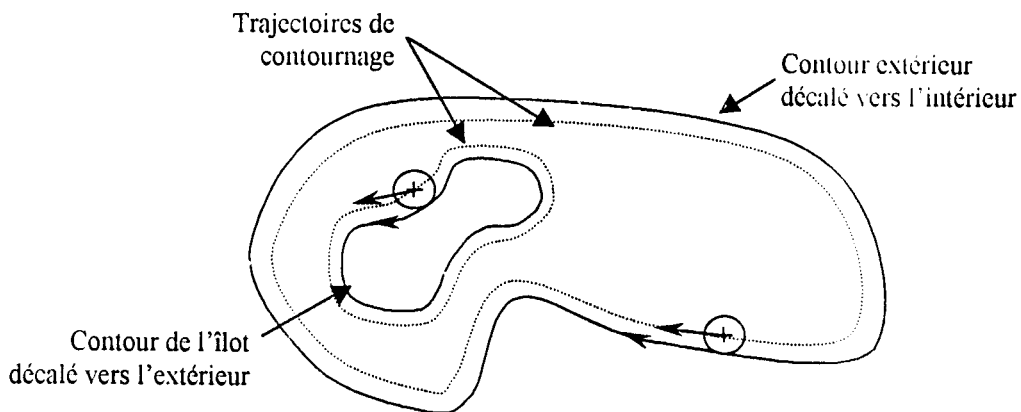


Figure 1.22 Trajectoires de contournage à l'outil cylindrique.

### b) Elimination des interférences éventuelles

La détermination des interférences éventuelles peut être réalisée à partir de l'analyse des singularités des contours décalés. Ces singularités peuvent être locales

et représentent dans ce cas des points de rebroussement (figure 1.23) ou des auto-intersections (figure II.1.24)

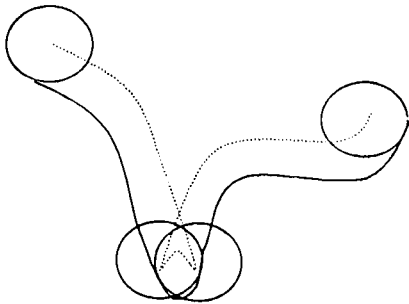


Figure 1.23  
Points de rebroussement sur la trajectoire d'outil donnant lieu à une interférence.

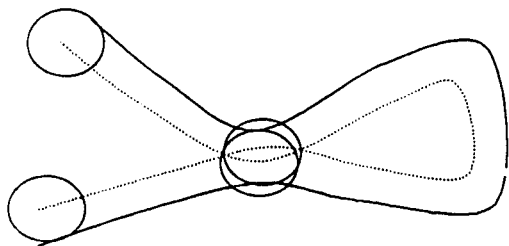


Figure 1.24  
Auto-intersection de la trajectoire d'outil donnant lieu à une interférence.

Comme elles peuvent être globales telles que des intersections inter-contour (figure 1.25), ou bien des discontinuités (figure 1.26).

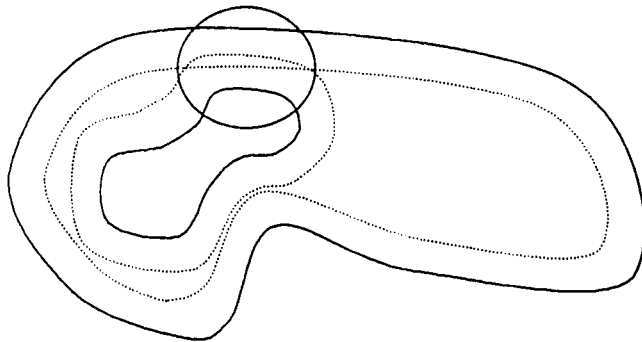


Figure 1.25  
Intersection inter- contour.

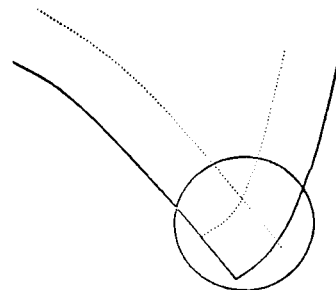
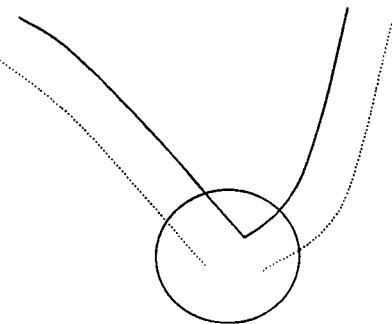


Figure 1.26 Discontinuités de la trajectoire d'outil.

Pour l'élimination des interférences et des singularités éventuelles, la méthode de Held sera utilisée [Held et al. '94], [Persson '78], [Akam et Limam '97], [Zirmi '98].

**c) Calcul du pas de plongé**

Le pas de plongée (Figure 1.27) est donné par la relation

$$P_{\max} = \frac{tsf}{\sin \alpha}$$

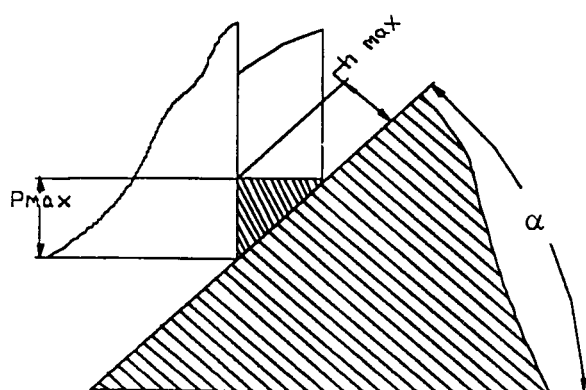


Figure 1.27  
 Calcul du pas de plongée  
 dans le cas de l'outil  
 cylindrique.

**B) Cas de l'outil hémisphérique**

**a) Calcul des points de localisation d'outils**

Ils sont calculés par la relation suivante (figure 1.28):

$$CLP_i = CCP_i + Rn$$

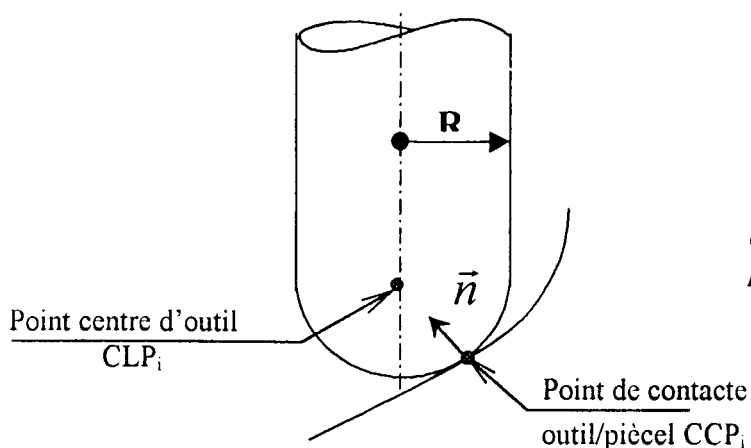


Figure 1.28  
 Passage d'un point de  
 contacte outil/piècel  
 au point de localisation de  
 l'outil correspondant.



## b) Elimination des interférences éventuelles

La technique d'élimination des interférences sera développée en détail lors de la description de la méthode d'usinage de finition (Cf. Chapitre 1.4).

## c) Calcul du pas de plongée

Le pas de plongée (Figure 1.29) est donné par la relation

$$P_{\max} = 2 \sqrt{2R \cdot tsf - tsf^2} \sin \alpha$$

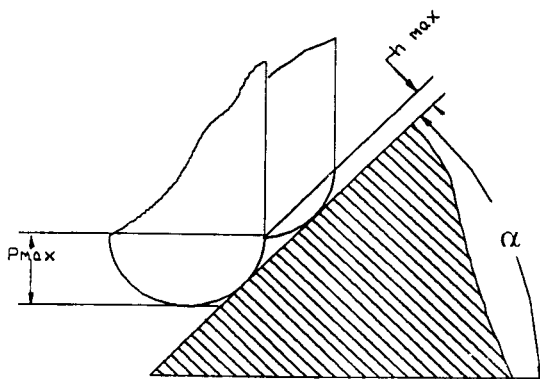


Figure 1.29  
 Calcul du pas de plongée  
 dans le cas de l'outil  
 hémisphérique.

## C- Cas de l'outil torique

### a) Calcul des points centre d'outils

Le calcul des points de contrôle de l'outil dont le détail est donné dans [Zhu'91] s'effectue de la façon suivante :

Pour un point de contact outil/pièce donné et son vecteur normal unitaire  $\vec{n}$ , un vecteur unitaire  $\vec{v}$  (figure 1.30) appartenant au plan formé par les vecteur  $\vec{n}$  et le vecteur axe d'outil  $\vec{T}_{ax}$  est calculé par l'expression

$$\vec{v} = \frac{\vec{n} - (\vec{n} \times \vec{T}_{ax}) \times \vec{T}_{ax}}{|\vec{n} - (\vec{n} \times \vec{T}_{ax}) \times \vec{T}_{ax}|}$$

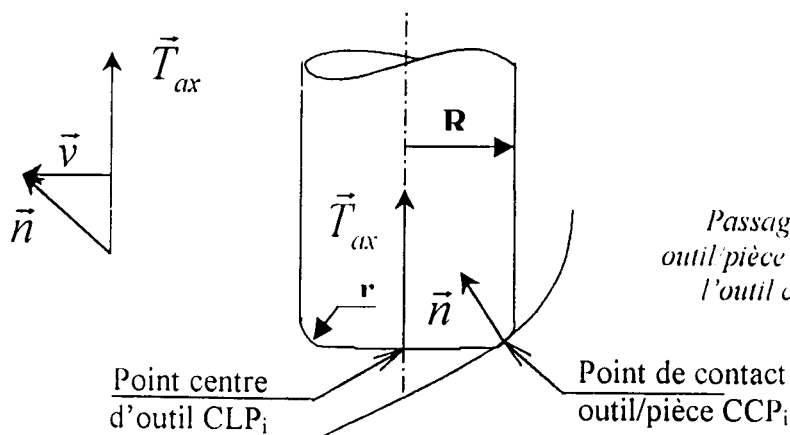


Figure 1.30  
 Passage d'un point de contact  
 outil/pièce au point de localisation de  
 l'outil correspondant de l'outil  
 torique

Le passage du point de contact  $CCP_i$  au point de localisation d'outil correspondant  $CLP_i$  sera donné par la relation

$$CLP_i = CCP_i + r\bar{n} + (R - r)\bar{v} - r\bar{T}_{ax}$$

### b) Elimination des interférences éventuelles

La technique d'élimination des interférences sera développée en détail lors de la description de la méthode d'usinage de finition (Cf. Chapitre 1.4).

### c) Calcul du pas de plongée

Des relations donnant l'expressions de la hauteur de crête (cf. Annexe 1), et en  $h$  par  $tsf$ , nous dérivons pour le :

1<sup>er</sup> cas : Intersection partie torique/partie torique.

$$P_{\max} = 2\sqrt{2r*tsf - tsf^2} \sin\alpha$$

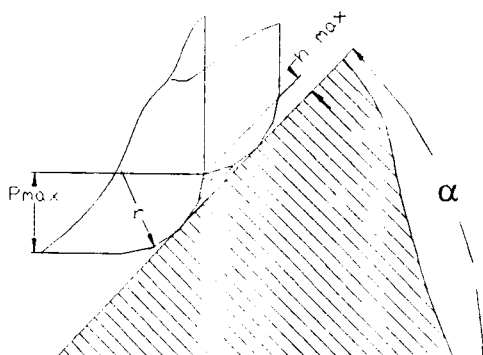


Figure 1.31  
 Intersection partie  
 torique/partie torique.

2<sup>em</sup> cas : Intersection partie torique/bout de fraise

$$P_{\max} = \left[ \sqrt{2r \cdot tsf - tsf^2} + \frac{tsf - r(1 - \cos \alpha)}{\operatorname{tg} \alpha} + r \sin \alpha \right] \sin \alpha$$

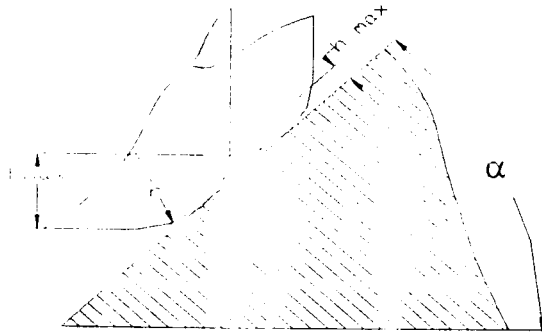


Figure 1.32  
 Intersection partie  
 torique bout de fraise

3<sup>em</sup> cas : Intersection génératrice/bout de fraise

$$P_{\max} = \left[ \sqrt{2rtsf - tsf^2} + \operatorname{tg}(tsf - r(1 - \sin \alpha)) + r \cos \alpha \right] \sin \alpha$$

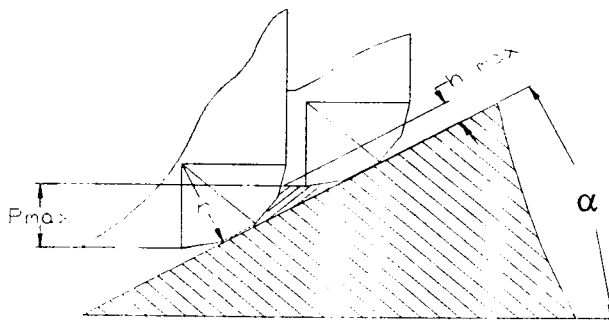


Figure 1.33  
 Intersection  
 génératrice/bout de fraise.

4<sup>em</sup> cas : Intersection génératrice/bout de fraise.

$$P_{\max} = r \sin \alpha + tsf - r(1 - \cos \alpha)$$

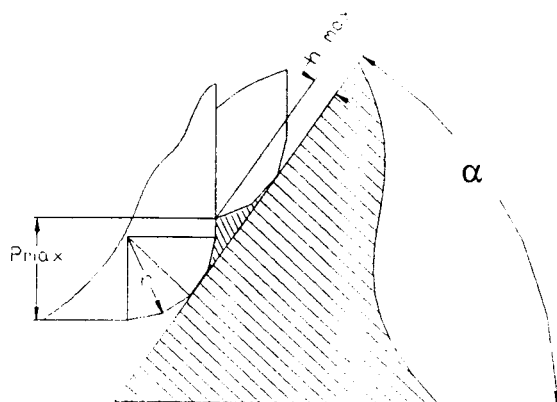


Figure 1.34  
 Intersection  
 génératrice bout  
 de fraise.

### 1.3.6.2 Polygonisation des contours

La méthode de génération de trajectoire d'outils développée permet de simplifier considérablement la géométrie des contours délimitant les zones de matière restante après contournage, ce qui implique ainsi une construction aisée des trajectoires d'évidement en zigzag parallèles (figure 1.35).

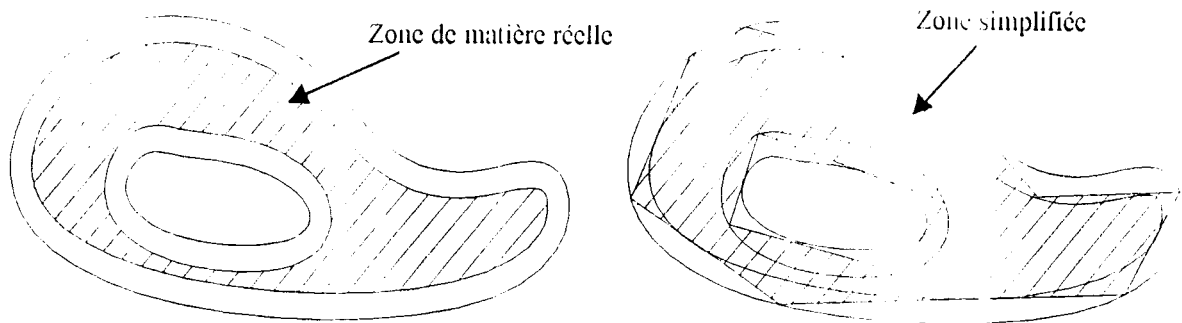


Figure 1.35 Polygonisation des contours.

### 1.3.6.3 Trajectoires d'évidements

Les trajectoires d'évidement sont construites à partir des contours polygonisés. La profondeur de passe est calculée, pour un rayon d'outil et un coefficient de recouvrement donné  $Rc$  (figure 1.36), en fonction de la puissance disponible sur la broche. L'opération étant un usinage combiné, la valeur du coefficient de recouvrement permettra d'apprécier la prédominance en roulant ou bien en bout.

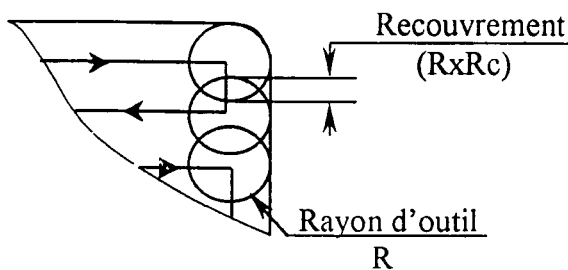


Figure 1.36  
Coefficient de recouvrement  
des trajectoires parallèles  
 $Rc$ .

### 1.3.6.4 Jumelage des trajectoires de contournage et d'évidement

Les trajectoires globales d'usinage sont construites par combinaison des trajectoires de contournage et d'évidement précédemment calculées. L'agencement

des passes de contournage par rapport aux passes d'évidement sera réalisé en fonction de la valeur de la cote  $z$  de chaque passe de contournage. Ainsi, pour une passe d'évidement définie par la cote  $Z_p$  du plan de coupe correspondant et la profondeur de passe  $a$ , toutes les trajectoires de contournage dont la cote  $Z_p'$  est comprise entre  $Z_p - a$  et  $Z_p$  lui seront associées.

## 1.4 Méthode d'usinage de finition proposée

Ne représentant pas l'objectif principal de ce travail, nous proposons néanmoins un module d'usinage de finition qui utilise les trajectoires par plans parallèles en raison des nombreuses propriétés qu'elles présentent (Cf. 1.2), qui repose en grande partie sur les algorithmes développés pour la phase d'ébauche, dans le but de compléter le processus de CFAO à même de nous assurer la conception et la réalisation des surfaces éprouvettes en vue de la validation envisagée.

### 1.4.1 Choix d'un type de trajectoires

Ce module permet de générer des trajectoires par balayage, en unidirectionnel ou en zigzag.

### 1.4.2 Calcul des points de contacts outil/pièce

Les points de contacts outil/pièce sont déterminés de façon similaire à l'usinage d'ébauche. L'algorithme de construction de contours (Cf. section 1.3.4) est toutefois adapté aux formes géométriques ouverts spécifiques à la construction des parcours d'outil de finition.

### 1.4.3 Calcul du pas transversal

Il s'effectue sous contrainte du respect de la hauteur de crête maximum. Les relations précédemment établies (Cf. section 1.3.6 et annexe D) permettent en effet de calculer le pas transversal en fonction de la valeur de la hauteur de crête maximum tolérée pour les trois types d'outils cylindrique, hémisphérique et torique. Pour un plan de coupe donné, la valeur du pas maximum sera calculée pour chaque point de contact *outil/pièce*, le pas transversal de décalage retenu sera celui dont la valeur est minimale.

### 1.4.4 Calcul des points de localisation de l'outil

L'usinage de finition est réalisé à l'outil hémisphérique ou à l'outil torique. Les relations développées pour le cas de l'usinage d'ébauche (Cf. Section 1.3.6.1) restent valables.

### 1.4.5 Gestion des interférences

#### 1.4.5.1 Considérations générales

La méthode de gestion des interférences élaborée repose sur les fondements utilisés dans la technique introduite par Hwang [Hwang'92]. Cette approche consiste en fait à calculer les points de contacts *outils/pièce* sans tenir compte, dans un premier temps, des risques d'interférences puis à éliminer les interférences, s'il y a lieu, lors du calcul des points de localisations d'outils. A cet effet, un point de localisation d'outil initial est calculé pour chaque point de contact *outil/pièce*, à l'aide des relations de passage précédemment développées. Une vérification aux interférences est ensuite effectuée en calculant la distance entre le point de localisation de l'outil et les facettes appartenant à la zone de contact *outil/pièce* (figure 1.37).

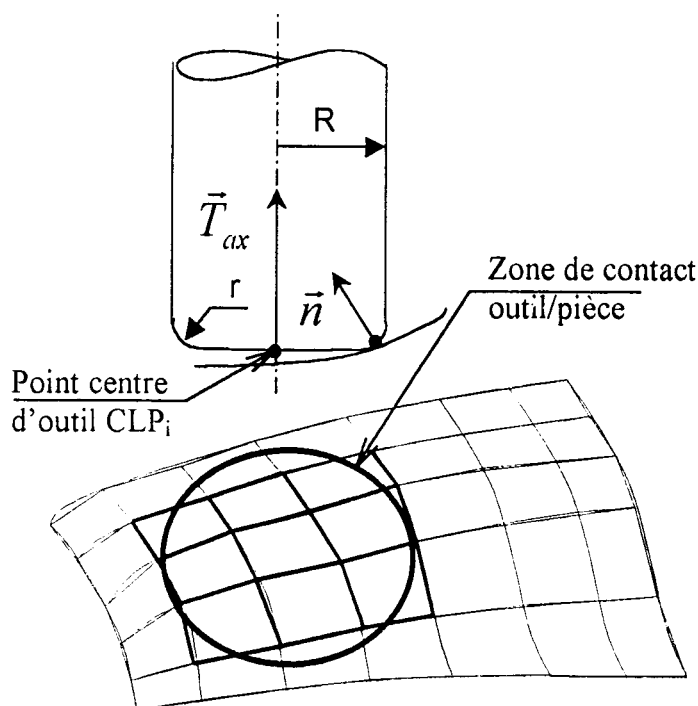


Figure 1.37  
Zone de contact outil/pièce.

Si une interférence est décelée, la position du point centre de l'outil est alors corrigée, le processus se répète ainsi pour tous les autres points de localisation. La méthode de Hwang, limité au cas de l'outil hémisphérique, corrige la position du point de localisation de l'outil en incrémentant la côte  $z$  d'une valeur calculée telle que l'outil soit tangent à la facette ayant généré l'interférence (figure 1.38).

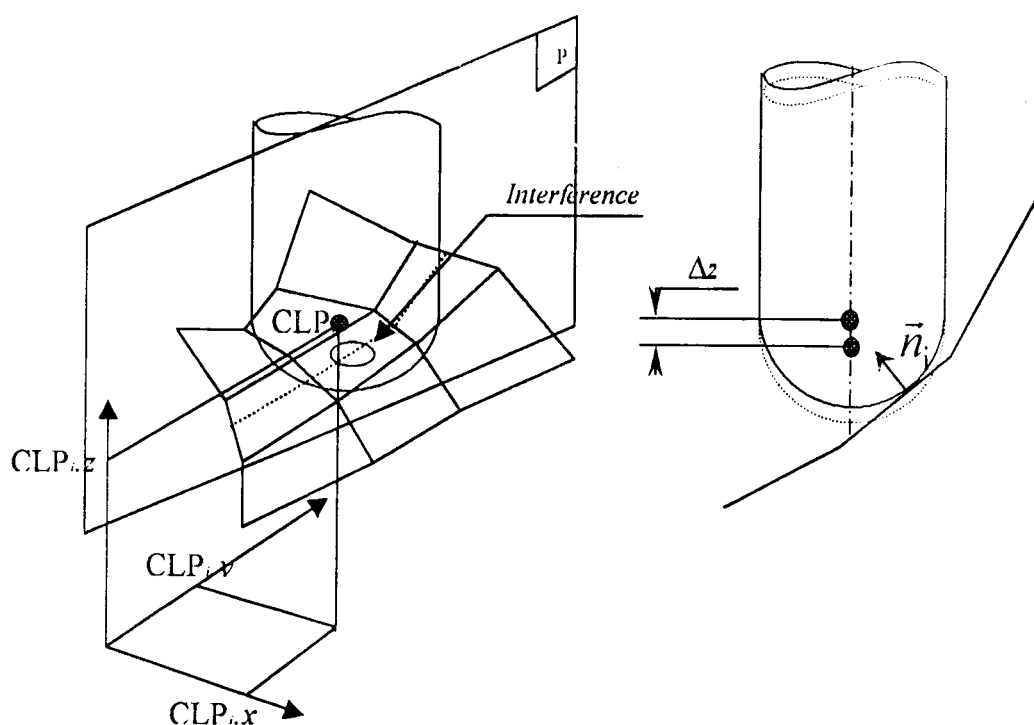


Figure 1.38 Elimination de l'interférence par incrémentation de la cote  $z$  de l'outil.

Nous remarquerons que cette correction a une répercussion néfaste sur les conditions d'enlèvement de copeau à l'outil hémisphérique, le point de contact étant décalé vers la zone de coupe nulle au voisinage du centre de l'outil. Ce phénomène est amplifié dans le cas de l'usinage en 3 axes de surfaces à faibles courbures dans les zones où la direction du vecteur normal se rapproche de celle de l'axe de l'outil, ce qui est courant en usinage de surfaces gauches (figure 1.39a). Pour remédier à ce problème, nous proposons comme alternative les fondements permettant une correction de la position du point de localisation de l'outil hémisphérique le long de la direction de la normale à la surface au point de contact considéré (figure 1.39b)

qu'il faudra valider par une réalisation expérimentale ultérieure.

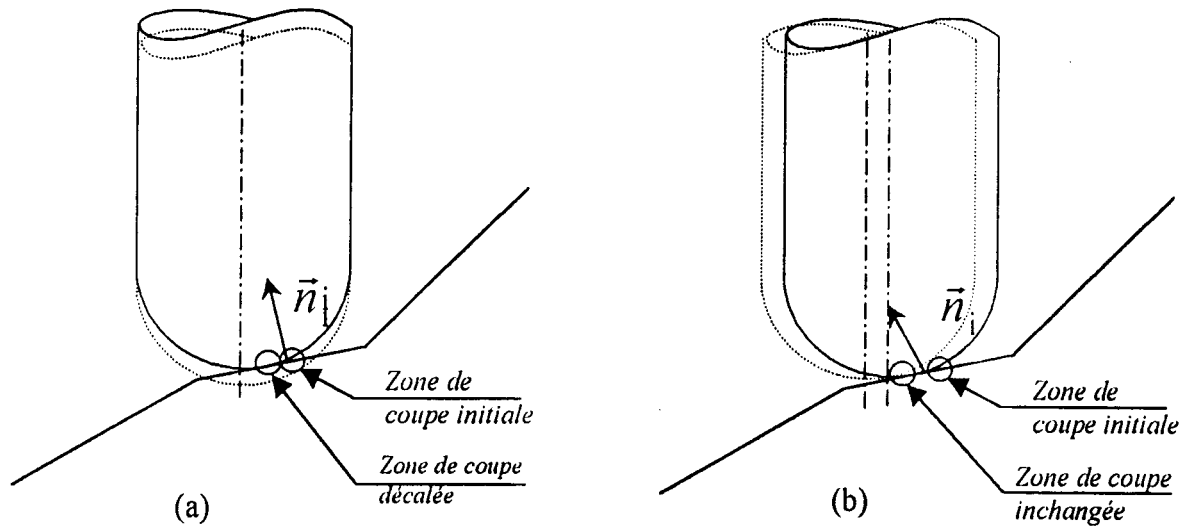


Figure 1.39 Répercussions de l'incrément de la cote z sur les conditions de formation de copeau. (a) correction de Hwang, (b) correction proposé.

Ainsi, nous développerons dans ce qui suit, les relations de détection d'interférence et de calcul du vecteur d'incrément de la position du point de localisation de l'outil hémisphérique permettant leur élimination.

Un changement de repère est effectué de sorte que l'axe z soit parallèle au vecteur normal à la surface au point de contact considéré, ce qui permettra une simplification considérable des calculs.

#### a) Identification de la zone de contact *outil/pièce*

La zone de contact *outil/pièce* pour un point de localisation de l'outil  $CLP_i$  donné, est définie par l'ensemble des points  $P_j$  de la surface telle que :

$$\left\{ |CLP_i - P_j|^2 - (T_{ax} \cdot (CLP_i - P_j))^2 < R^2 \right\} \quad [1.6]$$

avec  $T_{ax}$  : vecteur axe de l'outil.

La zone de contact *outil/pièce* sera définie par l'ensemble des facettes dont l'un des sommets vérifie la relation [1.6].



## b) Détection et élimination des interférences

Chaque facette initiale sera divisée en deux facettes triangulaires qui serviront ensuite pour les tests et les calculs. Dans un premier temps, la vérification est réalisée avec le plan infini contenant la facette; si une interférence est identifiée, il faudra alors vérifier si le lieu d'interférence appartient à la facette.

## c) Etablissement des relations de correction de la position de l'outil

Le calcul de la distance du  $CLP_i$  initial au plan infini permet la détection des interférences éventuelles. En effet, si cette distance est inférieure à la valeur du rayon d'outil  $R$ , alors une interférence aura lieu. Si le plan infini admet pour équation

$$ax + by + cz + d = 0,$$

dans le repère  $(O,X,Y,Z)$  avec  $n = (a, b, c)$  son vecteur normal unitaire, alors la distance du point de localisation de l'outil  $CLP_i$  au plan de la facette est donnée par

$$D = a * CLP_i.X + b * CLP_i.Y + c * CLP_i.Z + d.$$

Si la distance  $D$  est inférieure au rayon de l'outil  $R$ , alors la position du centre d'outil devra être corrigée tel que l'outil soit tangent à la facette. La direction du vecteur de correction étant donné par celle du vecteur normal, il suffit de déterminer alors la valeur du déplacement  $\Delta p$ .

Dans le repère local  $(o,x,y,z)$ , défini tel que l'axe  $z$  soit parallèle au vecteur normal  $n$ , et centré au point  $o$ , projection orthogonale du point de localisation de l'outil  $CLP_i$  sur la facette, le vecteur normal  $n$  aura pour composantes  $(0,0,1)$  et l'équation du plan s'écrira

$$z = 0$$

la distance du point de localisation d'outil  $CLP_i$  au plan de la facette sera donnée par

$$D' = CLP_i.z$$

De même, si la distance  $D'$  est inférieure au rayon de l'outil  $R$  alors l'outil interfère avec la facette. Dans le repère local, la correction de la position de l'outil revient à calculer uniquement la cote  $z'$  du point de localisation de l'outil telle que la distance  $D'$  soit égale au rayon d'outil  $R$ .

$$D' = R \Rightarrow CLP_i.z' = R$$

avec  $\Delta p = CLP_i.z' - CLP_i.z$

Connaissant la nouvelle position du point de localisation de l'outil  $CLP_i(0,0,CLP_i.z)$  dans le repère local, on peut déterminer ses nouvelles composantes dans le repère global par la matrice de passage associée.

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c & \Delta x \\ -b & 1 - \frac{b^2}{1+a} & -\frac{bc}{1+a} & \Delta y \\ -c & -\frac{bc}{1+a} & 1 - \frac{c^2}{1+a} & \Delta z \\ 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

soit

$$\begin{cases} X = ax + by + cz + \Delta x \\ Y = -bx + \left(1 - \frac{b^2}{1+a}\right)y - \left(\frac{bc}{1+a}\right)z + \Delta y \\ Z = -cx - \left(\frac{bc}{1+a}\right)y + \left(1 - \frac{c^2}{1+a}\right)z + \Delta z \end{cases}$$

comme

$$\begin{cases} CLP_i.x = CLP_i.y = 0 \\ CLP_i.z = R \end{cases}$$

donc le point de localisation de l'outil  $CLP_i$  sera donné par

$$\begin{cases} CLP_i.X = c * R + \Delta x \\ CLP_i.Y = \left(\frac{bc}{1+a}\right)R + \Delta y \\ CLP_i.Z = \left(1 - \frac{c^2}{1+a}\right)R + \Delta z \end{cases}$$

### 1.4.6 Génération des informations pour une reprise locale d'usinage

Afin de garantir la conformité de la pièce usinée avec le modèle de départ, une reprise d'usinage locale s'impose pour l'enlèvement des surplus de matière laissés après élimination des interférences (figure 1.40). Les informations nécessaires à cette opération doivent être mémorisées. A cet effet, il convient d'identifier d'abord toutes les facettes appartenant aux zones considérées. Ensuite, pour chaque facette, les dimensions de l'outil permettant une reprise d'usinage totale sans interférence, doivent être calculés. La première information est obtenue par une simple mémorisation de toutes les facettes ayant nécessité un traitement d'interférence. Le calcul des valeurs successives de  $\Delta p$  pour chaque facette nous permettra de calculer la valeur du rayon d'outil maximum compatible avec la géométrie locale de la surface. En effet, pour toutes les facettes d'une région d'interférence donnée, si nous prenons un rayon d'outil égal au rayon initial  $R$ , diminué de la valeur maximale de tous les incréments calculés avec l'ensemble des facettes de la zone considérée, alors aucune interférence n'interviendrait dans cette zone.

$$R_{max} = R - \max(\Delta p)$$

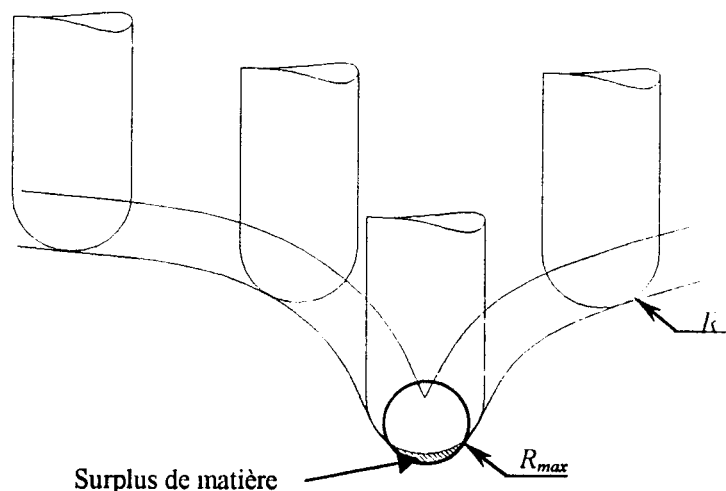


Figure 1.40 Passe de finition ( $R > R_{max}$ ) nécessitant une reprise locale d'usinage

Les trajectoires de reprise seront ensuite calculées pour le balayage des zones d'interférences précédemment identifiées, comme dans le cas de l'usinage initial de

la surface. Le rayon d'outil choisi étant bien entendu inférieur ou égale à  $R_{max}$  (figure 1.41).

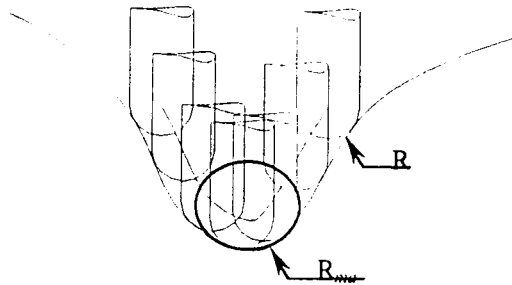


Figure 1.41  
Reprise locale d'usinage ( $R \leq R_{max}$ )

## 1.5 Conclusion

Le système élaboré permet de combler certaines lacunes au niveau des caractéristiques fonctionnelles des systèmes existant (Cf. sections 2.4.1.2 et 2.5.2). En effet<sup>2</sup>, il permet l'usinage de surfaces gauches composites sans restrictions de modèles ni de degrés. Sur le plan des tolérances, une gestion rigoureuse des erreurs de cordes et de hauteur de crête est mise en œuvre. La nouvelle méthode de planification des trajectoires d'outil a permis une optimisation des trajectoires d'usinage d'ébauche ainsi qu'une utilisation optimale de la puissance machine disponible. En finition, la maîtrise des tolérances et la gestion efficace des interférences garantit la conformité de la pièce usinée.

## Chapitre II.2 Mise en œuvre informatique et validation

### 2.1 Introduction

L'évolution technologique que nous vivons, et qui ne cesse de prendre de l'ampleur, a conduit à une tendance de spécialisation de plus en plus pointue. Afin de maintenir des degrés d'efficacité acceptable, les systèmes de CFAO ont du s'intégrer à cette tendance, en offrant des ouvertures qui permettent l'adjonction de modules métiers, développés par des spécialistes.

Le schéma général d'intégration de nouvelles fonctionnalités et d'applications à un logiciel de CFAO, adaptées à un métier ou à une tâche spécifique, peut prendre trois formes distinctes (*Figure 2.1, 2.2 et 2.3*). La première configuration consiste à générer dans un fichier les données qui seront traitées par une application externe autonome, qui, à son tour renverra les résultats dans un autre fichier (*Figure 2.1*). Cette configuration, de par son manque de convivialité, n'est adaptée qu'aux applications de calculs importants ne nécessitant pas d'intervention de la part de l'utilisateur.

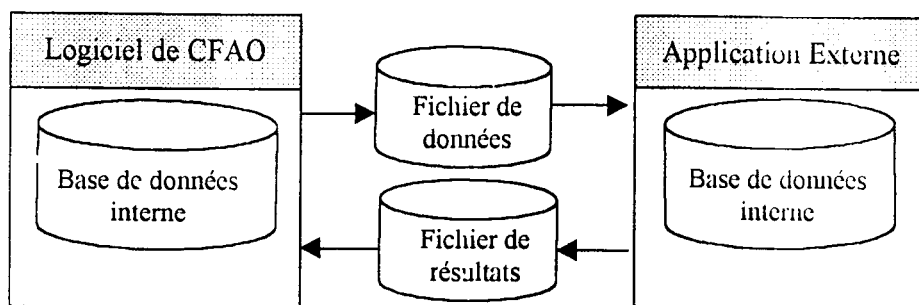


figure 2.1 Application externe autonome

La deuxième configuration consiste à remplacer les fichiers d'échange par un accès partagé aux bases de données internes du logiciel (*Figure 2.2*), l'application étant toujours autonome. Dans ce cas, le degré de convivialité dépend de la qualité de l'application puisqu'il lui incombe de gérer le dialogue pendant son exécution avec

l'utilisateur et de restituer l'environnement d'exécution avant de rendre la main au logiciel.

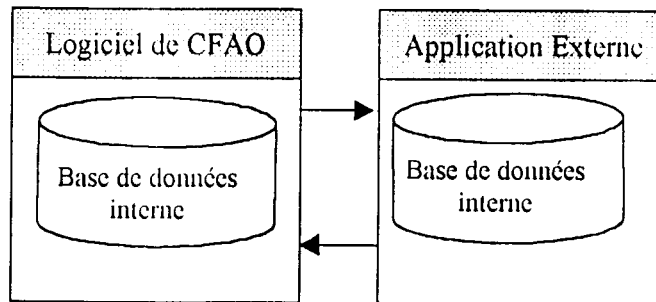


Figure 2.2 Application externe autonome avec accès partagé à la base de données.

La troisième configuration (Figure 2.3), née des évolutions importantes du génie logiciel, permet de développer des applications totalement intégrées au logiciel et invoquée le moment opportun par un chargement dynamique des codes exécutables. L'environnement d'exécution étant inchangé, l'opération est totalement transparente pour l'utilisateur qui dispose ainsi de la même interface et du même contexte de travail. Cet avantage nous a amené à intégrer à un logiciel industriel les modules d'usinage développés sous forme d'une application de ce type.

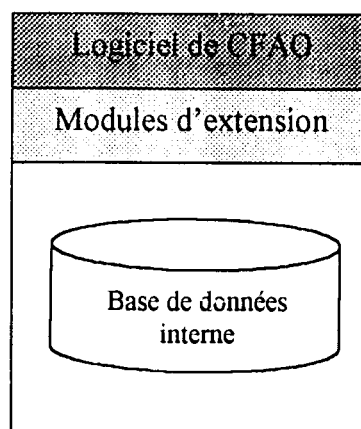


Figure 2.3 Application totalement intégrée au logiciel.

## 2.2 Environnement de développement utilisé

Pour le développement de l'application, notre choix a porté sur le logiciel de CAO AutoCAD-R14 [Autocad'94a] en raison de son ouverture et des larges possibilités d'intégration qu'il offre. En effet, il existe sous AutoCAD trois possibilités d'intégration de nouvelles fonctionnalités. L'utilisateur dispose d'AUTOLISP [Autocad'94b], un langage propre à AutoCAD qui est dérivé du langage de programmation interprété « Lisp » et du langage scientifique évolué "C" dans sa variante classique ou orientée objet, offrant l'opportunité d'invoquer des bibliothèques de fonctions riches et très largement documentées, qui se présentent sous forme de systèmes de développement ADS "AutoCAD Développement Système " et ARX "AutoCAD Runtime eXtension". Les bibliothèques ARX qui se caractérisent par leurs performances au niveau des temps d'exécution et leur compatibilité avec le langage C++ Microsoft, nous ont permis d'intégrer notre application en bénéficiant de la puissance, de la richesse et de la convivialité de l'environnement de développement Microsoft Visual C++ Ver4 (Figure 2.7). Les détails d'intégration sont donnés dans l'annexe III.

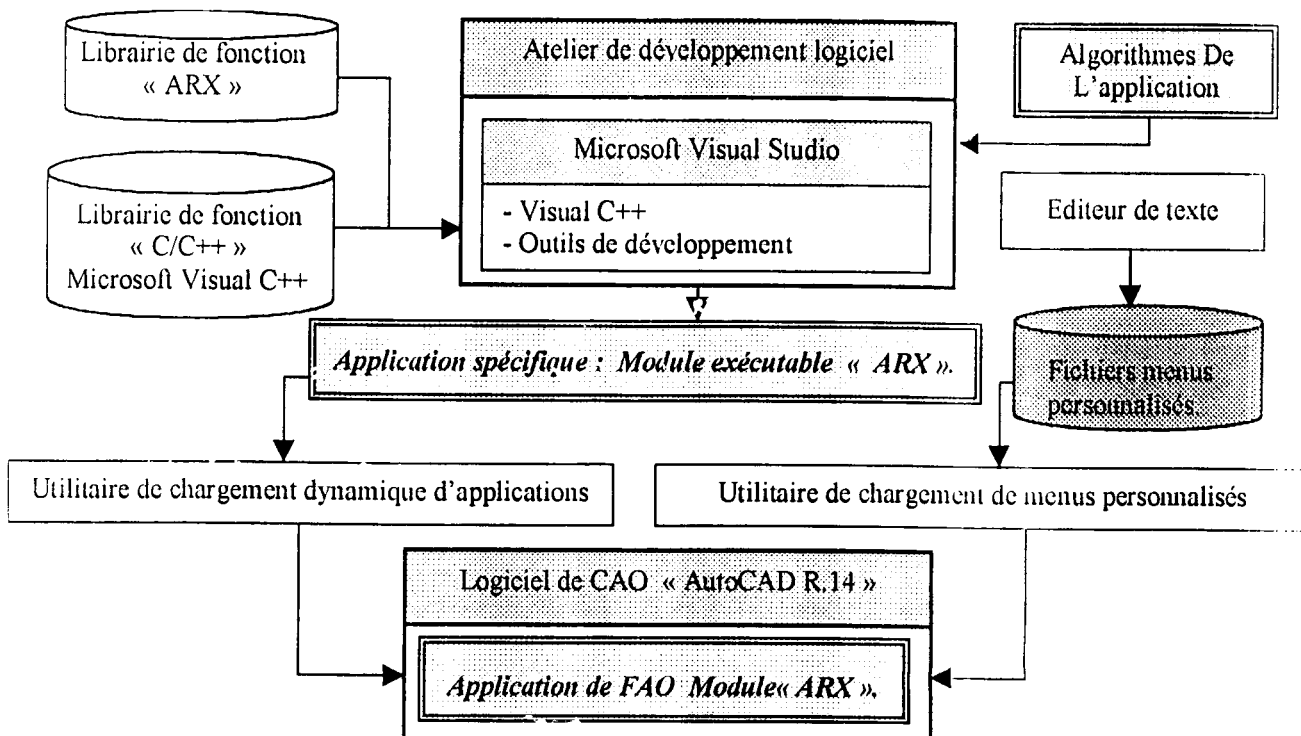


Figure 2.3 Environnement de développement.

## 2.3 Module de Conversion/Discretisation

Les surfaces représentant le modèle CAO de la pièce à usiner sont empilées dans une structure dynamique de stockage temporaire, puis les algorithmes de conversion /discretisation dont les procédures sont détaillées ci-dessous, sont appliqués de façon récursive. L'organigramme général du module est donné en annexe II.

### 2.3.1 Algorithmes et Implémentation

#### A- Conversion par approximation d'une surface de Bézier rationnelle

Les carreaux de surfaces de type Béziérs dont le degré est élevé sont convertis en carreaux de Béziérs de degrés faibles grâce à l'algorithme développé dans [Asma '98].

#### B- Procédure d'insertion de nœuds pour une courbe B-Spline

L'algorithme utilisé est développé par Boehm [Boehm '80], [Boehm et Prautzsch '85] et permet, par insertion répétée de nœuds, de convertir une courbe B-Spline en courbes de Béziérs équivalentes.

##### 1-entrés:

- Courbe B-Spline  $C(u)$ , définie par :
  - \* son polygone caractéristique  $P_n$ ,
  - \* son degré  $m$ ,
  - \* son vecteur nodal  $u_i, i \in \{0, 1, \dots, n+m+1\}$
  - \* valeur  $t$  à insérer

##### 2-Algorithmes :

- 1- **DEBUT**
- 2- Rechercher l'intervalle nodal d'insertion
  - valeur de  $p$  tel que ( $u_p \leq t \leq u_{p+1}$ )
- 3- insérer le nouveau nœud
  - 3.1-  $i = p + m$
  - 3.2-  $k = p + 1 + m$
  - 3.3-  $P_{k-m} = P_{i-m}$
  - 3.4- **Si**  $t \leq u_i$  et  $i > p$  **Alors**
    - $u_k = u_i$
    - $i = i - 1$
    - $k = k - 1$
  - Fin Si**
  - 3.5- **Pour**  $l = 1$  à  $n$  **Faire**
    - $q = u_{k+l} - t$
    - **Si**  $q \leq 0$  **Alors**
      - $p_{k-n+l-1} = p_{k-m+1}$
    - Si Non**
      - $q = q / (u_{k+l} - u_{i-m+1})$
      - $p_{k-m+l-1} = q * p_{k-m+l-1} + (1-q) * p_{k-m+1}$
  - Fin Si**
  - Fin Faire**



$uk=t$

$n=n+1$

5- FIN

### 3-Sorties:

- Nouvelle courbe B-Spline  $C'(u)$ , définie par :

\* son polygone caractéristique  $P_n$ ,

\* son degré  $m$ ,

\* son vecteur nodal  $u_i, i \in \{0, 1, \dots, n'+m+1\}$

### C- Conversion d'une surface B-Spline rationnelle en carreaux de Bézier rationnels

Les carreaux de surfaces de type B-Spline sont convertis en carreaux de Béziérs en appliquant l'algorithme d'insertion de nœuds de Boehm ci-dessus [Wong'90], [Ben Salah'90].

### D- Procédure de subdivision d'une courbe de Bézier rationnelle

La procédure élaborée pour la subdivision d'une courbe de Bézier rationnelle est la suivante :

#### 1-entrés:

- Courbe de Bézier Rationnelle  $C(u)$ , définie par :

\* son polygone caractéristique  $P_n$

\* affecté des poids  $w_n$

#### 2-Algorithmes :

1- DEBUT

2-Lire le polygone caractéristique  $P_n$  de la courbe  $C(u)$  avec les poids  $w_n$

3- Lire le paramètre de subdivision  $u^*$

4- Calculer les polygones caractéristiques des nouvelles courbes avec leurs poids

Pour  $i=0$  à  $n$  Faire

$$S_i^0 = P_i$$

$$\varphi_i^0 = w_i$$

Fin de Faire

Pour  $j=0$  à  $n$  Faire

Pour  $i=0$  à  $(n-j)$  Faire

$$S_i^j(u^*) = (1-u^*) \frac{w_i^{j-1}(u^*)}{w_i^j(u^*)} S_i^{j-1}(u^*) + u^* \frac{w_{i+1}^{j-1}(u^*)}{w_i^j(u^*)} S_{i+1}^{j-1}(u^*)$$

$$\varphi_i^j(u^*) = (1-u^*) w_i^{j-1}(u^*) + u^* w_{i+1}^{j-1}(u^*)$$

Fin de Faire // boucle sur  $i$

Fin de Faire // boucle sur  $j$

Pour  $i=0$  à  $n$  Faire

$$R_i = S_0^i(u^*) \quad \text{et} \quad \alpha_i = \varphi_0^i(u^*)$$

$$Q_i = S_i^{n-i}(u^*) \quad \text{et} \quad \beta_i = \varphi_i^{n-i}(u^*)$$

*Fin de Faire*

5- FIN

3-Sorties:

- 2 courbes Bézier  $C1(u)$ ,  $C2(u)$  définies par :
  - \* leurs polygones caractéristiques  $R_n$ ,  $Q_n$
  - \* affecté respectivement des poids  $\alpha_n$ , et  $\beta_n$ ,

**E- Procédure de subdivision d'un carreau de Bézier rationnel**

Pour la subdivision d'un carreau de Bézier  $S(u,v)$ , définie par  $(P_{m,n}, w_{m,n})$  aux paramètres  $(u^*, v^*)$ , on appliquera l'algorithme de subdivision des courbes de Bézier successivement aux  $(m+1)$  polygones caractéristiques dans la direction  $u$ , au paramètre  $u^*$ , puis aux  $2*(n+1)$  polygones caractéristiques ainsi obtenus, dans la direction  $v$ , au paramètre  $v^*$ . L'algorithme élaboré à cet effet est mis en œuvre de la façon suivante :

1-entrés:

- Carreau de Bézier  $S(u,v)$ , défini par :
  - \* son réseau caractéristique  $R_{m,n}$
  - \* affecté des poids  $w_{m,n}$

2-Algorithmes :

1- DEBUT

2- Lire le réseau caractéristique  $R_{m,n}$  de la surface  $S(u,v)$  avec les poids  $w_{m,n}$

3- Lire les paramètres de subdivision  $u^*$  et  $v^*$

4- Pour les  $(m+1)$  polygones caractéristiques  $(P_{ni})$  de poids  $(v_{ni})$  dans la direction  $u$  **Faire**

**Pour  $i=0$  à  $m$  Faire**

-Appliquer l'algorithme de subdivision

$((P_{ni}), (v_{ni}), u^*)$  en  $((R_{ni}), (\alpha_{ni}))$  et  $((Q_{ni}), (\beta_{ni}))$

**Fin de Faire**

**Fin de Faire**

5- Déterminer les deux nouveaux carreaux  $S1(R1_{m,n}, w1_{m,n})$

**Pour  $j=0$  à  $m$  Faire**

**Pour  $k=0$  à  $n$  Faire**

-  $(R1_{j,k}) = (R_{nj})_k$        $(w1_{j,k}) = (\alpha_{ni})_k$

-  $(R2_{j,k}) = (Q_{ni})_k$        $(w2_{j,k}) = (\beta_{ni})_k$

**Fin de Faire**

**Fin de Faire**

6- Subdiviser le nouveau carreau  $S1(R1_{m,n}, w1_{m,n})$

6.1- Pour les  $(m+1)$  polygones caractéristiques  $(P_{ni})$  de poids  $(v_{ni})$ , dans la direction  $v$  **Faire**

**Pour  $i=0$  à  $m$  Faire**

-Appliquer l'algorithme de subdivision

$((P_{ni}), (v_{ni}), u^*)$  en  $((R_{ni}), (\alpha_{ni}))$  et  $((Q_{ni}), (\beta_{ni}))$

**Fin de Faire**

**Fin de Faire**

6.2- Déterminer les deux nouveaux carreaux  $S3(R3_{m,n}, w3_{m,n})$  et  $S4(R4_{m,n}, w4_{m,n})$

Pour  $j=0$  à  $m$  Faire

Pour  $k=0$  à  $n$  Faire

$$- (R3_{j,k}) = (R_{n\setminus k}); \quad (w3_{j,k}) = (\alpha_{n\setminus k})$$

$$- (R4_{j,k}) = (Q_{n\setminus k}); \quad (w4_{j,k}) = (\beta_{n\setminus k})$$

Fin de Faire

Fin de Faire

7- Subdiviser le nouveau carreau  $S2(R2_{m,n}, w2_{m,n})$

7.1-Pour les  $(m+1)$  polygones caractéristiques  $(P_n)_i$ , de poids  $(v_n)_i$  dans la direction  $v$  Faire

Pour  $i=0$  à  $m$  Faire

-Appliquer l'algorithme de subdivision

$$((P_n)_i, (v_n)_i, v^*) \text{ en } ((R_n)_i, (\alpha_n)_i) \text{ et } ((Q_n)_i, (\beta_n)_i)$$

Fin de Faire

Fin de Faire

7.2- Déterminer les deux nouveaux carreaux  $S1(R3_{m,n}, w3_{m,n})$  et  $S2(R4_{m,n}, w4_{m,n})$

Pour  $j=0$  à  $m$  Faire

Pour  $k=0$  à  $n$  Faire

$$- (R3_{j,k}) = (R_{n\setminus k}); \quad (w3_{j,k}) = (\alpha_{n\setminus k})$$

$$- (R4_{j,k}) = (Q_{n\setminus k}); \quad (w4_{j,k}) = (\beta_{n\setminus k})$$

Fin de Faire

Fin de Faire

8- FIN

### 3-Sorties:

- 4 Carreaux de Bézier  $S1(u,v)$ ,  $S2(u,v)$ ,  $S3(u,v)$ ,  $S4(u,v)$ , définis par :

\* leurs réseaux caractéristique  $R1_{m,n}$ ,  $R2_{m,n}$ ,  $R3_{m,n}$ ,  $R4_{m,n}$ ,

\* affectés respectivement des poids  $w1_{m,n}$ ,  $w2_{m,n}$ ,  $w3_{m,n}$ ,  $w4_{m,n}$ .

## F- Procédure de discrétisation des carreaux de Bézier

La procédure élaborée pour la discrétisation, utilisant la subdivision récursive des carreaux, jusqu'à satisfaction totale des tolérances imposées par l'utilisateur est la suivante :

### 1-Entrées:

- liste chaînée des carreaux de Bézier constituant le modèle :

### 2-Algorithme :

1- DEBUT

2-Lire le réseau caractéristique d'un carreau

3-Calculer l'erreur de linéarité des bords et de planéité

3.1- Si la tolérance n'est pas vérifiée Alors

- subdiviser le carreau en quatre

- Remplacer le carreau initial par les quatre sous carreaux

Fin Si

Si Non // la tolérance est vérifiée

- Mémoriser le carreau comme facette

Fin Si

4- Si autre carreau à traiter Alors

- Aller a 2

*Fin Si // pas d'autre carreau à traiter*  
5- Lire le réseau caractéristique d'un carreau/Facette  
6- Calculer le vecteur normal au centre  $N(0.5,0.5)$   
7- Déterminer le repère local tel que  $(\bar{oz} // N_{0.5,0.5})$   
8- Calculer  $H = z_{\max} - z_{\min}$   
8.1- **Si**  $H > Tc$  **Alors**  
- subdiviser le carreau en quatre  
- Remplacer le carreau initial par les quatre sous carreaux  
**Fin Si**  
**Si Non** // la tolérance est vérifiée  
- Mémoriser le carreau comme facette finale  
**Fin Si**  
9- **Si** autre carreau/facette à traiter **Alors**  
- Aller a 5  
**Fin Si** // pas d'autre carreau/facette à traiter  
10- **FIN**  
**3-Sorties:**  
- Ensembles des carreaux/facettes.

### 2.3.2 structure de données

Les performances d'un algorithme sont en grande partie conditionnées par les structures de données qu'il implémente, surtout lorsque le volume de données à traiter est important. De plus, une utilisation rationnelle des ressources mémoire s'impose ; on est donc amené à l'utilisation des structures de données dynamiques.

L'algorithme proposé repose sur trois types de structures dynamiques : les piles, les listes chaînées et les arbres.

La pile servira au stockage temporaire des informations relatives aux carreaux à subdiviser ; l'arbre, de type quaternaire (*Figure 2.6*), sera utilisé pour mémoriser les informations des facettes (*Figure 2.5*). La liste chaînée sera couplée à l'arbre quaternaire (*Figure 2.7*) afin d'en simplifier le parcours, elle mémorisera les adresses des nœuds terminaux de l'arbre.

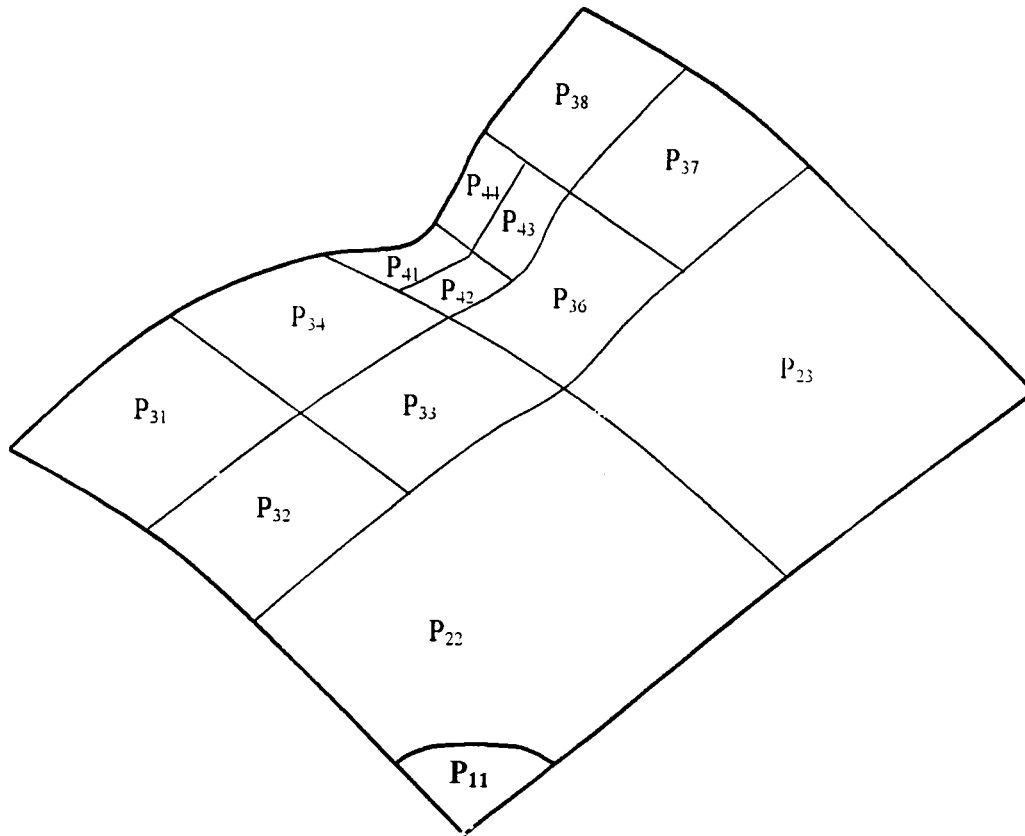


Figure 2.5 Exemple de simulation du processus de subdivision.

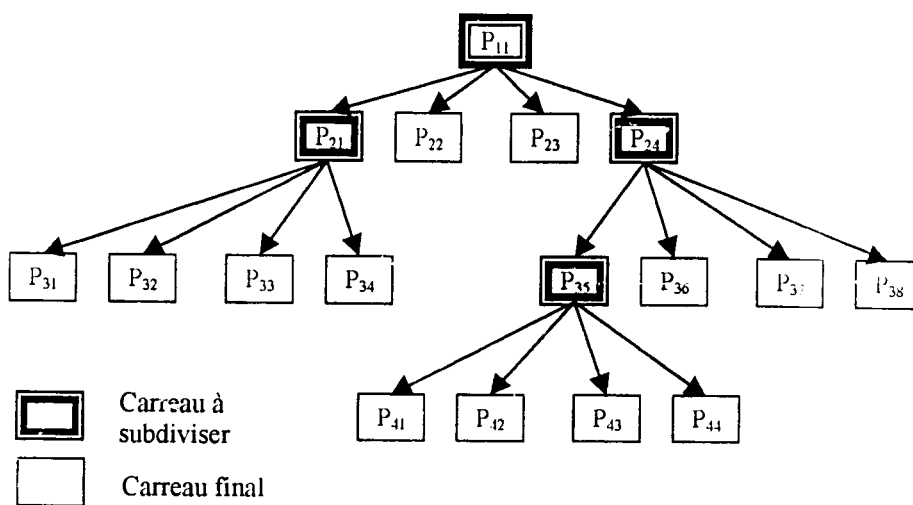


Figure 2.6 Arbre quaternaire issu du processus de subdivision.

Phase	Pile	Liste
PHASE 1	P <sub>11</sub>	Vide
PHASE 2	P <sub>21</sub> P <sub>24</sub>	P <sub>22</sub> P <sub>23</sub>
PHASE 3	P <sub>35</sub>	P <sub>31</sub> P <sub>32</sub> P <sub>33</sub> P <sub>34</sub> P <sub>36</sub> P <sub>37</sub> P <sub>38</sub>
PHASE 4	Vide	P <sub>31</sub> P <sub>32</sub> P <sub>33</sub> P <sub>34</sub> P <sub>36</sub> P <sub>37</sub> P <sub>38</sub> P <sub>41</sub> P <sub>42</sub> P <sub>43</sub> P <sub>44</sub>

Figure 2.7 Etat de la pile et de la liste au cours du processus de subdivision de la Figure 2.5.

### 2.3.3 Exemples d'exécution

La Figure 2.8 montre un carreau de Bézier de degré (6\*6), modélisé à l'aide du logiciel AutoCAD V14.

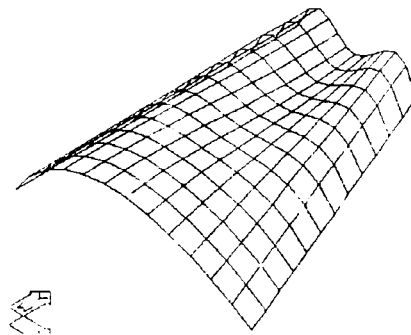


Figure 2.8 Surface de Bézier de degré (6x6).

Le module de conversion et discrétisation nous permet de réaliser une discrétisation auto-adaptative du carreau en garantissant le respect des tolérances utilisateur. La Figure 2.9a montre le résultat obtenu avec une tolérance imposée de  $0.2mm$ , alors que pour la Figure 2.9b cette tolérance a été réduite à  $0.02mm$ .



Figure 2.9 Facettisation de la surface avec une tolérance de  $0.2mm$ (a) et  $0.02mm$  (b).

## 2.4 Module d'usinage d'Ebauche

### 2.4.1 Organigramme général

Les algorithmes de calculs des différentes parties du module d'ébauche dont l'organigramme général du module est donné en annexe II, sont décrits en détails dans ce qui suit.

### 2.4.2 Algorithmes et Implémentation

#### A- Procédure de calcul des points d'intersection plan/facettes

Le calcul des points d'intersections plan/facette est réalisé à l'aide de l'algorithme suivant (Cf. annexe II).

##### 1-Entrées:

- Liste chaînée des facettes
- Plan de coupe  $[ N(a,b,c) \text{ et } p(p_x,p_y,p_z) ]$

##### 2-Algorithmes :

1- DEBUT

- 2- Lire les données du plan de coupe [  $N(a,b,c)$  et  $p(p_x,p_y,p_z)$  ]
- 3- Si plan de coupe non parallèle a (xy) Alors
  - Calculer la matrice de passage (O,X,Y,Z) -> (o,x,y,z)
- 4- Lire une facette
  - Calculer les composantes dans le nouveau repère (uniquement les z)
- 5- pour les quatre cotés de la facette Faire
- 6- Vérifier la possibilité d'intersection
- 7- Si intersection possible Alors
  - Si coté contenu dans le plan Alors
    - Prendre le point milieu comme point d'intersection
  - Si Non
    - Si extrémité appartient au plan Alors
      - Prendre le point extrémité comme point d'intersection
    - Fin Si
  - Fin Si
- Si Non
  - Calculer la valeur approchée du paramètre t.
- Fin Si
- 8- en fonction du coté considéré Faire
  - Si coté 1 Alors
    - $U^*=t ; v^*=0 .$
    - Fin Si
  - Si coté 2 Alors
    - $U^*=1 ; v^*=t .$
    - Fin Si
  - Si coté 3 Alors
    - $U^*=t ; v^*=1 .$
    - Fin Si
  - Si coté 4 Alors
    - $U^*=0 ; v^*=t .$
    - Fin Si
- Fin Faire
- 9- Calculer le point d'intersection exacte
  - Procédure Newton-Raphson ( $u^*,v^*$ )
- 9- Si autres facettes Alors
  - aller à 4
- 10- FIN

### 3- Sorties:

- Liste chaînée des segments d'intersections.

## **B- Procédure de calcul des points d'intersections exacts**

Le passage des points d'intersection approchés au point correspondant sur la surface est réalisé grâce à la procédure ci dessous.

### 1-entrées:

- Réseau caractéristique du carreau
- Plan de coupe



- Matrice de passage repère global / repère local
- Solution initiale  $(u^*, v^*)$
- Tolérance de chaînage "Tol\_Chain"

**2-Algorithmme :**

1- DEBUT

2-  $u_{tmp} = u^*$  et  $v_{tmp} = v^*$

3- Passer au repère local

4- Si  $u=0$  ou bien  $u=1$  Alors

Faire

-  $v = v_{tmp}$

$$\delta v = \frac{Z_s(u, v) - Z_p}{\frac{\partial Z_s(u, v)}{\partial v}}$$

-  $v_{tmp} = v + \delta v$

Tant Que  $(|v - v_{tmp}| > Tol\_Chain)$

-  $v = v_{tmp}$

Si Non

Faire

-  $u = u_{tmp}$

$$\delta u = \frac{Z_s(u, v) - Z_p}{\frac{\partial Z_s(u, v)}{\partial u}}$$

-  $u_{tmp} = u + \delta u$

Tant Que  $(|u - u_{tmp}| > Tol\_Chain)$

-  $u = u_{tmp}$

Fin Si

5-FIN

**3-Sorties:**

- Solution exacte  $(u, v)$

**C- Procédure de construction et d'orientation des courbes d'intersection**

L'organigramme de la procédure de construction et d'orientation des courbes d'intersection est donnée en annexe II.

**1-entrés:**

- Liste chaînée des segments d'intersection.

**2-Algorithmme :**

1- DEBUT

2- Créer la liste des courbes d'intersection

3- Créer une nouvelle liste chaînée « points d'une même courbe d'intersection »

4- Enlever le premier segment Seg1 de la liste des segments d'intersection

5- Orienter le segment en fonction du vecteur normal sortant

6- Insérer Seg1.origine dans la liste chaînée « points d'une même courbe d'intersection »

7- Pour tous les segments de la liste des segments d'intersection **Faire :**

7.1- Lire le premier segment Seg2 de la liste des segments d'intersection

Si Seg2.origine confondue avec Seg1.extrémité **Alors**

- Enlever le premier segment Seg2 de la liste des segments d'intersection
- Insérer Seg2.origine en fin de la liste « points d'une même courbe d'intersection »
- Seg1.extrémité = Seg2.extrémité

**Fin Si**

**Si Non**

**Si Seg2.extrémité confondue avec Seg1.extrémité Alors**

- Enlever le premier segment Seg2 de la liste des segments d'intersection
- Insérer Seg2.extrémité en fin de la liste « points d'une même courbe d'intersection »
- Seg1.extrémité = Seg2.origine

**Fin Si**

**Si Non**

**Si Seg2.origine confondue avec Seg1.origine Alors**

- Enlever le premier segment Seg2 de la liste des segments d'intersection
- Insérer Seg2.extrémité en tête de la liste « points d'une même courbe d'intersection »
- Seg1.origine = Seg2.extrémité

**Fin Si**

**Si Non**

**Si Seg2.extrémité confondue avec Seg1.origine Alors**

- Enlever le premier segment Seg2 de la liste des segments d'intersection
- Insérer Seg2.origine en tête de la liste « points d'une même courbe d'intersection »
- Seg1.origine = Seg2.origine

**Fin Si**

**Fin de Faire**

**Si Seg1.origine n'est pas confondue avec le premier point de la courbe d'intersection Alors**

- Insérer Seg1.origine en tête de la liste

**Si Seg1.extrémité n'est pas confondue avec le dernier point de la courbe d'intersection Alors**

- Insérer Seg1.extrémité en fin de la liste

8- Si la liste des segments d'intersection n'est pas vide Alors

- aller à 1.

9- FIN.

### 3- Sorties:

- Liste chaînée des courbes d'intersections orientées

## D- Procédure de calcul du pas de plongée de coutournage

L'organigramme de la procédure de construction et d'orientation des courbes d'intersection est donné en annexe II.

### 1-entrées:

- Ensemble des facettes des surfaces à usiner
- Tolérance de surépaisseur de finition TSF
- Données outil : type et dimensions

### 2-Algorithmes :

1- DEBUT

2- Initialiser pas de plongée

- PasPlg = Pmax

3- Pour toutes les facettes coupées par le plan Faire

- 3.1- En fonction du type d'outil Faire

Cas 1 : outil cylindrique

$$- P = tsf \cos \alpha$$

Fin Cas 1

Cas 2 : outil hémisphérique

$$- P = 2\sqrt{2r * tsf - tsf^2} \cos \alpha$$

Fin Cas 2

Cas 3 : outil Torique

En Fonction de la configuration (organigramme Annexe II.) Faire

Cas 1 : intersection Torique/torique

$$- P = 2\sqrt{2r * tsf - tsf^2} \cos \alpha$$

Cas 2 : intersection Torique/torique

$$- P = \left( \sqrt{2rtsf - tsf^2} + \frac{tsf - r(1 - \cos \alpha)}{\tan \alpha} + r \sin \alpha \right) \cos \alpha$$

Cas 3 : intersection Torique/torique

$$- P = \left( \sqrt{2rtsf - tsf^2} + \tan(tsf - r(1 - \sin \alpha) + r \cos \alpha) \right) \cos \alpha$$

Cas 4 : intersection Torique/torique

$$- P = r + \frac{tsf - r(1 - \cos \alpha)}{\sin \alpha}$$

Fin Faire

Fin Cas 3

Fin Faire

3.2- Si  $P < PasPlg$  Alors

$$- PasPlg = P$$

Fin Faire

2- FIN

3-Sorties:

- Valeur du pas de plongée maximum "PasPlg".

## E- Procédure de génération des trajectoires de contournage

Après la construction des contours orientés, la procédure suivante permet la génération automatique des trajectoires de contournage.

1-entrées:

- Contours d'intersection orientés
- Données outil : type et dimensions

2-Algorithmes :

1- DEBUT

2-Si outil cylindrique Alors

- Calculer les contours décalés
- Analyser les contours décalés et éliminer les singularités
  - intra-contour (auto-intersections, des points de rebroussement, discontinuités)
  - inter-contours ( intersections inter-contour, discontinuités)

Si Non

Pour chaque point de contact outil / pièce Faire

En fonction du type d'outil Faire

cas1 : Outil hémisphérique

- calculer le point de contrôle d'outil avec élimination des interférences à l'aide de l'algorithme de finition à l'outil hémisphérique.

cas1 : Outil torique

- calculer le point de contrôle d'outil avec élimination des interférences à l'aide de l'algorithme de finition à l'outil torique.

Fin Faire

Fin Faire

Fin Si

3- FIN

### 3-Sorties:

- Trajectoires d'outil de contournage pour la passe courante.

## F- Procédure de calcul du pas de plongée d'évidement

La génération des trajectoires d'évidement nécessite le calcul préalable du pas de plongé maximum. La procédure développée à cet effet est la suivante :

### 1-entrées:

- Puissance utile machine  $P_u$
- Données outil : type et dimensions
- Donnée matière à usiner
- Vitesse d'avance  $A$ (mm/min)
- Coefficient de recouvrement  $R_c$

### 2-Algorithmes :

1- DEBUT

2- Calculer le coefficient de proportionnalité puissance / débit " $K=P/Q$ "

3- Calcul du pas de plongée maximum " PasEvid "

En fonction du type d'outil Faire

cas1 : outil cylindrique

$$- \text{PasEvid} = \frac{P_u}{K * A * (2R - R_c)}$$

Fin cas 1

cas1 : outil hémisphérique

- PasEvid= « Relations à développer »

Fin cas 1

cas1 : outil torique

- PasEvid= « Relations à développer »

Fin cas 1

Fin Faire

2- FIN

### 3-Sorties:

- Valeur du pas de plongée maximum en évidement "PasEvid".

## G- Procédure de génération des trajectoires globales

Les trajectoires globales sont obtenues à partir de l'agencement judicieux des différentes passes de contournage et d'évidement, précédemment calculées, conformément à la procédure suivante :

### 1-entrés:

- Ensemble des facettes des surfaces à usiner
- Tolérance de surépaisseur de finition T<sub>SF</sub>
- Puissance disponible machine
- Données outil : type et dimensions
- Donnée matière à usiner

### 2-Algorithmes :

#### 1- DEBUT

2-Calculer le pas de plongée maximum  $\Delta Z$  pour les trajectoires d'évidement

- $\Delta Z = f(\text{puissance utile, matière usinée, outil})$

3-Initialiser le compteur de passes CP à 1

- CP=1

4-Pour Z=Z<sub>max</sub> à Z=Z<sub>min</sub> Faire

4.1- Calculer les courbes d'intersections Plan Z/surfaces

4.2- Construire les contours orientés

4.3- Générer les trajectoires de contournages avec éliminations des interférences

4.4- Calculer le pas de plongée de contournage dz

- $dz = f(T_{SF}, \text{Type et dimensions d'outil})$

4.5- Décrementer Z

- $Z = Z - dz$

4.6- Incrémenter le compteur de passe de 1

- CP= CP+1

4.7- Si ( Z < Z<sub>max</sub>-CP\* $\Delta Z$  ) Alors

- Générer les trajectoires d'évidement pour le plan courant

- Sauvegarder les trajectoires d'évidement puis celles de contournage dans le fichier neutre CL-File

Fin Si

Fin Faire

5- FIN

### 3-Sorties:

- Fichier neutre des trajectoires d'outils.

## 2.4.3 Structure de données

### *Structures de données d'entrées*

Les algorithmes opèrent sur la liste chaînée des segments d'intersection (figure

2.10), construite lors de la détermination des points d'intersections plan/facettes.

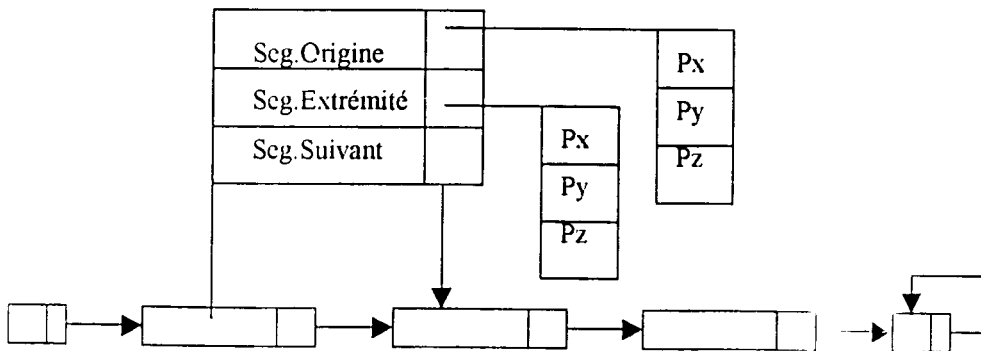


Figure 2.10 Structure de données des segments d'intersections

### Structures de données de sorties

L'algorithme construit une liste chaînée des courbes d'intersection avec orientation automatique. Pour chaque courbe, les points d'intersections sont ordonnés puis mémorisés dans une autre liste chaînée suivant la structure ci-dessous:

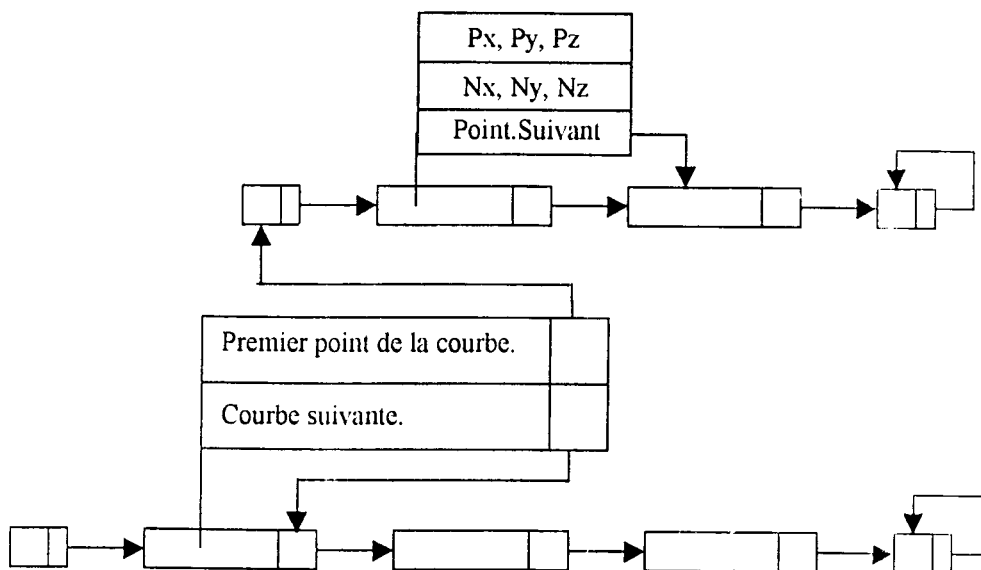


Figure 2.11 Structure de données des courbes d'intersections

### 2.4.4 Exemples d'exécution

La Figure 2.12a représente les courbes d'intersections obtenues en coupant la surface par un plan perpendiculaire à l'axe z. L'algorithme permet de calculer toutes les portions de courbes d'intersection. Dans la figure 2.12b, la surface est coupée par

un réseau de plans parallèles, perpendiculaires à l'axe z.

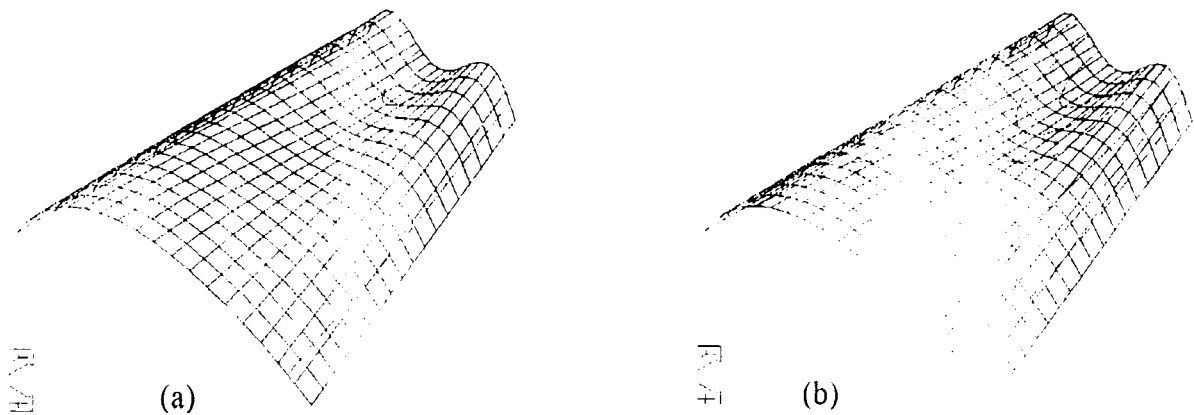


Figure 2.12 Courbes d'intersection surface/plan  
(a) et surface/réseau de plans parallèles (b).

La Figure 2.13 représente les trajectoires d'usinage d'ébauche, pour une passe d'évidement avec les différentes passes de contournage correspondantes.

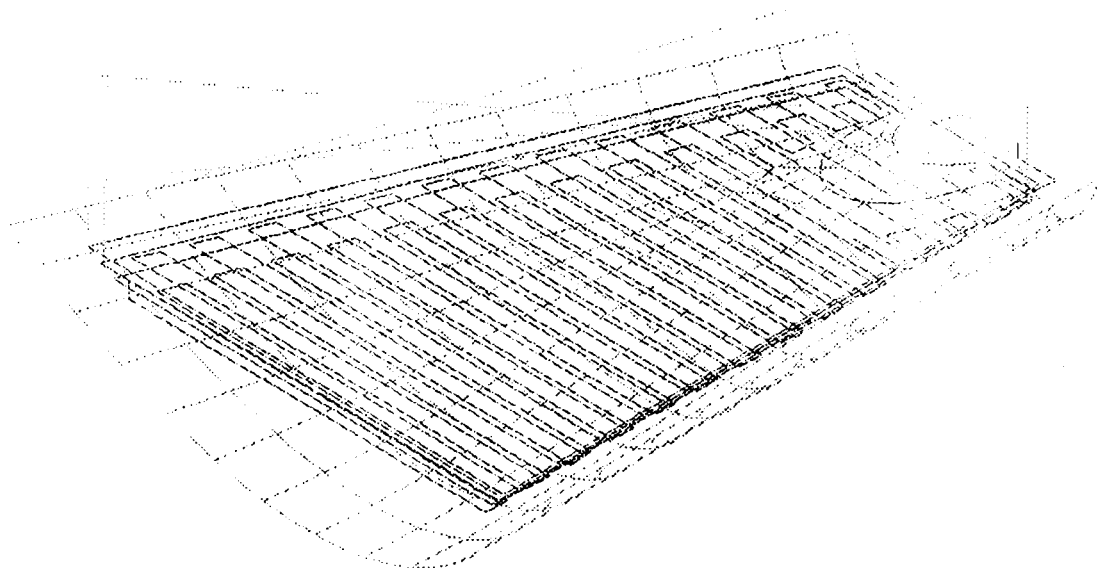


Figure 2.13 Trajectoire d'outil correspondant à  
une passe d'évidement.

Les trajectoires d'ébauche globales obtenues sont présentées en Figure 2.14.

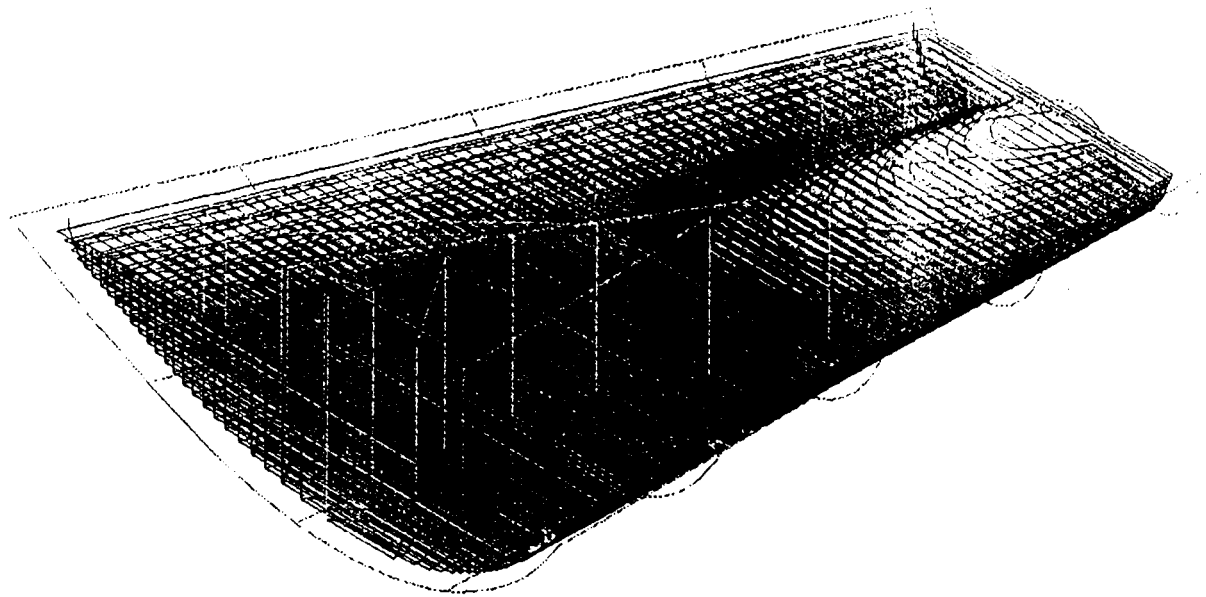


Figure 2.14 Trajectoire d'ébauche.



## 2.5 Module d'usinage de Finition

### 2.5.1 Organigramme général

Les algorithmes de calculs des différentes parties du module de finition dont L'organigramme général est donné en annexe II, sont décrits en détails dans ce qui suit :

### 2.5.2 Algorithmes et Implémentation

#### A- Procédure de Calcul du pas transversal

Le calcul du pas transversal se fera en fonction du type d'outil. Pour chaque facette coupée par le plan de coupe, un pas maximum permettant le respect de la valeur de hauteur de crête prescrite est calculé. La valeur minimale des différents pas ainsi calculés sera prise comme pas d'incrémentement transversal. Pour l'outil torique, l'algorithme de Rubio [Rubio '94] dont le détail a été donné aux chapitres 2 et 3 sera implémenté comme suit.

#### 1-Entrées:

- Ensemble des facettes des surfaces à usiner
- Hauteur de crête maximale prescrite
- Plan de coupe courant
- Données outil : type et dimensions

#### 2-Algorithmes :

##### 1- DEBUT

2- Initialiser le pas transversal "pas\_trans"  
 - pas\_trans=R

3- Pour toutes les facettes coupées par le plan de coupe **Faire**

**Si** outil hémisphérique **Alors**

$$- pas = 2\sqrt{h(2R - h)} \sin \alpha$$

**Si Non**

**Si** courbure positive et l'outil descend **Alors**

**En fonction de la configuration Faire**

**cas1:** intersection torique1 / torique2

$$- pas = |(\rho + r)(\sin 2(\beta - \alpha) - \sin \alpha)|$$

**Fin Cas**

**cas2:** intersection circulaire1 / plane2

$$- pas = |(\rho + r)(\sin \alpha - (\rho + h)\sin \beta - r)|$$

**Fin Cas**

**cas3:** intersection plane1 / torique2

$$- \cos(\delta) = \frac{(\rho + h)^2 + (\rho + R)^2 - R^2}{2(\rho + R)(\rho + h)}$$

$$- pas = |(\rho + r)(\sin \alpha - \sin(\beta + \delta))|$$

**Fin Cas**

*cas4: intersection plane1 / plane2*

$$- pas = |(\rho + r)\sin \alpha - (\rho + h)\sin \beta - r|$$

**Fin Cas**

**Fin Si**

*Si courbure positive et l'outil monte Alors*

*En fonction de la configuration Faire*

*cas1: intersection torique1 / torique2*

$$- pas = |r - r - 2b - a - \sin a|$$

**Fin Cas**

*cas2: intersection circulaire1 / plane2*

$$- pas = |(\rho + r)\sin \alpha - \sqrt{(\rho + r)^2 - ((\rho + h)\cos \beta + r)^2}|$$

**Fin Cas**

*cas3: intersection plane1 / torique2*

$$- \cos(\delta) = \frac{(\rho + h)^2 + (\rho + r)^2 - r^2}{2(\rho + r)(\rho + h)}$$

$$- pas = |(\rho + r)(\sin(\beta - \delta) - \sin \alpha)|$$

**Fin Cas**

*cas4: intersection plane1 / plane2*

$$- pas = |(\rho + r)\sin \alpha - \sqrt{(\rho + r)^2 - ((\rho + h)\cos \beta + r)^2}|$$

**Fin Cas**

**Fin Si**

*Si courbure négative et l'outil descend Alors*

*En fonction de la configuration Faire*

*cas1: intersection torique1 / torique2*

$$- pas = |(r - r)(\sin(2b - a) - \sin a)|$$

**Fin Cas**

*cas2: intersection circulaire1 / plane2*

$$- pas = |(\rho - r)\sin \alpha - (\rho - h)\sin \beta - r|$$

**Fin Cas**

*cas3: intersection plane1 / torique2*

$$- \cos(\delta) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

$$- pas = |(\rho - r)(\sin \alpha - \sin(\beta - \delta))|$$

**Fin Cas**

*cas4: intersection plane1 / plane2*

$$- pas = |(\rho - r)\sin \alpha - (\rho - h)\sin \beta - r|$$

**Fin Cas**

**Fin Si**

*Si courbure négative et l'outil monte Alors*

*En fonction de la configuration Faire*

*cas1: intersection torique1 / torique2*

$$- pas = |(r - r)(\sin(2b - a) - \sin a)|$$

*Fin Cas*

*cas2: intersection circulaire1 / plane2*

$$- pas = \left| (\rho - r) \sin \alpha - \sqrt{(\rho - r)^2 - (r - (\rho - h) \cos \beta)^2} \right|$$

*Fin Cas*

*cas3: intersection plane1 / torique2*

$$- \cos(\delta) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

$$- pas = |(\rho - r)(\sin(\beta + \delta) - \sin \alpha)|$$

*Fin Cas*

*cas4: intersection plane1 / plane2*

$$- pas = \left| (\rho - r) \sin \alpha - \sqrt{(\rho - r)^2 - (r - (\rho - h) \cos \beta)^2} \right|$$

*Fin Cas*

*Fin Si*

*Fin Si*

*Fin Faire*

4- Affecter la valeur min calculée au "pas\_trans"

- Si (pas < pas\_trans) Alors

- pas\_trans = R

5- FIN

3-Sortie:

- Pas transversal maximum pour le plan de coupe suivant.

A partir de l'analyse des cas extrêmes, définies par les valeurs  $h_0$  et  $h_1$ , les organigrammes donnés en annexe II permettent de déterminer le cas de figure d'intersection.

$$\begin{cases} h_0 = \left| \sqrt{(\rho + r)^2 + r^2 - 2(\rho + r)r \sin \alpha} - \rho \right| \\ h_1 = \left| \sqrt{\rho(\rho + 2r) + r^2 - \sin^2 \beta} - \rho - r \sin \beta \right| \end{cases}$$

## B- Procédure de Calcul des points centre d'outil avec élimination des interférences

### 1-entrés:

- Ensemble des facettes des surfaces à usiner
- Point de contact outil / pièce, et vecteur normal à la surface en ce point
- Données outil : type et dimensions

### 2-Algorithmes :

1- **DEBUT**

2- Calculer le point centre d'outil initial  $CLP_i$  correspondant au point de contact  $CCP_i$ , donné

3- **Si** outil hémisphérique **Alors**

$$- CLP_i = CCP_i + rn_i$$

**Si Non**

$$- CLP_i = CCP_i + rn_i + (R - r)v_i - rT_{ax}$$

**Fin Si**

4- initialiser  $\Delta p$  à zéro

5- **Pour** chaque facette **Faire**

5.1 **Si** la facette appartient à la zone de contact **Alors**

**Si** l'outil interfère avec le plan infini de la facette **Alors**

**Si** le point d'interférence appartient à l'aire de la facette **Alors**

**Si** outil hémisphérique **Alors**

- Calculer la valeur de l'incrément  $\Delta p_i$  de la position de l'outil le long de la normale

**Si** outil Torique **Alors**

- Calculer la valeur de l'incrément  $\Delta p_i$  de la position de l'outil le long de l'axe z

**Si**  $\Delta p < \Delta p_i$  **Alors**

$$- \Delta p = \Delta p_i$$

**Si Non**

**Si** l'interférence a lieu sur l'un des cotés de la facette **Alors**

**Si** outil hémisphérique **Alors**

- Calculer la valeur de l'incrément  $\Delta p_i$  de la position de l'outil le long de la normale

**Si** outil Torique **Alors**

- Calculer la valeur de l'incrément  $\Delta p_i$  de la position de l'outil le long de l'axe z

**Si**  $\Delta p < \Delta p_i$  **Alors**

$$- \Delta p = \Delta p_i$$

**Fin Si**

**Fin Si**

**Fin Si**

**Fin Si**

**Fin de Faire**

- Corriger la position de l'outil

$$- CLP_i = CLP_i + \Delta p_i$$

4- Si autre point de contact Alors aller a 2

5- FIN

### 3-Sortie:

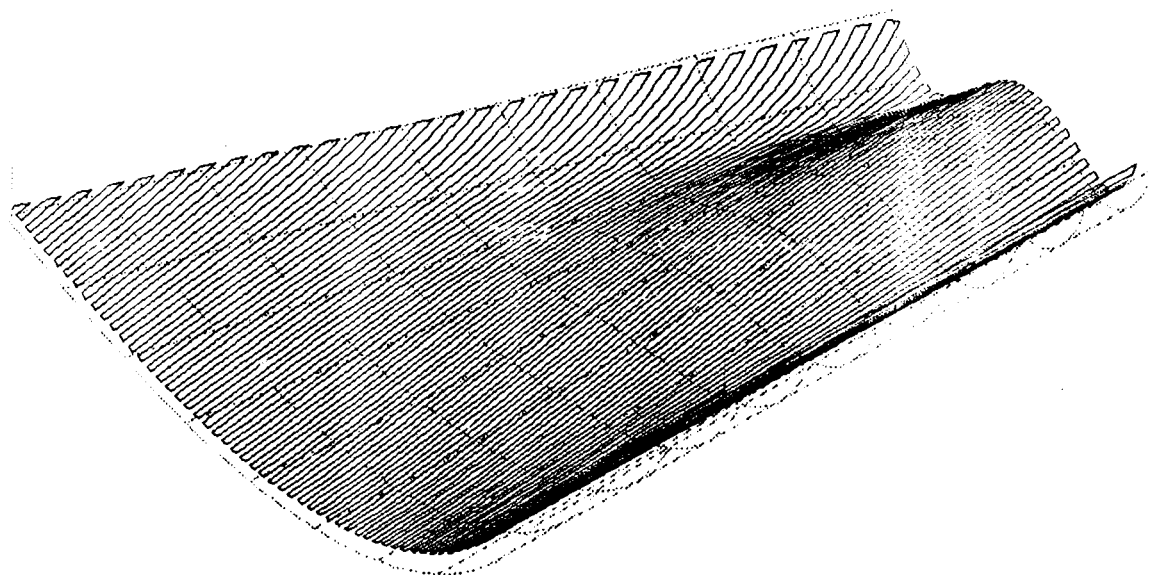
- Point centre d'outil correspondant au point de contact donné.

### 2.5.3 Structure de données

Les structures de données utilisées pour la génération des trajectoires de finition sont similaires à celle utilisées pour les trajectoires d'ébauche, à savoir la liste chaînée des segments d'intersection en entrée (*Figure 2.10*) et en sortie la liste chaînée des courbes d'intersections (*Figure 2.11*) dont l'orientation dépend du type de balayage choisi. Pour un balayage unidirectionnel les courbes doivent être orientées dans le même sens, alors que pour un balayage en zigzag, le sens sera alterné d'un plan de coupe à l'autre.

### 2.5.4 Exemples d'exécution

Les trajectoires de finition générées sont présentées en *Figure 2.15*.



*Figure 2.15 Trajectoire d'usinage de finition par plans parallèles.*

## 2.6 Validation

### 2.6.1 Objet :

Cette validation a pour objet de montrer les performances de la méthode proposée par rapport à celles d'un logiciel industriel de CFAO des formes complexes en comparant les résultats obtenus selon les critères établis au chapitre 2.

### 2.6.2 Chaîne expérimentale du processus de validation mise en place

Les éléments mis en place dans le cadre de ce processus de validation sont configurés dans l'organigramme de la figure 2.16 ci-dessous. La création du modèle de surface CAO s'effectue sous le logiciel « AutoCAD », l'application sera ensuite actionnée pour générer les trajectoires d'outils au format APT, après avoir introduit les paramètres technologiques. Pour le même modèles CAO et pour les mêmes conditions d'usinage, nous générons les trajectoires d'usinage au même format « APT » sous le logiciel SURFCAM [*Surfcam '97*] en vue d'une comparaison.

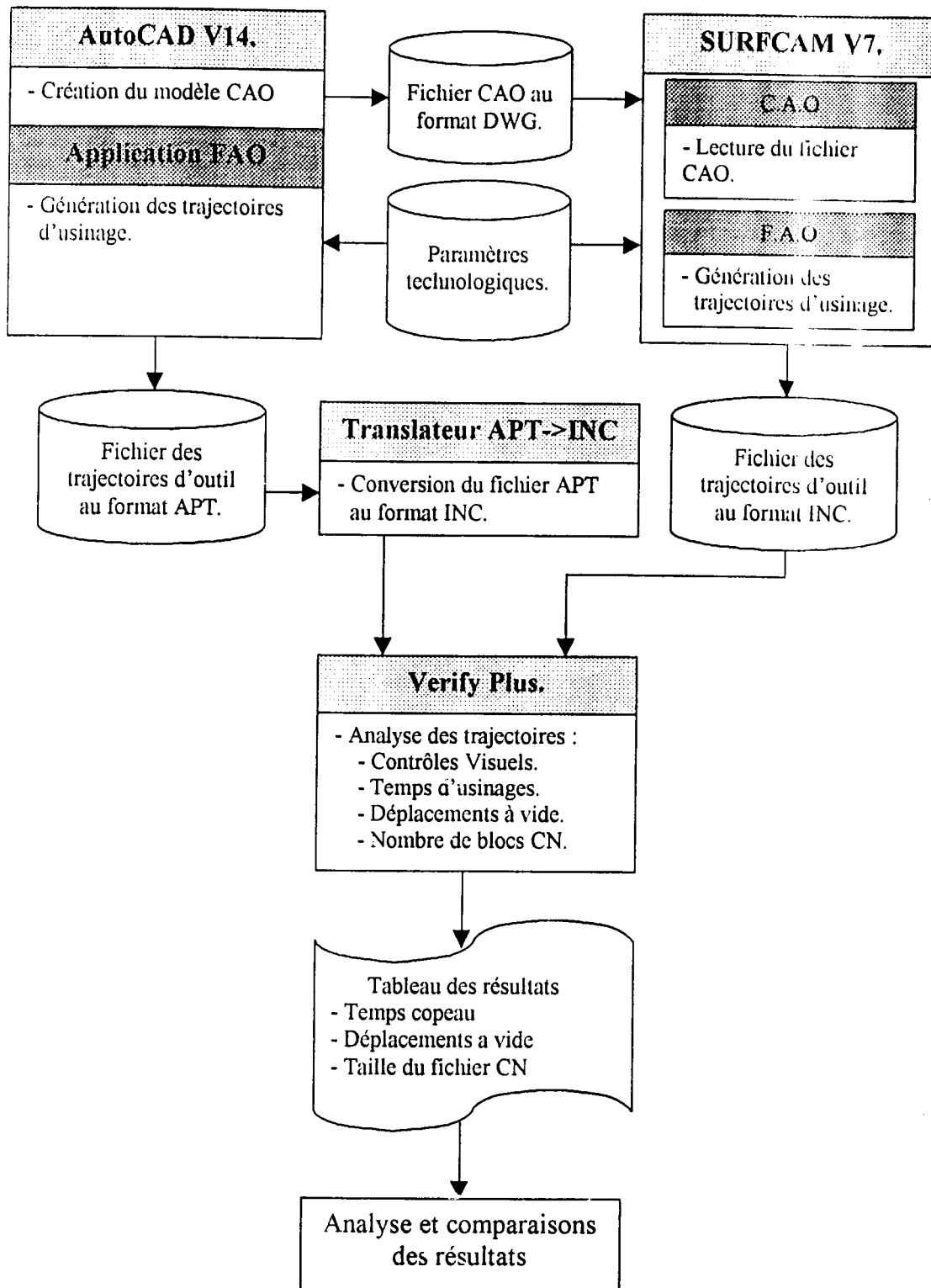


Figure 2.16 Organigramme du processus de validation.

### 2.6.3 Présentation du logiciel *Surfcam*

Le logiciel de CFAO "*SURFCAM*" est destiné principalement à la modélisation et l'usinage des formes complexes. Il est né des développements réalisés en interne chez BOEING et est édité depuis 1987 par la société américaine « SURFWARE Inc. ». Il représente l'un des systèmes les plus performants tournant sur PC. Ses possibilités de modélisation sont très vastes et permettent la génération automatique des trajectoires aussi bien pour les travaux de tournage, de fraisage de 2 à 5 axes, de découpe laser et à jet d'eau, ainsi que pour l'électroérosion à fil 2 et 4 axes.

### 2.6.4 Conditions expérimentales

Pour mener à bien cette étude, il convient de définir des conditions expérimentales à même de garantir la cohérence des résultats. Dans un premier temps, des limitations communes à « *SURFCAM* » et à la méthode proposée sont mis en évidence, à partir d'un même modèle CAO. La seconde étape nous permettra de mettre en relief des possibilités étendues de la méthode proposée par rapport à celles de « *SURFCAM* », en dernier lieu nous réaliserons l'analyse des trajectoires d'usinage générées pour les mêmes modèles CAO et dans les mêmes conditions d'usinage, c'est-à-dire avec les mêmes paramètres machine, les mêmes outils et dans les mêmes conditions de coupe et de passe, afin d'évaluer les temps d'usinages ainsi que les volumes de fichiers CN obtenus.

### 2.6.5 Résultats obtenus

#### 2.6.5.1 Limitations communes à «*SURFCAM*» et à la méthode proposée

Concernant la prise en charge de l'erreur de crête, les deux systèmes reposent sur le critère du cas le plus défavorable, une optimisation supplémentaire reste donc possible. A ce niveau, l'utilisation des trajectoires à courbes de remplissage pourrait apporter un gain de productivité appréciable.



### 2.6.5.2 Possibilités étendues de la méthode proposée

A la différence du logiciel «SURFCAM», la méthode proposée ne présente aucune restriction quant aux modèles mathématiques et aux degrés des surfaces utilisées.

De plus, au niveau des tolérances, le modèle de prédiction des erreurs du processus d'usinage est à même de permettre à l'utilisateur de spécifier une tolérance de forme proche du langage du mécanicien et adaptée au processus d'usinage.

Pour la phase d'ébauche, la méthode proposée permet la mise en œuvre de critères d'optimisation, a titre d'exemple, nous avons mis en œuvre un module de calcul du pas de plongée qui tient compte de la puissance disponible sur la broche; d'où une utilisation rationnelle des moyens de production.

La possibilité d'élimination systématique de la passe de demi-finition, avec contrôle de la régularité de la surépaisseur de finition *TSF* représente un autre atout important par rapport à SURFCAM. Enfin, le système élaboré permet la génération automatique des données nécessaires à une reprise locale d'usinage sans interférences.

Au cours des essais comparatif, une limitation des algorithmes de constructions des courbes d'intersections implémenté dans «*SURFCAM*» à aussi été décelé (fig.2.17b), ce qui démontre la robustesse de l'algorithme proposé dans la méthodes que nous avons développée (fig.2.17a),.

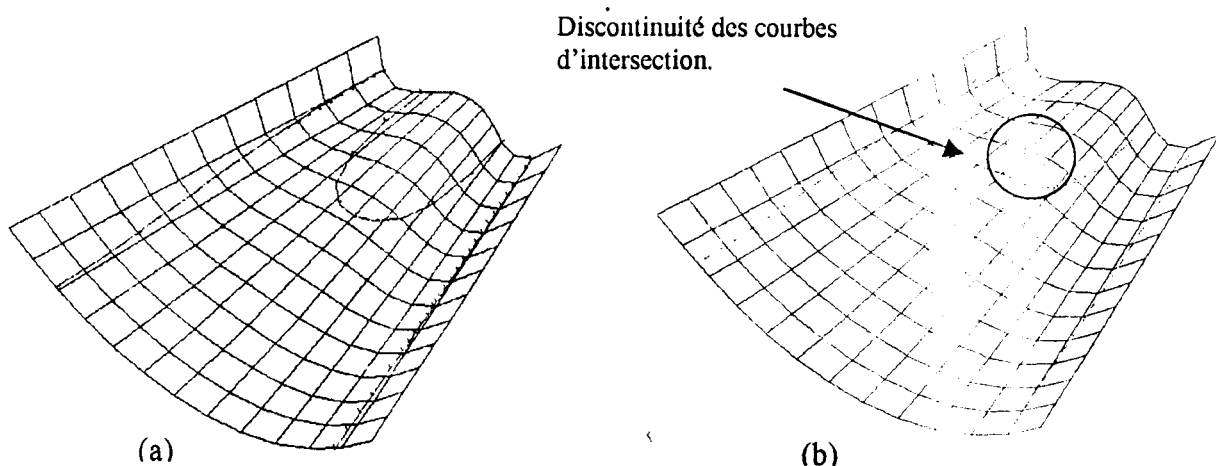


Figure 2.17 Comparaison des résultats de construction des courbes d'intersection (a) avec le système élaboré, (b) avec SURFCAM.

### 2.6.5.3 Analyse des trajectoires d'usinage obtenu pour un même modèle CAO

L'analyse visuelle des trajectoires d'ébauche obtenues avec SURFCAM, met bien en évidence les inconvénients des trajectoires par décalage de contour. L'influence de la forme géométrique finale de la surface étant propagée intégralement par offsetisation. Ce type de trajectoires donne lieu à un nombre très important d'interpolations linéaires, injustifiées pour les zones d'évidement (partie centrale sur la figure 2.18). Les critères antagonistes de *débit maximum de copeau* et de *hauteur d'aspérités minimales* sont aussi mis en exergue de façon très claire. En effet, une augmentation du pas de plongée entraîne inévitablement des hauteurs d'aspérités en forme de "marches d'escaliers" importantes (figures 2.18), qui imposent une phase de demi finition à même de garantir un usinage de finition régulier (figures 2.19), alors qu'une réduction de ces aspérités passe automatiquement par une perte de productivité due à la sous-exploitation des machines outils et des outillages.

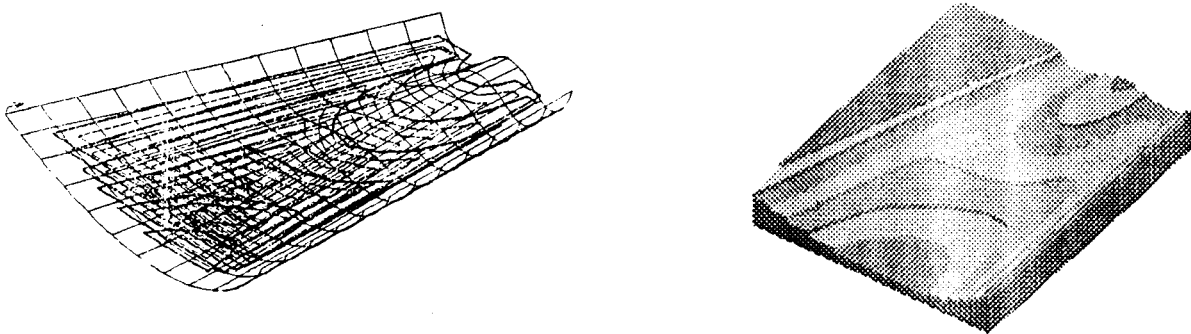


Figure 2.18 Trajectoires d'ébauche obtenues avec SURFCAM. (a) Trajectoires, (b) Usinage simulé.

Sur le plan quantitatif, les résultats numériques comparés, regroupés dans le tableau 2.1 ci-dessous, pour un même modèle CAO, montrent un gain de productivité significatif. Par contre, les gains en taille de fichier CN sont très importants en

ébauche, comparé à la taille de la surface usinée (84mm\*100mm). En effet ce gain devrait augmenter avec le volume total de copeaux à enlevé et de l'étendue de la surface, le caractère global étant amplifié par rapport au caractère local lié à la forme géométrique de la surface.

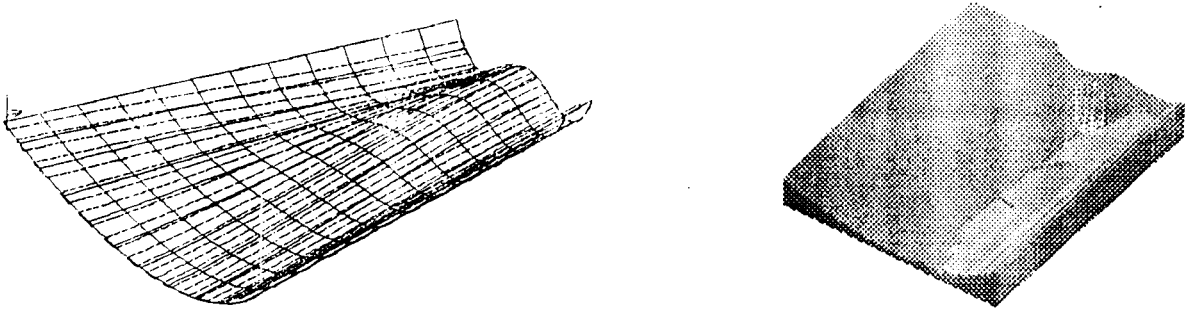


Figure 2.19 Trajectoires de semi-finition obtenues avec SURFCAM. (a) Trajectoires, (b) Usinage simulé.

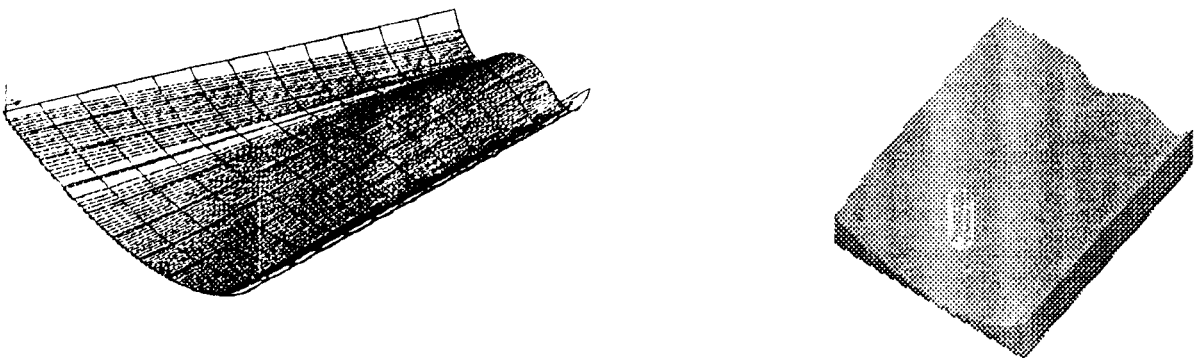


Figure 2.20 Trajectoires de finition obtenues avec SURFCAM. (a) Trajectoires, (b) Usinage simulé.

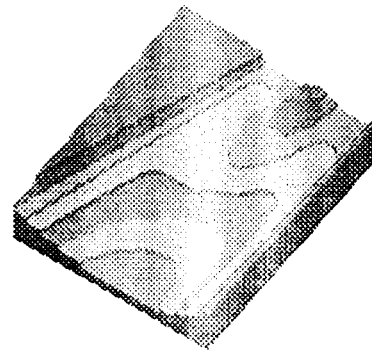
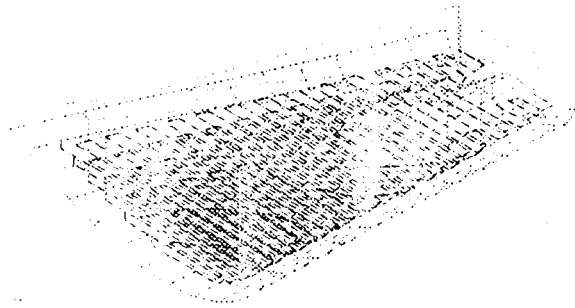


Figure 2.21 Trajectoires d'ébauche obtenues avec la méthode proposée pour une TSF très importante. (a) Trajectoires, (b) Usinage simulé.

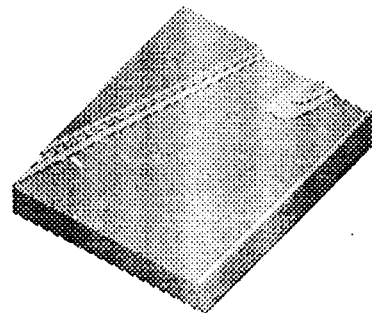
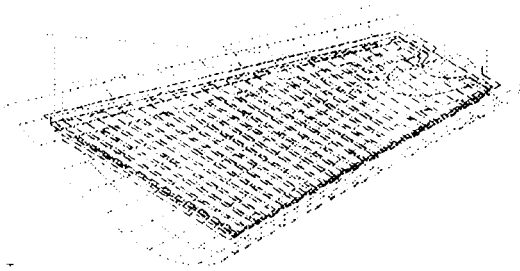


Figure 2.22 Portion des trajectoires d'ébauche /semi-finition obtenues avec la méthode proposée pour une TSF de 0.5mm. (a) Trajectoires, (b) Usinage simulé.

Modèles CAO	Phase usinage	Paramètres Technologiques						Temps d'usinage			Taille fichier CN (Ko)				
		Coupe			Outil									Tol.	
		Vc	Vf	a	rc	Type	D	Tc	h	1	2	1	2		Gain
Modèle 1	Ebauche	994	99	2.5	3	Cyl	10	0.025	*	34	63	12	9	42%	52%
	F/2	994	159	0.5 <sup>1</sup>	*	Hém	8	0.125	0.5	18	14	53	-3%	-4%	
	Finition	994	159	0.1	*	Hém	8	0.025	0.025	67	65	67	76		
		TOTALE :							101	146	67	76			

**Légendes :**

Vc : Vitesse de coupe  
 Cyl. : Fraise cylindrique  
 Tc : Tolérance de conception

Vf : Vitesse d'avance  
 Hém. : Fraise hémisphérique  
 h : Hauteur de crête.

a : Profondeur de passe Rc : Coefficient de recouvrement  
 D : Diamètre d'outil

Tableau 2.1: comparaison des résultats de génération des trajectoires d'outils obtenues avec le système élaboré (1) et ceux obtenues avec « SURFCAM » (2).

### 2.6.6 conclusions

A la lumière des différents essais comparatifs et des résultats obtenus ci-dessus, nous pouvons conclure :

- Le gain considérable en temps d'usinage en ébauche est permis grâce à la mise en œuvre de la méthode de planification proposée.
- Les trajectoires parallèles en zigzag optimisées permettent une réduction très importante de la taille des fichiers de code CN.
- Les trajectoires d'usinage de finition restent semblables sur le plan géométrique, par contre une comparaison des qualités d'usinage, tant sur le point des erreurs de forme que sur le plan de la qualité micro géométrique nécessitent la mise en œuvre de procédures et de moyens de contrôle adapté qui ne sont disponibles au niveau de notre laboratoire.
- Néanmoins les algorithmes de calculs et de prédiction des tolérances d'erreurs mis en œuvre devraient donner des résultats au moins comparables, si ce n'est meilleurs que ceux obtenus avec « SURFCAM ».

*Conclusion*

*générale*

## Conclusion

Notre travail présente pour la première fois une synthèse bibliographique approfondie sur les méthodes et techniques d'usinage 2/5 axes, présentée sous forme d'une classification systématique qui situe bien l'état de l'art dans le domaine. Ainsi, nous avons pu recenser les différents types de planification des trajectoires d'outils et les méthodes de calculs correspondantes qui ont permis de mettre en exergue les cercles de problèmes qui demeurent encore posés. Il s'agit notamment de la compatibilité des modèles CAO et FAO, de la difficulté de maîtrise des tolérances de CFAO, du problème du choix adapté d'une technique de planification qui garantit la productivité et la qualité de l'usinage et enfin du degré d'efficacité des algorithmes de gestion des interférences et de reprise d'usinage.

Le Système d'usinage élaboré peut être considéré comme une innovation et est structuré en deux modules complémentaires. D'une part, la maîtrise d'une construction flexible de formes complexes est assurée par le module de conversion de surfaces qui permet le passage d'un modèle de Bézier de degré élevé à un autre Modèle de Bézier de degré plus faible ainsi que le passage des modèles NURBS vers des Bézier de degré faibles. Ceci nous a permis de mettre à profit tous les avantages caractérisant ce modèle telles que la facilité de mise en œuvre et la stabilité numérique des algorithmes tout en permettant une prédiction des tolérances de corde et de crête et en garantissant une compatibilité totale les modèles CAO-FAO.

D'autre part, le module d'usinage développé obéit aux critères de performances requis. Il est basé sur une approche nouvelle pour la génération et l'optimisation des trajectoires d'outils en usinage d'ébauche et finition de surface en 2<sup>1/2</sup>/3Axes avec respect des tolérances de CFAO. La méthode d'usinage d'ébauche proposée permet en effet l'optimisation des trajectoires d'outils en 2 axes 1/2 pour l'usinage de surfaces gauches composites. Elle utilise un algorithme de calcul des courbes d'intersections surface/plan, basé sur une technique de discrétisation auto-adaptative de la surface en facettes qui assure une génération robuste de trajectoires. En outre, cette nouvelle approche permet d'éliminer la passe de demi-finition et d'optimiser les déplacements à vide. En finition, les trajectoires d'usinage par plans parallèles à l'outil hémisphérique ou torique en 3Axes générées garantissent le respect des tolérances de CFAO prescrites ainsi que l'élimination systématique des interférences, avec la possibilité éventuelle de reprises locales de l'usinage.

Le système possède une structure modulaire dont les algorithmes sont présentés sous forme procédurale en langage clair, permettant une mise en œuvre et une réutilisation aisée. La base de donnée interne, conçue avec des structures de données dynamiques, a permis d'atteindre des niveaux de performance très élevés. Ce système a été implémenté sous le logiciel « AutoCADV14 », ce qui peut être considéré comme une extension de ce logiciel à l'usinage de surfaces gauches.



---

A la lumière des résultats de l'étude comparative, nous avons enregistré des améliorations substantielles notamment en terme de compatibilité totale CAO-FAO, en matière de productivité pour les opérations d'ébauche et enfin au niveau du respect de la qualité prescrite pour les opérations de finition.

Toutefois, les algorithmes de construction des contours fermés restent à améliorer pour gérer le problème des cas singuliers des courbes ouvertes aux frontières de surfaces qui demeure encore posé. La productivité de l'usinage d'ébauche peut être également augmentée par l'implémentation d'un algorithme d'optimisation du choix de la direction de balayage des parcours d'évidement en zigzag. Le développement et l'implémentation de modèles de prédiction des tolérances pour des surfaces usinées 3/5 tenant compte des paramètres caractéristiques relatifs à l'outil et aux types de trajectoires est une autre opportunité à exploiter en vue d'une généralisation du module d'usinage de finition qu'il convient d'étendre à cet effet à l'usinage 4 /5 axes.

## *Références Bibliographiques*

**Références Bibliographiques**

- [Akam et Limam '97] H. Akam et B. Limam 'Usinage de poches 2/3 axes sous contraintes des paramètres technologiques' mémoire de fin d'études d'ingénieurs labo CFAO INGM 1997.
- [Asma '98] F. Asma 'Etude comparative des méthodes de réduction de degrés des courbes gauches : Application à l'échange de données entre systèmes' Mémoire de magistère. labo CFAO, Université de Boumerdès, 1999.
- [Asma et al. '99a] F. Asma, I. Belaidi, G. Ishiomin et G. Coffignal 'Réduction de degré des courbes de Bézier de degré élevé : application à l'échange de données entre systèmes de CFAO'. A paraître.
- [Asma et al. '99b] F. Asma, I. Belaidi, G. Ishiomin et G. Coffignal 'Réduction de degré des courbes de Bézier de degré élevé : une étude comparative des méthodes publiées'. A paraître.
- [Aomura '90] S. Aomura and T. Uehara 'Self intersection of an offset surface' Computer-Aided Design Vol22 N°7 September 1990
- [Autocad '94a] Autodesk Inc. 'Manuel de l'utilisateur : AutoCAD R13' Autodesk Inc. 1994.
- [Autocad '94b] Autodesk Inc. 'Manuel de personnalisation : AutoCAD R13' Autodesk Inc. 1994.
- [Barmhill et al. '87] R.E. Barmhill, G. Farin, M. Jordan, B.R. Piper 'surface/surface intersection' Computer Aided Geometric Design N°4 1987.
- [Barmhill et al. '90] R.E. Barmhill, S.N. Kersey 'A marching method for parametric surface/surface intersection' Computer Aided Geometric Design N°7 1990.
- [Belaidi '98] I. Belaidi 'Réduction de degré des courbes de Bézier, Qualité d'équivalence en relation avec des paramètres d'usage' Thèse de doctorat, LMS-ENSAM-CER de Paris 1998.
- [Belaidi et al. '99a] I. Belaidi, Guy Ishiomin et Gérard Coffignal 'Prédiction des tolérances de forme et de rugosité des courbes de Bézier'. A paraître
- [Belaidi et al. '99b] I. Belaidi, Guy Ishiomin et Gérard Coffignal 'Critères d'équivalence de CFAO des courbes de Bézier de degré élevé'. 1er Colloque international de productique, organisé par l'Université Mouloud Mammeri de Tizi-Ouzou et l'ENIT de Belfort (France), mai 1998.
- [Belaidi et al. '99c] I. Belaidi, Guy Ishiomin et Gérard Coffignal 'Réduction de degré des courbes de Bézier non rationnelles. Aperçu sur les applications potentielles en CFAO'. En préparation.

- [Bensalah '90] S. Bensalah 'Equivalence et algorithmique des transformations dans les modèles mathématiques pour la conception et la fabrication assistées par ordinateur des courbes et des surfaces', Thèse de doctorat, ENSAM 1990
- [Bensalah '93] S. Bensalah 'Développements pour la réduction du degré' Revue de CFAO et d'infographie. Vol.8 N°1 1993.
- [Boehm '80] W. Boehm 'Inserting new knots into B-Spline curves' Computer Aided Design Vol 12 N°4 1980.
- [Boehm '81] W. Boehm 'Generating the Bézier points of B-spline curves and surfaces' Computer Aided Design Vol 13 N°6 1981.
- [Boehm et Prautzsch '85] W. Boehm and H. Prautzsch 'The inserting algorithm' Computer Aided Design Vol 17 N°4 1985.
- [Bobrow '85] J. Bobrow 'NC machining tool path generation from CSG part representation' Computer Aided Design Vol 17 N°2 1985.
- [Bret '88] M. Bret 'Images de synthèse : Méthodes et algorithmes pour la réalisation d'images numériques' DUNOD Informatique 1988.
- [Béziers '87] P. Bézier 'Mathématiques et C.A.O : courbes et surfaces' Ed. Hermés (1987)
- [Cho et al. '94] J.H Cho, M.W. Cho and K. Kim 'Volumetric error analysis of multi-axis machine tool machining a sculptured surface workpiece' International Journal of Production Research Vol32 N°2 1994
- [Choi et al. '88a] B.K Choi, J.S Hwang and C.S June 'Compound surface modeling and machining' Computer-Aided Design Vol20 N°3 June 1988.
- [Choi et al. '88b] B.K Choi, H.Y Shin Y.I Yoon and J W Lee 'Triangulation of scattered data in 3D space' Computer-Aided Design Vol20 N°5 June 1988
- [Choi et Jun '89] B.K Choi and C S Jun 'Ball-end cutter interference avoidance in NC machining of sculptured surfaces' Computer-Aided Design Vol21 N°6 July 1989
- [Choi et al. '93] B.K Choi, J.W Park and C.S Jun 'Cutter-location data optimization in 5-axis surface machining' Computer-Aided Design Vol25 N°6 June 1993
- [Chou '92] J. Chou and D.C.H Yang 'On the generation of coordinated motion of five-axis CNC/CMM machines' Journal of Engineering for Industry Vol114 February 1992
- [Cox et al. '94] J.J Cox, Y. Tekezaki, H.R.P Ferguson, K E Kohkonen and E L Mulkay ' Space-filling curves in tool-path applications ' Computer-Aided Design Vol26 N°3 March 1994

- [Dahmane '93] N. Dahmane (1993) «Optimisation des opérations d'ébauche dans le cas du fraisage dans la masse» Thèse de doctorat ENSAM Paris N°93-05.
- [Du et al. '90] W. Du and F J M Schmit 'On the G1 continuity of piecewise Bézier surfaces: a review with new results ' Computer-Aided Design Vol 22 N° 9 1990
- [Eck '93] M. Eck 'Degree reduction of Bézier curves' Computer Aided Geometric Design, N°10 1993.
- [Eck '95] M. Eck 'Least squares degree reduction of Bézier curves' Computer Aided Design, Vol 27 N°11 Nov 1995
- [Farouki '86] R.T Farouki 'The approximation of non degenerate offset surfaces' Computer Aided Geometric Design N°3 1986
- [Faux et pratt '87] I.D Faux and M.J Prat 'Computational geometry for design and manufacture' Chichester :Ellis Horwood 1987.
- [Filip '90] D.J Filip 'Determining the orientation of closed planar curves' Computer-Aided Design Vol22 N°7 March 1990
- [Gourdin et Boumahrat '91] A. Gourdin et M. Boumahrat 'Méthodes numériques appliquées' O.P.U (1991)
- [Grandine et Klein '97] A.T Grandine and F.W Klein 'A new approach to the surface intersections problem' Computer-Aided Geometric Design N°14, 1997
- [Hassaid et Belaidi '98] B. Hassaid et I. Belaidi 'Usinage d'ébauche de surfaces gauches en 2 axes 1/2' 1<sup>er</sup> colloque national sur la productique Tizi Ouzouz, Algérie Mai 1998.
- [Hassaid et Belaidi '99] B. Hassaid et I. Belaidi 'Optimisation de l'usinage d'ébauche de surfaces gauches par combinaison de topologie de trajectoires' .
- [Hatna '95] A. Hatna 'Contribution à l'élaboration d'une méthodologie d'évidement de poches complexes et usinage sur surfaces gauches' Mémoire de magistère. labo CFAO, Université de Bumerdès, 1995.
- [Held et al. '94] M. Held, G. Lukcas and L. Andor 'Pocket machining based on contour-parallel tool paths generated by means of proximity maps' Computer-Aided Design Vol26 N°3 March 1994
- [Houghton et al. '85] E.G Houghton, R.F Emmett, J.D Factor and C.L Sabharwal 'Implementation of a divide-and-conquer method for intersection of parametric surfaces' Computer Aided Geometric Design N°2 1985
- [Hoschek '85] J. Hoschek 'Offset curves in the plane' Computer-Aided Design Vol17 N° 2 arch 1985

- [Hochek'88] J. Hoschek 'Intrinsic parameterization for approximation' *Computer Aided Geometric Design* N°5 1988.
- [Hochek et al.'89] J. Hoschek, F.J. Schneider, P.Wassum 'Optimal approximate conversion of spline surfaces' *Computer Aided Geometric Design*, N°6 1989
- [Hochek'90] J. Hoschek 'Exact approximate conversion of spline curves and spline surfaces' *Computation of Curves and Surfaces*, W.Dahmen et al 1990
- [Hwang'92] J.S Hwang 'Interference-free tool-path generation in the nc machining of parametric compound surfaces' *Computer-Aided Design* Vol24 N°12 December 1992
- [Kim et al.'88] K. Kim and J.E Biegel 'An integrated approach to sculptured surface design an manufacture' *Computers Industry Engineering* Vol 14 N°3 1988
- [Juran'74] J.M Juran 'Quality control handbook' Mc Graw Hill, New York 19774.
- [Kim et Ko'94] K. Kim and B. Ko 'Generating Cartesian NC tool paths for sculptured surface manufacture' *Computer ind. Engng.* Vol 26 N°2.
- [Kim et al.'94] B.H Kim and C N Chu 'Effect of cutter mark on surface roughness and scallop height in sculptured surface machining' *Computer-Aided Design* Vol26 N°3 March 1994.
- [Kim et al.'94] K.I Kim and K. Kim 'A new machine strategy for sculptured surfaces using offset surface' *Inb. J. PROD. RES.* Vol33 N°6 1995.
- [Kim et al.'95] C.B. Kim, S Park and M Y Yang 'Verification of NC tool path and manual and automatic editing of NC code' *International Journal of Production Research* Vol33 N°3 1995
- [Kimmel'93] R. Kimmel and A.M Bruckstein 'Shape offsets via level sets ' *Computer-Aided Design* Vol25 N°3 March 1993
- [Kondo'94] M. Kondo 'Decomposition of complex geometry for manufacturing application' *Computer-Aided Design* Vol 26 N° 3 March 1994
- [Kuragano'92] T. Kuragano 'FRES DAM system for design of aesthetically pleasing free-form objects and generation of collision-free tool-paths' *Computer-Aided Design* Vol24 N°11 November 1992

- [Lagodimos '84] A.G Lagodimos and A.J Scarr 'Interactive computer program for the selection of interference fits' *Computer-Aided Design* Vol6 N'5 November 1984
- [Lakkaraju et al. '92] R.K Lakkaraju, S. Raman and S.A Irani 'A mathematical model for optimization of NC cutting tool-path ' *International Journal of Production Research* Vol30 N'1 1992
- [Lee et al. '92] Y.S Lee, B.K Choi and T.C Chang 'Cut distribution and cutter selection for sculptured surface cavity machining ' *International Journal of Production Research* Vol30 N'6 1992
- [Lee et Chang '91] Y. Lee and T. Chang 'CASCAM- An automated system for sculptured surface cavity machining' *Computers in Industry* 16 1991
- [Leon '91] J.C Leon 'Modélisation et construction de surface pour la CFAO' Hermès Paris 1991.
- [Li et Jerard '94] S.X Li and R.B Jerard '5-axis machining of sculptured surfaces with a flat-end cutter' *Computer-Aided Design* Vol26 N'3 March 1994.
- [Loney et Ozoy '87] G.C Loney and T.M Ozsoy 'Machining of free form surfaces' *Computer-Aided Design* Vol19 N'2 1987.
- [Marciniak '87] K. Marciniak 'Influence of surface shape on admissible tool positions in 5-axis face milling' *Computer-Aided Design* Vol19 N'5 June 1987.
- [Martin et al. '88] R.R Martin and P.C Stephenson 'putting objects into boxes' *Computer-Aided Design* Vol20 N'9, 1988.
- [Marshall et Griffiths '94a] S. Marshall and J.G Griffiths 'A survey of cutter path construction techniques for milling machines' *International Journal of Production Research* Vol23 N'12 1994.
- [Marshall et Griffiths '94b] S. Marshall and J.G Griffiths 'A new cutter path topology for milling machines' *Computer-Aided Design* Vol26 N'3 March 1994.
- [Menq et al. '92] C.H Menq, H.T Yan and G.Y Lai 'Automated precision measurement of surface profile in CAD-directed inspection' *IEEE transaction on robotics and automation*. N°2, Vol8, 1992.
- [Milenkovic '93] V. Milenkovic 'Robust polygon modeling' *Computer-Aided Design* Vol25 N'9, 1993.
- [Oliver et al. '93] J.H Oliver, D.A Wysocki and E.D Goodman 'Gouge detection algorithms for sculptured surface NC generation ' *Journal of Engineering for Industry* Vol115 February 1993.

- [Parabhu et al. '90] P.V Prabhu, A.K Gramopadhye and H Wang ' A general mathematical model for optimizing NC tool-path for face milling of flat convex polygonal surface' International Journal of Production Research Vol28 N°1 1990.
- [Peroche '90] P. Peroche ' La synthèse d'images' Hermès Paris 1990.
- [Peng'84] Q.S Peng 'An algorithm for finding the intersection lines between two B-Spline surfaces' Computer-Aided Design Vol16 N°4 July 1984.
- [Person'78] H. Persson 'NC machining of arbitrarily shaped pockets' Computer Aided Design Vol10 N°3 May1978.
- [Pham '92] B. Pham 'Offset curves and surfaces: a brief survey ' Computer-Aided Design Vol24 N°4 April 1992.
- [Ralph'92] W.L Ralph and M. Loftus 'Cusp geometry analysis in free-form surface machining' International Journal of Production Research Vol30 N°11 1992
- [Rubio '93a] W. Rubio 'Génération de trajectoires du centre de l'outil pour l'usinage de surfaces complexes sur machines à trois et cinq axes' Thèse de doctorat Université Paul Sabatier Toulouse 1993.
- [Scherrer et al. '78] P.K Scherrer and B.M Hillberry 'Determining distance to a surface represented in piecewise fashion with surface paths' Computer-Aided Design Vol10 N°5 September 1978
- [Sederberg'84] W. Sederberg 'Algorithm for algebraic curve intersection' Computer-Aided Design Vol 21 N°9 November 1989
- [Sederberg et al. '84] T.W Sederberg, D.C Anderson and R.N Goldman 'Implicit representation of parametric curves and surfaces' Computer Vision, Graphics, and Image Processing 28 1984
- [Sederberg et al. '84] W. Sederberg, D.C Anderson and R.N Goldman 'Implicitization, inversion, and Intersection of planar rational cubic curves' Computer Vision, Graphics, and Image Processing 31 1984
- [Sederberg et al. '86] T.W Sederberg and S.R Parry 'Comparison of three curve intersection algorithms' Computer-Aided Design Vol18 N°1 January/February 1986.
- [Suh et Lee '90] Y.S Shu and K. Lee 'NC milling tool path generation for arbitrary pockets defined by sculptured surfaces' Computer-Aided Design Vol22 N°5 1990.
- [Sun et Tsai'94] R.H Sun and Y.C Tsai 'A modified analytical model for optimization of NC-tool cutting path ' International Journal of Production Research Vol32 N°10 1994.



- [Sun et Lee '94] Y.S Suh and K. lee 'NC milling tool path generation for arbitrary pockets defined by sculptured surfaces ' Computer-Aided Design Vol26 N°3 March 1994.
- [Surech et al. '94] K. Suresh and D.C H Yang 'Constant scallop-height machining of free-form surfaces' Journal of Engineering for Industry Vol 116 May 1994.
- [Surfcam '97] SURFWARE Inc. 'Manuel de l'utilisateur : SurfCAM R6.1' 1997.
- [Tan et al. '90] S.T Tan, W.S Sze and W.Y Wong 'A survey and classification of cutter path generation methods for three-axis NC milling' Advanced Manufacturing Engineering Vol 2 April 1990.
- [Wonnacott '95] T.H and R.J Wannacott 'Statistiques' Ed. Economica, Paris 1995.
- [Wong '90] D. Wong 'Modelisation de surfaces B-Splines s'appuyant sur des courbes de Bézier ou B-Splines' These de doctorat, ENSAM Paris 1990.
- [You et al. '91] S.J You and K.F Ehman 'Synthesis and generation of surfaces milled by ball nose end mills under tertiary cutter motion' Journal of Engineering for Industry Vol 113 February 1991.
- [Zhu '91] C. Zhu 'Tool-path generation in manufacturing sculptured surfaces with a cylindrical end-milling cutter ' Computer in Industry N°17 1991.
- [Zirmi '98] O. Zirmi 'contribution à l'usinage de poches sous contraintes des paramètres technologiques' mémoire de fin d'études d'ingénieur, labo CFAO INGM 1998.

## *Annexe I*

*Détails de calcul de la hauteur de crête.*

## *Annexe II*

*Organigrammes.*

## *Annexe III*

*Intégration sous AutoCAD V.14.*

## I.1- Détails d'établissement des relations qui lient la hauteur de crête au pas transversal dans le cas de l'usinage 3 axes

### I.1.1. Fraise a bout hémisphérique

La surface étant approximée localement par une calotte sphérique de rayon  $\rho$  et conformément à la construction géométrique de la figure A1.1, les relations qui lient le pas transversal  $l$  à la valeur de la hauteur de crête  $h$  seront établies comme suit :

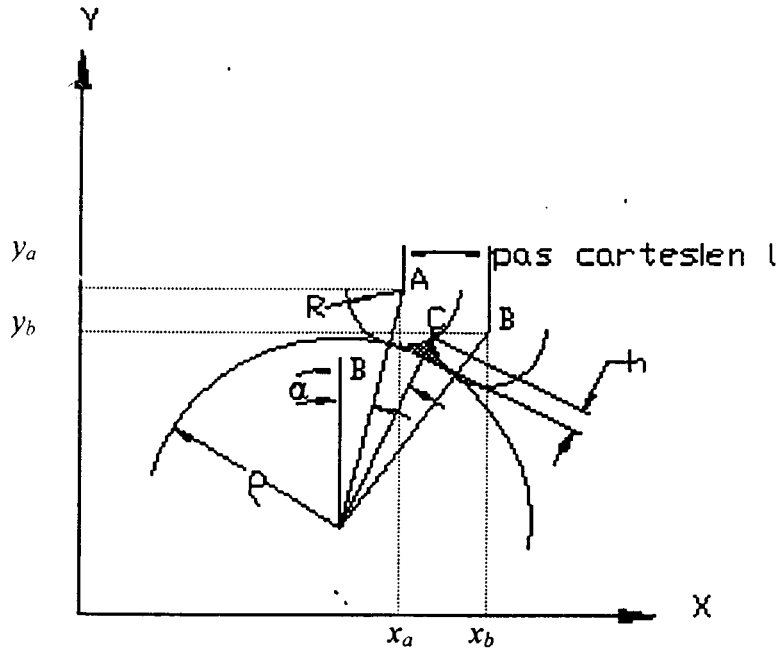


Figure A1.1 Construction géométrique de crête.

Comme nous avons :

$$x_a = (\rho + R) \sin \alpha$$

$$x_b = (\rho + R) \sin(2\beta - \alpha)$$

à partir du triangle  $AOC$  nous pouvons calculer l'angle  $\beta$  à l'aide de la relation :

$$\cos(\alpha - \beta) = \frac{(\rho + h)^2 + (\rho + R)^2 - R^2}{2(\rho + R)(\rho + h)}$$

connaissant l'angle  $\beta$ , nous pouvons calculer la valeur du pas cartésien  $l$  par :

$$l = |x_a - x_b| = |(\rho + R)(\sin(2\beta - \alpha) - \sin(\alpha))|$$

Une autre relation similaire, liant le pas curviligne  $L$ , module du vecteur  $AB$  et la hauteur de crête  $h$  [Kim et al. '88], [Choi et al. '88] est donnée par :

$$L = \frac{|\rho| \sqrt{4(R + \rho)^2 (h + \rho)^2 - (\rho + 2\rho R + (h + \rho)^2)^2}}{(R + \rho)(h + \rho)}$$

Si le rayon de courbure est très grand devant le rayon de l'outil, nous pourrions utiliser les relations suivantes :

$$L = 2\sqrt{h(2R - h)}$$

$$l = 2\sqrt{h(2R - h)} \sin \alpha$$

### I.1.2. Fraise torique

L'établissement des relations qui lient la hauteur de crête au pas cartésien dans les différentes configurations de position d'outils et en adoptants les notations de la figure *AI.1* est réalisé comme suit :

#### a- Première configuration : courbure positive et l'outil descend

*CAS 1* : intersection sur les parties circulaires :

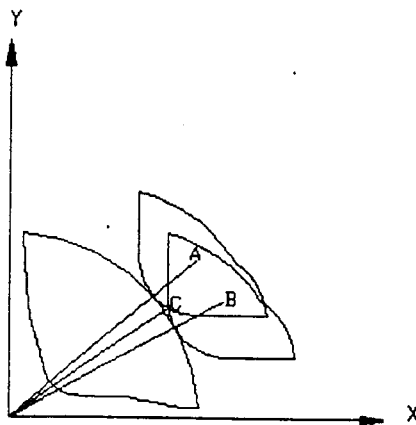


Figure AI.2  
Cas de l'intersection partie  
1 circulaire/ partie 2  
circulaire.

Ce cas est similaire à celui de la fraise à bout sphérique, le rayon du tore étant  $r$  nous aurons :

$$l = |x_a - x_b| = |(\rho + r)(\sin(2\beta - \alpha) - \sin(\alpha))|$$

Avec :

$$\cos(\alpha - \beta) = \frac{(\rho + h)^2 + (\rho + r)^2 - r^2}{2(\rho + r)(\rho + h)}$$

Si le rayon de courbure est infini alors :

$$l = 2\sqrt{h(2r - h)} \sin \alpha$$

CAS 2 : intersection sur la partie circulaire de la première fraise et la partie plane de la deuxième:

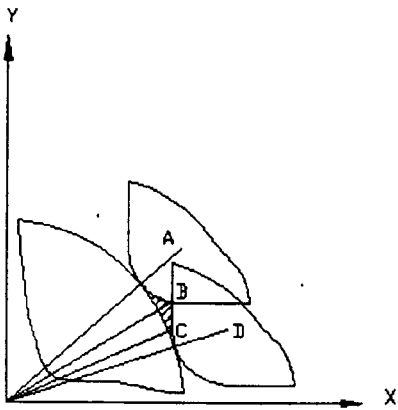


Figure A1.3  
Cas de l'intersection partie  
1 circulaire/partie 2 planè.

Nous avons :

$$x_a = (\rho + r) \sin \alpha$$

$$x_b = (\rho + r) \sin \beta$$

$$x_d = (\rho + h) \sin \beta + r$$

comme :

$$\cos(\alpha - \beta) = \frac{(\rho + h)^2 + (\rho + r)^2 - r^2}{2(\rho + r)(\rho + h)}$$

Donc :

$$l = |x_a - x_d| = |(\rho + r) \sin \alpha - (\rho + h) \sin \beta - r|$$

Si le rayon de courbure est infini alors :

$$l = 2\sqrt{2rh - h^2}$$

**CAS 3** : intersection sur la partie plane de la première fraise et la partie circulaire de la deuxième.

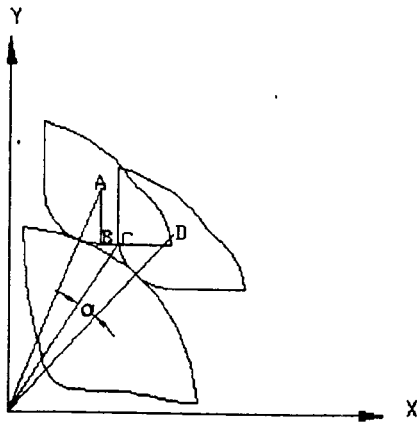


Figure AI.4  
Cas de l'intersection partie  
1 plane/partie 2 circulaire.

Le pas  $l$  est donné par :

$$l = |x_a - x_d|$$

Par ailleurs nous avons :

$$x_a = (\rho + r) \sin \alpha$$

$$y_a = (\rho + r) \cos \alpha$$

$$y_b = (\rho + r) \cos \alpha - r$$

$$y_c = (\rho + h) \cos \beta$$

Comme :

$$y_b = y_c$$

Nous en déduisons :

$$\cos(\beta) = \frac{(\rho + r) \cos \alpha - r}{(\rho + h)}$$

Nous voyons que :

$$x_d = (\rho + r) \sin(\beta + \delta)$$

Donc :

$$l = |x_a - x_d| = |(\rho + r)(\sin \alpha - \sin(\beta + \delta))|$$

Avec :

$$\cos(\delta) = \frac{(\rho + h)^2 + (\rho + R)^2 - R^2}{2(\rho + R)(\rho + h)}$$

Si le rayon de courbure est infini alors :

$$l = (2\sqrt{2rh - h^2} + \operatorname{tg}\alpha(h - r(1 - \sin\alpha)) + r \cos\alpha) \cos\alpha$$

**CAS 4** : intersection sur la partie plane de la première fraise et la partie plane de la deuxième

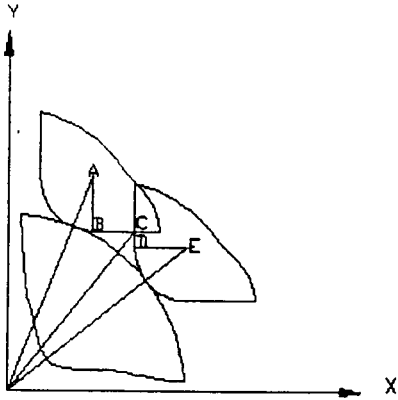


Figure A1.5  
Cas de l'intersection partie  
1 plane/partie 2 plane.

Nous avons :

$$\begin{aligned}x_a &= (\rho + r) \sin\alpha \\y_a &= (\rho + r) \cos\alpha \\y_b &= (\rho + r) \cos\alpha - r \\y_c &= y_b = (\rho + h) \cos\beta\end{aligned}$$

Nous en déduisons :

$$\cos(\beta) = \frac{(\rho + r) \cos\alpha - r}{(\rho + h)}$$

De plus :

$$\begin{aligned}x_c &= (\rho + h) \sin\beta \\x_d &= x_c \\x_e &= x_d + r\end{aligned}$$

D'où la valeur du pas :

$$l = |x_a - x_e| = |(\rho + r) \sin\alpha - (\rho + h) \sin\beta - r|$$

Si le rayon de courbure est infini alors :

$$l = (2\sqrt{2rh - h^2} + \operatorname{tg}\alpha(h - r(1 - \sin\alpha)) + r \cos\alpha) \cos\alpha$$

Pour pouvoir exploiter les relations précédemment établies, il faudra pouvoir distinguer quand ces différents cas ont lieu. Pour cela, on remarque que la hauteur de crête présente deux valeurs particulières correspondant aux limites des parties toriques des deux fraises.

Pour la première fraise, nous avons (figure AI.6) :

$$h_0 = \left| \sqrt{(\rho + r)^2 + r^2 - 2(\rho + r)r \sin \alpha} - \rho \right|$$

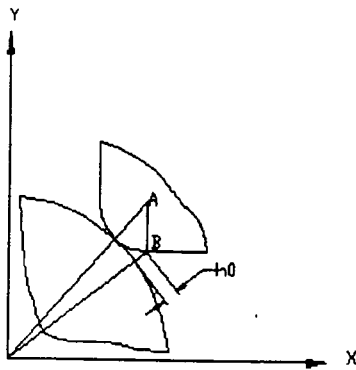


Figure AI.6  
Valeur particulière de la  
hauteur de crête (fraise 1).

Si  $h > h_0$ , alors l'intersection se fait sur la partie plane de la première fraise.

De même pour la deuxième fraise nous avons (figure AI.7) :

$$x_a = (\rho + h_1) \sin \beta = (\rho + r) \sin \gamma - r$$

$$y_a = (\rho + h_1) \cos \beta = (\rho + r) \cos \gamma$$

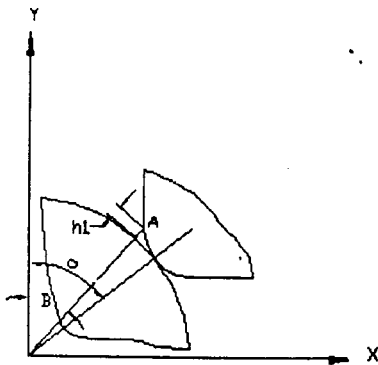


Figure AI.7  
Valeur particulière de la  
hauteur de crête (fraise 2).

En reportant les équations précédentes dans l'équation ci dessous.

$$\cos^2 \gamma + \sin^2 \gamma = 1$$



Nous obtenons donc :

$$h_1 = \left| \sqrt{\rho(\rho + 2r) + r^2 \sin^2 \beta} - \rho - r \sin \beta \right|$$

Si  $h > h_1$ , alors l'intersection se fait sur les parties planes de la fraise.

### b- Deuxième configuration : courbure positive et l'outil monte

De la même manière que pour la première configuration, on établit les relations suivantes :

*CAS 1* : intersection sur les parties circulaires

$$l = |(\rho + r)(\sin(2\beta - \alpha) - \sin(\alpha))|$$

Avec :

$$\cos(\alpha - \beta) = \frac{(\rho + h)^2 + (\rho + r)^2 - r^2}{2(\rho + r)(\rho + h)}$$

*CAS 2* : intersection sur la partie circulaire de la première fraise et la partie plane de la deuxième:

$$l = \left| (\rho + r) \sin \alpha - \sqrt{(\rho + r)^2 - ((\rho + h) \cos \beta + r)^2} \right|$$

Avec :

$$\cos(\alpha - \beta) = \frac{(\rho + h)^2 + (\rho + r)^2 - r^2}{2(\rho + r)(\rho + h)}$$

*CAS 3* : intersection sur la partie plane de la première fraise et la partie circulaire de la deuxième

$$l = |(\rho + r)(\sin(\beta - \delta) - \sin \alpha)|$$

Avec :

$$\cos(\delta) = \frac{(\rho + h)^2 + (\rho + r)^2 - r^2}{2(\rho + r)(\rho + h)}$$

Et :

$$\sin(\beta) = \frac{(\rho + r) \sin \alpha - r}{\rho + h}$$

**CAS 4** : intersection sur la partie plane de la première fraise et la partie plane de la deuxième

$$l = \left| (\rho + r) \sin \alpha - \sqrt{(\rho + r)^2 - ((\rho + h) \cos \beta + r)^2} \right|$$

Avec :

$$\cos(\beta) = \frac{(\rho + r) \sin \alpha - r}{\rho + h}$$

Les valeurs limites  $h_0$  et  $h_1$  sont données par :

Pour la première fraise nous avons :

$$h_0 = \left| \sqrt{(\rho + r)^2 + r^2 - 2(\rho + r)r \sin \alpha} - \rho \right|$$

Si  $h > h_0$ , alors l'intersection se fait sur la partie plane de la première fraise.

Pour la deuxième fraise nous avons :

$$h_1 = \left| \sqrt{\rho(\rho + 2r) + r^2 \sin^2 \beta} - \rho - r \sin \beta \right|$$

Si  $h > h_1$ , alors l'intersection se fait sur les parties planes des fraises.

### c- Troisième configuration : courbure négative et l'outil descend

**CAS 1** : intersection sur les parties circulaires :

$$l = \left| (\rho - r)(\sin(2\beta - \alpha) - \sin \alpha) \right|$$

Avec :

$$\cos(\beta - \alpha) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

**CAS 2** : intersection sur la partie circulaire de la première fraise et la partie plane de la deuxième :

$$l = \left| (\rho - r) \sin \alpha - (\rho - h) \sin \beta - r \right|$$

Avec :

$$\cos(\beta - \alpha) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

**CAS 3** : intersection sur la partie plane de la première fraise et la partie circulaire de la deuxième

$$l = |(\rho - r)(\sin \alpha - \sin(\beta - \delta))|$$

Avec :

$$\cos(\delta) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

Et :

$$\cos(\beta) = \frac{(\rho - r)\cos \alpha - r}{\rho - h}$$

**CAS 4** : intersection sur la partie plane de la première fraise et la partie plane de la deuxième

$$l = |(\rho - r)\sin \alpha - (\rho - h)\sin \beta - r|$$

Avec :

$$\cos(\beta) = \frac{(\rho - r)\cos \alpha - r}{\rho - h}$$

Les valeurs limites  $h_0$  et  $h_1$  sont données par :

Pour la première fraise nous avons :

$$h_0 = \left| \sqrt{(\rho - r)^2 + r^2 - 2(\rho - r)r \cos \alpha} - \rho \right|$$

Si  $h > h_0$ , alors l'intersection se fait sur la partie plane de la première fraise.

Pour la deuxième fraise nous avons :

$$h_1 = \left| \sqrt{\rho(\rho - 2r) + r^2 \sin^2 \beta} - \rho - r \sin \beta \right|$$

Si  $h > h_1$ , alors l'intersection se fait sur les parties planes des fraise.

**d- Quatrième configuration : courbure négative et l'outil monte**

**CAS 1** : intersection sur les parties circulaires :

$$l = |(\rho - r)(\sin(2\beta - \alpha) - \sin \alpha)|$$

Avec :

$$\cos(\beta - \alpha) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

**CAS 2** : intersection sur la partie circulaire de la première fraise et la partie plane de la deuxième

$$l = \left| (\rho - r) \sin \alpha - \sqrt{(\rho - r)^2 - (r - (\rho - h) \cos \beta)^2} \right|$$

Avec :

$$\cos(\beta - \alpha) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

**CAS 3** : intersection sur la partie plane de la première fraise et la partie circulaire de la deuxième

$$l = |(\rho - r)(\sin(\beta + \delta) - \sin \alpha)|$$

Avec :

$$\cos(\delta) = \frac{(\rho - h)^2 + (\rho - r)^2 - r^2}{2(\rho - r)(\rho - h)}$$

Et :

$$\sin(\beta) = \frac{(\rho - r) \sin \alpha + r}{\rho - h}$$

**CAS 4** : intersection sur la partie plane de la première fraise et la partie plane de la deuxième

$$l = \left| (\rho - r) \sin \alpha - \sqrt{(\rho - r)^2 - (r - (\rho - h) \cos \beta)^2} \right|$$

Avec :

$$\sin(\beta) = \frac{(\rho - r) \cos \alpha + r}{\rho - h}$$

Les valeurs limites  $h_0$  et  $h_1$  sont données par :

Pour la première fraise nous avons :

$$h_0 = \left| \sqrt{(\rho - r)^2 + r^2 - 2(\rho - r)r \cos \alpha} - \rho \right|$$

Si  $h > h_0$ , alors l'intersection se fait sur la partie plane de la première fraise.

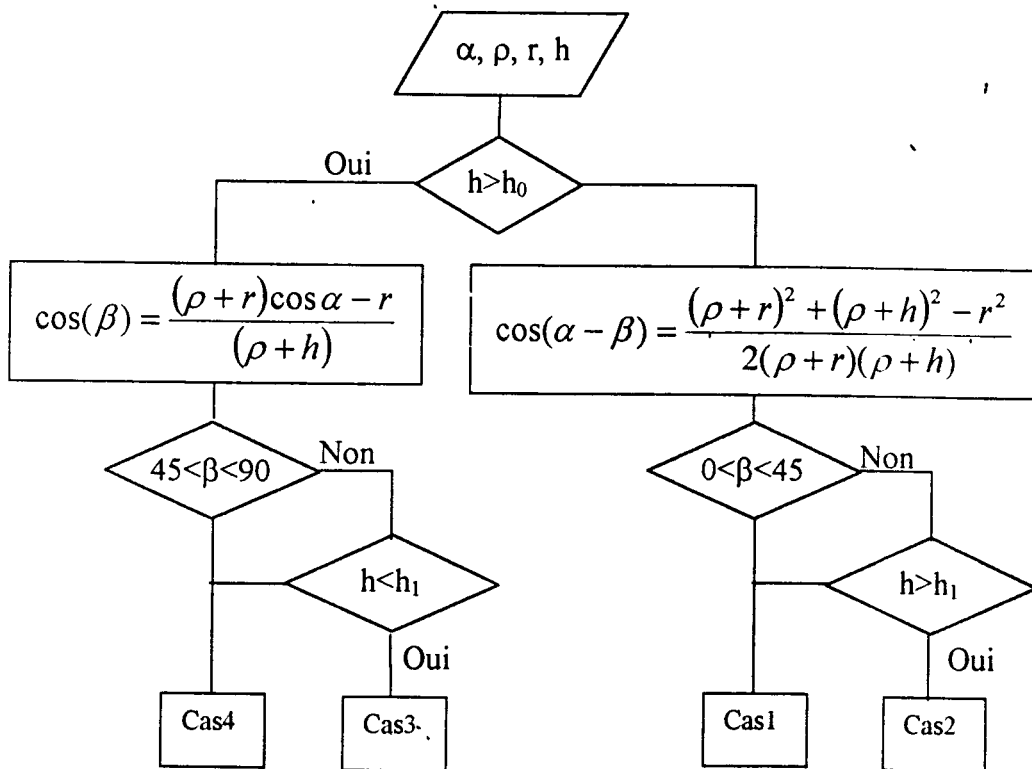
Pour la deuxième fraise nous avons :

$$h_1 = \left| \sqrt{\rho(\rho - 2r) + r^2 \sin^2 \beta} - \rho - r \sin \beta \right|$$

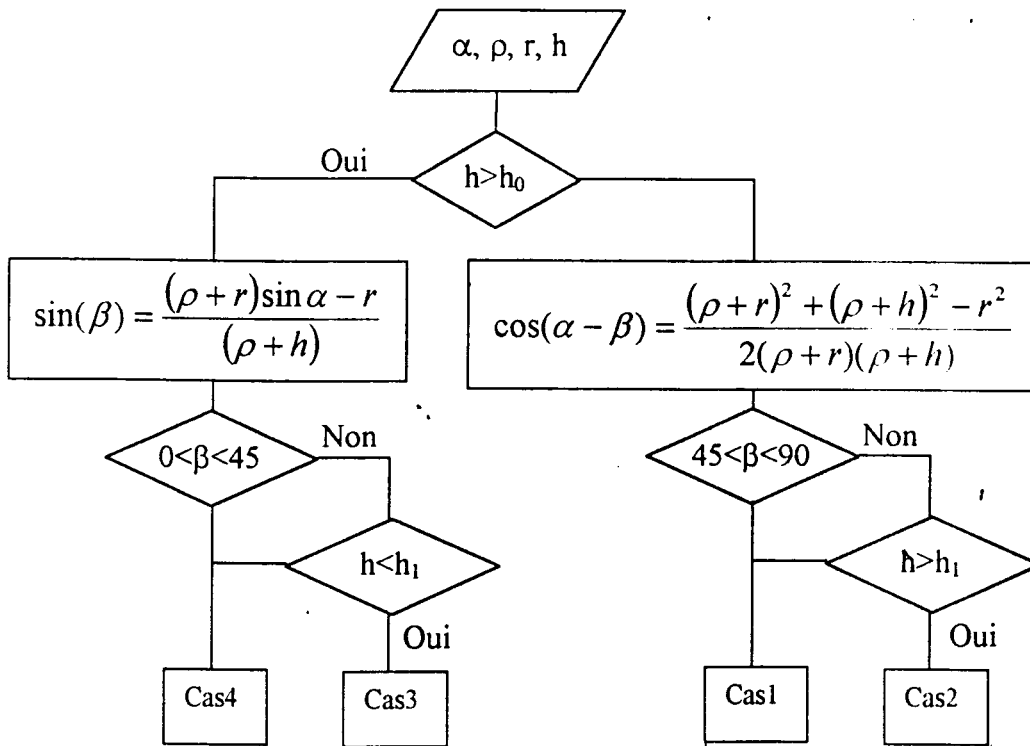
Si  $h > h_1$ , alors l'intersection se fait sur les parties planes de la fraise.

Les organigrammes suivants permettent de déterminer les différents cas de figures possibles précédemment établis.

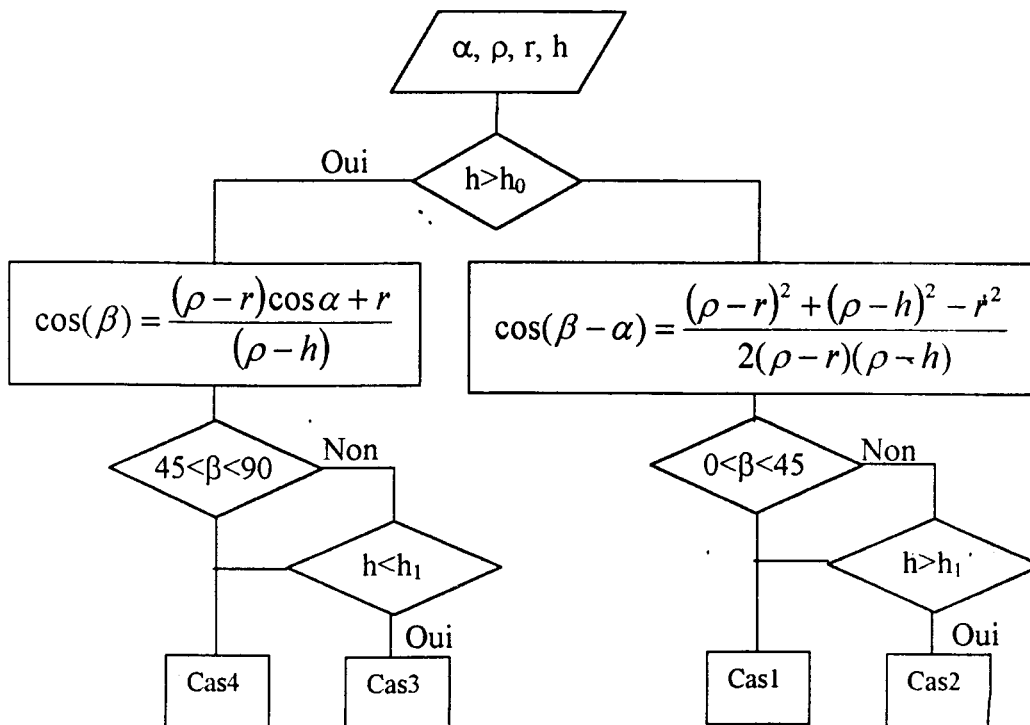
### Organigramme de calcul du pas avec une courbure positive et la fraise qui descend



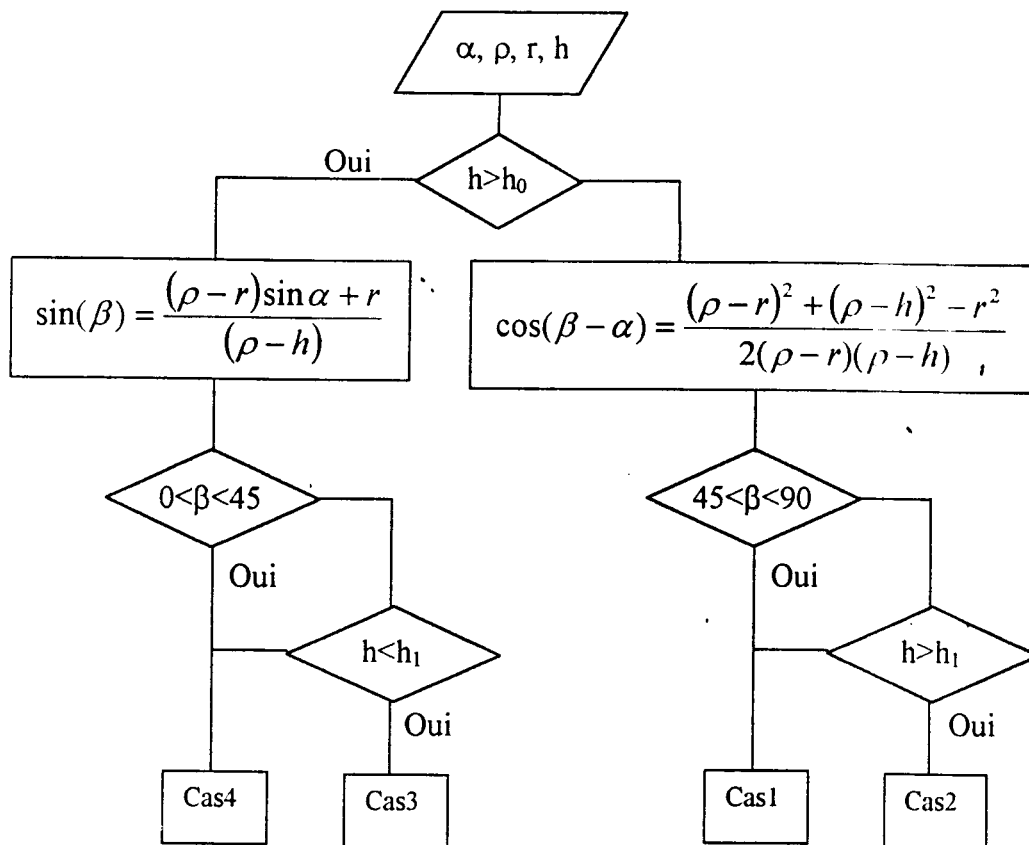
## Organigramme de calcul du pas avec une courbure positive et la fraise qui monte



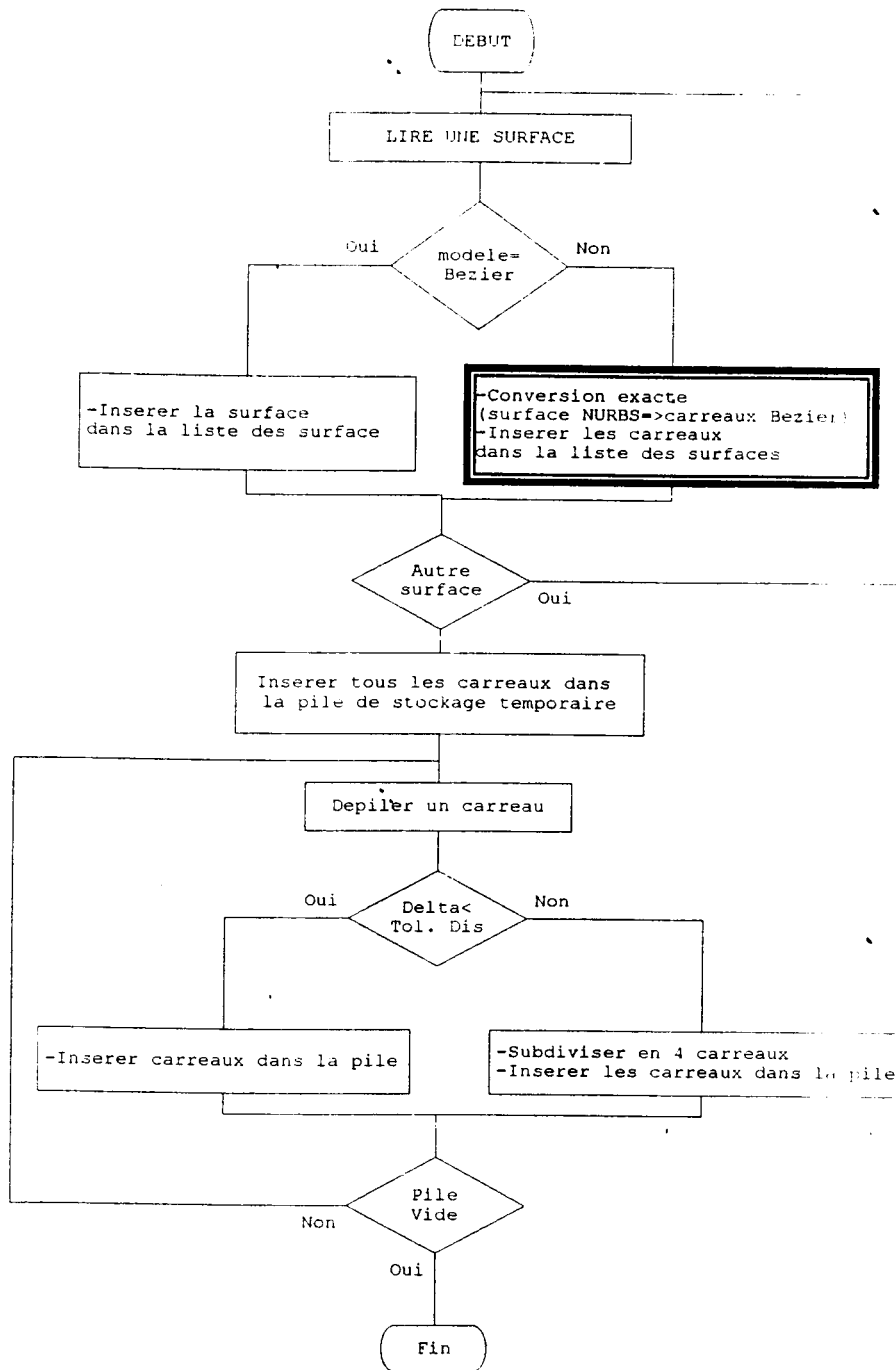
## II.8-Organigramme de calcul du pas avec une courbure négative et la fraise qui descend



## II.9-Organigramme de calcul du pas avec une courbure positive et la fraise qui descend

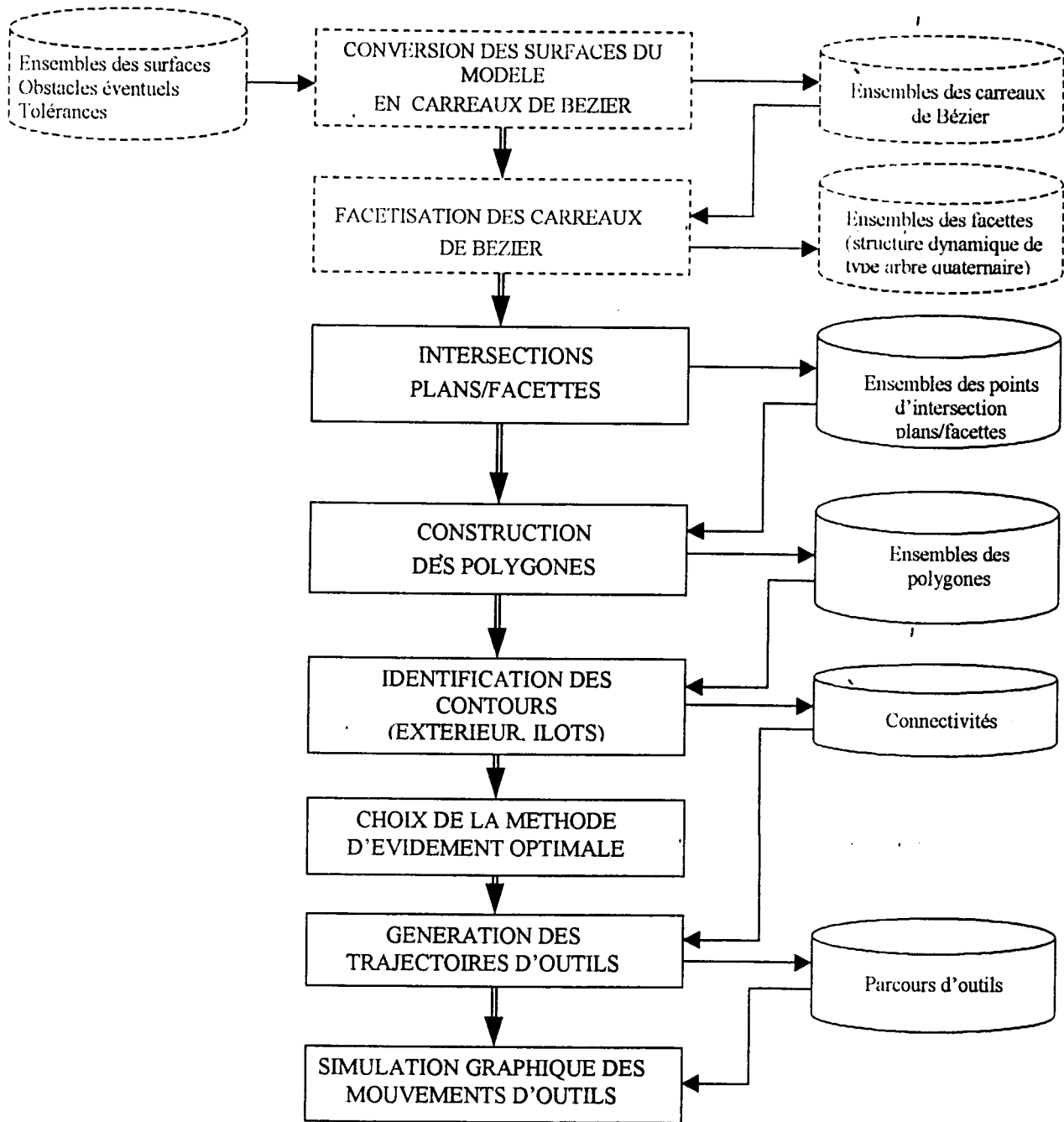


## II.1-Organigramme général du module de conversion/discretisation

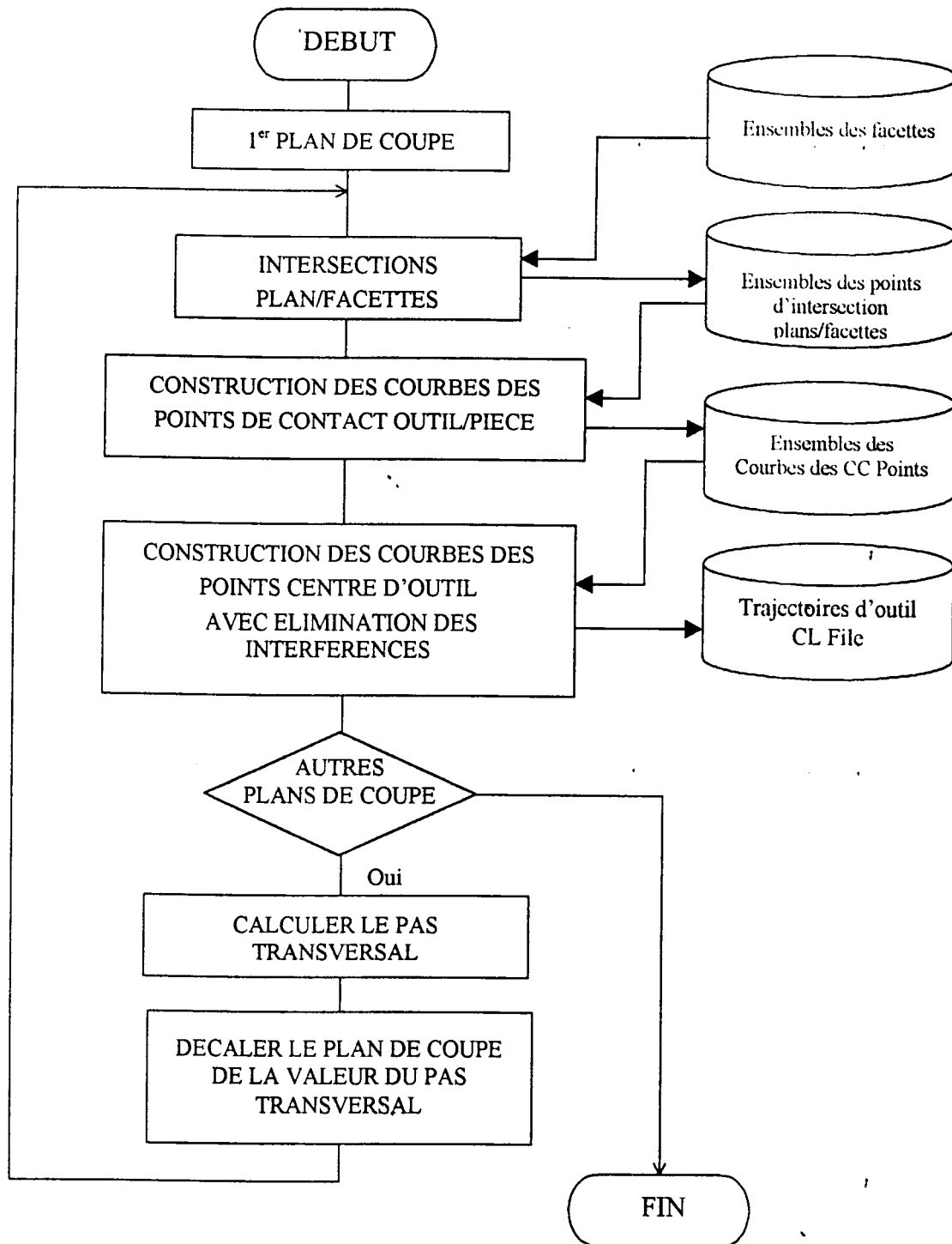




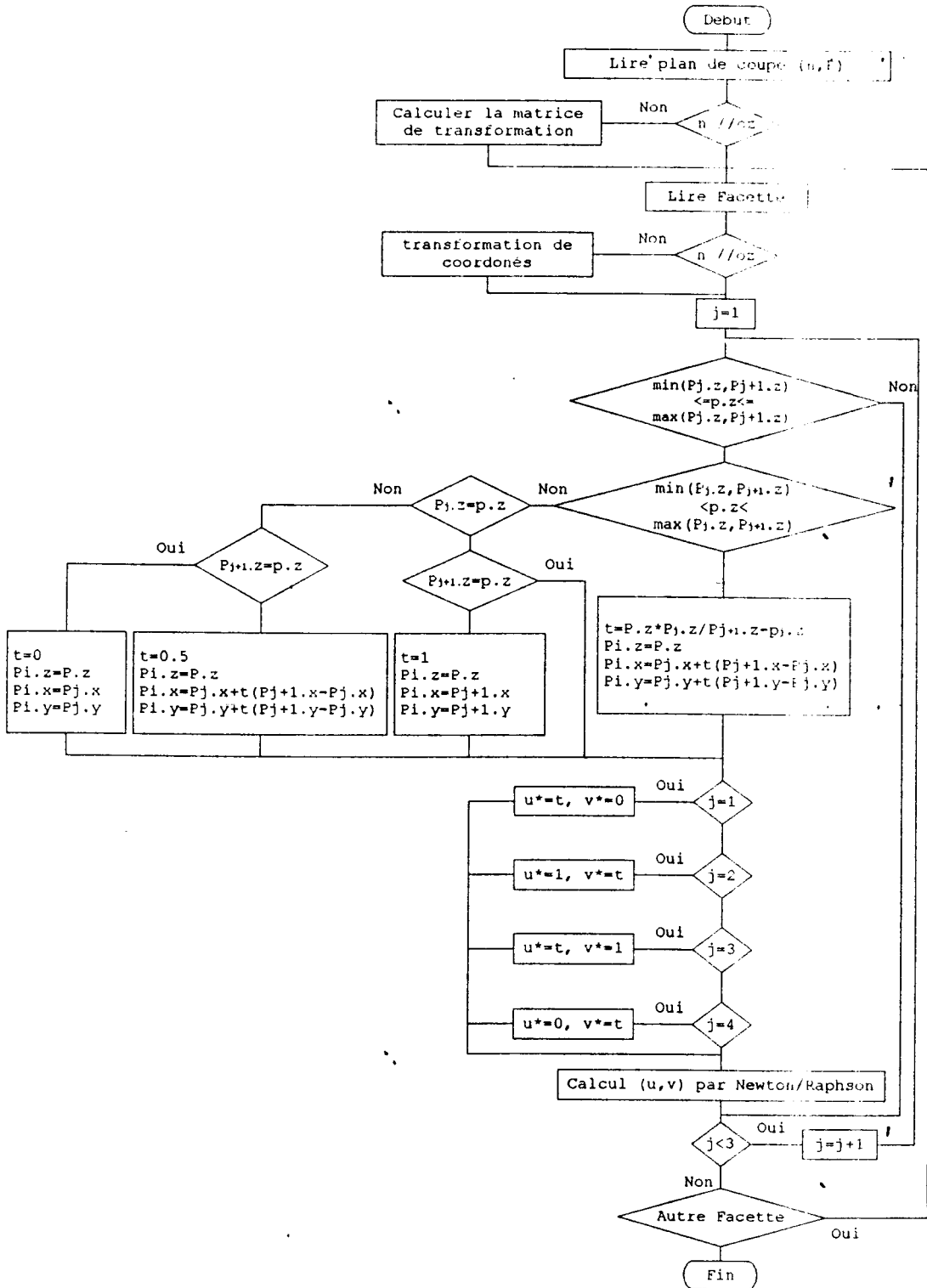
## II.2-Organigramme général du module d'ébauche



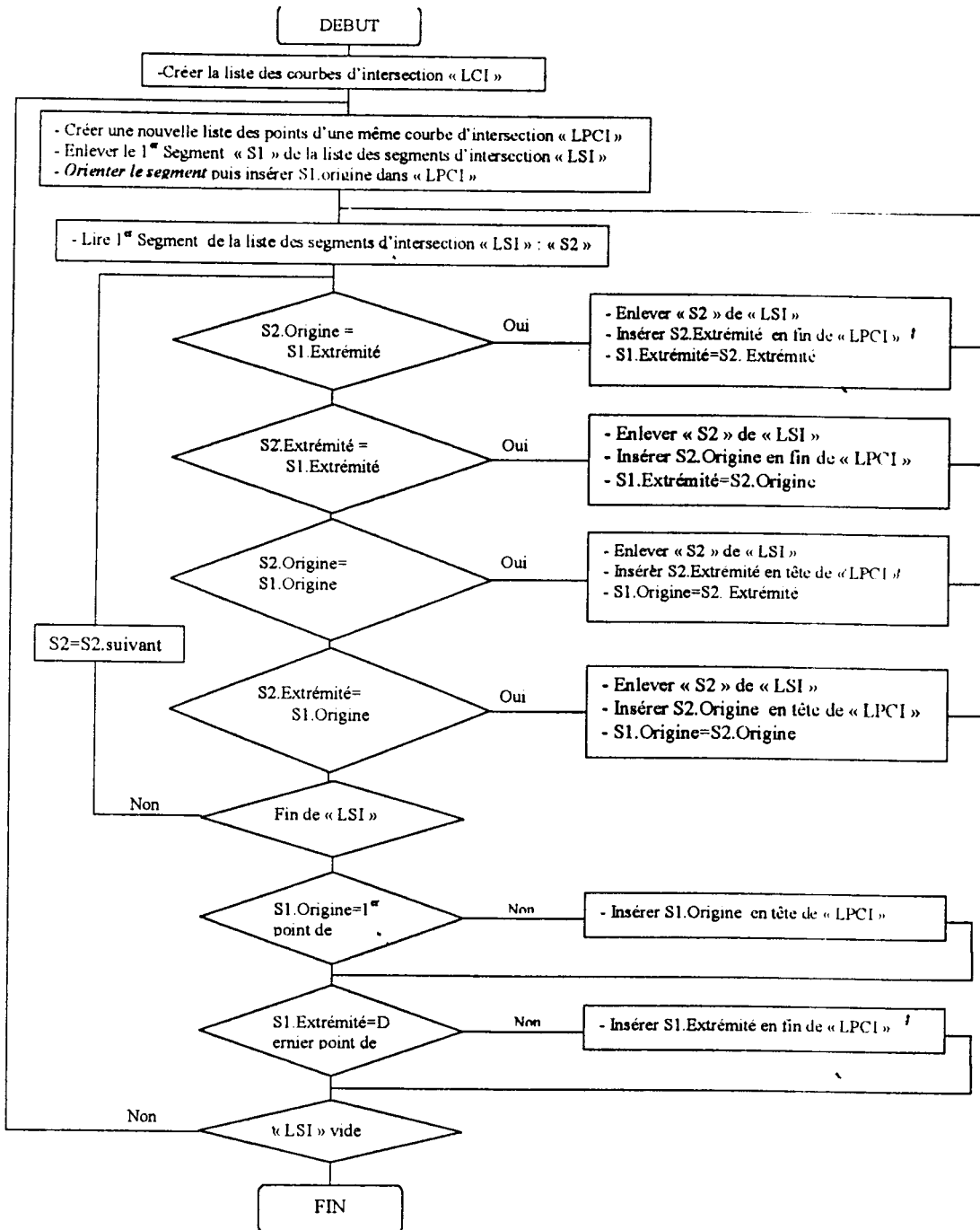
### II.3-Organigramme général du module de finition



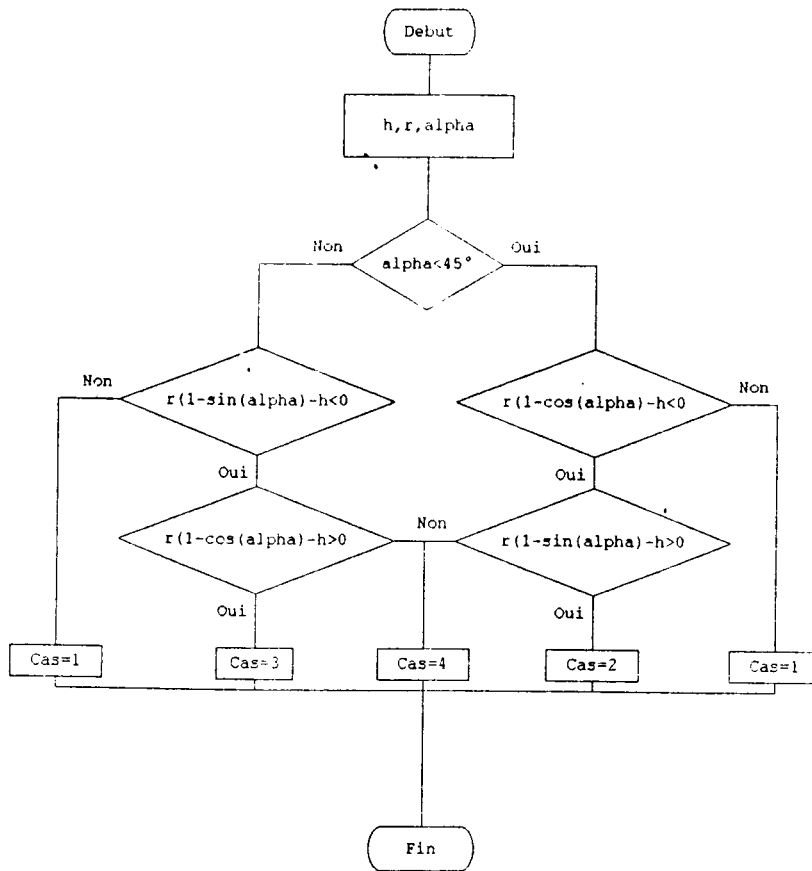
II.4-Organigramme de calcul des segments d'intersections



## II.5-Organigramme de construction et d'orientation des contours d'intersections



II.6- Organigramme de choix des relations de calculs du pas de plongée pour l'usinage du modèle discrétisé avec un outil torique.



### III.1- Menus personnalisés de l'application

La personnalisation des menus d'« AutoCAD » permet une intégration de l'application en toute harmonie avec l'interface utilisateur d'origine (*Figure AIII.1*). La personnalisation se fait par le biais de fichiers menus au format "ASCII" (*Figure AIII.2*) et de règles syntaxiques simples [*Autocad'94b*].

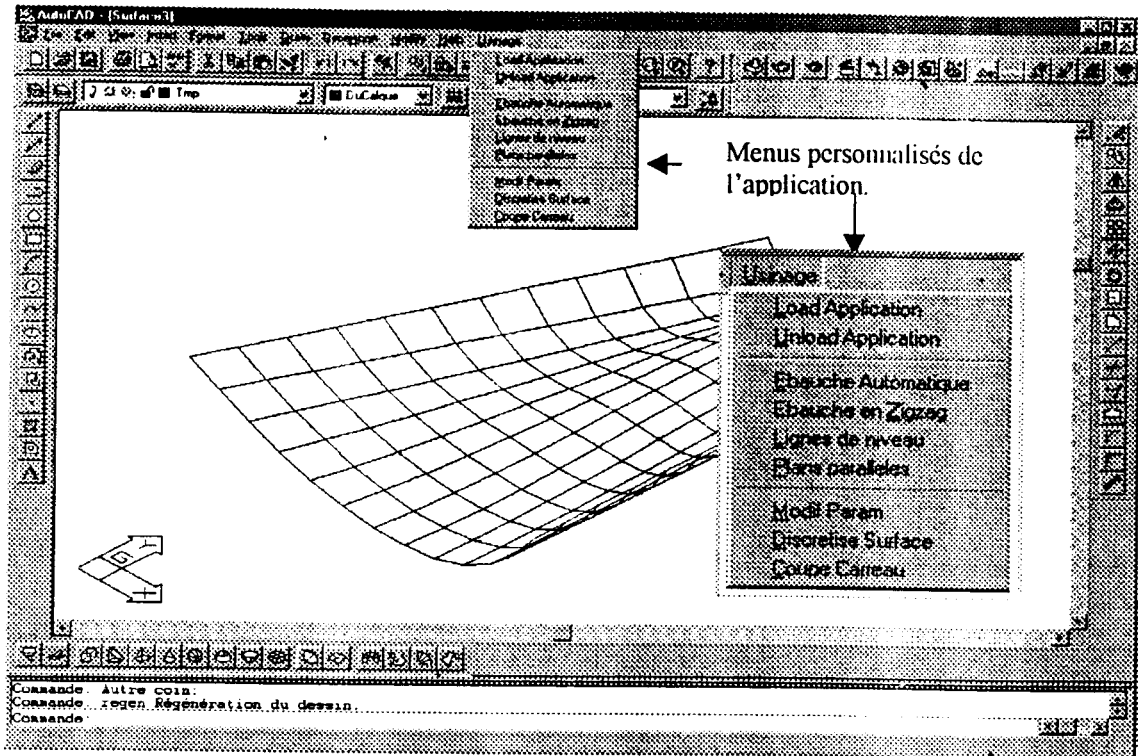


Figure AIII.1 Interface utilisateur après personnalisation des menus d'AutoCAD.

```

***POP11
**USINAGE
ID_MnUsinage  [&Usinage]
ID_Load       [&Load Application] ^C^C(arxload  "usisurf.arx")
ID_Unload     [&Unload Application]^C^C(arxunload "usisurf.arx")
              [--]
ID_Ebauche   [&Ebauche Automatique]ebauche
ID_Ebauche   [Ebauche en &Zigzag]zigzag
ID_TrajCont  [&Lignes de niveau]contournage
ID_TrajFin   [&Plans parallèles]finition
              [--]
ID_ModifParam [&Modif Param]tolerances
ID_Discretise [&Discretise Surface]discretise
ID_CoupeSurf [&Coupe Carreau]coupesurf
  
```

Figure AIII.2 Extrait du fichier menu d'AutoCAD  
'acad.mnu'.

### III.2- Données utilisateur et boîtes de dialogue

La conception des boîtes de dialogues est réalisée grâce à un langage "DCL" spécifique à « AutoCAD » [Autocad'94b]. La figure AIII.3 en donne un extrait relatif à la définition de la boîte de dialogue des paramètres de tolérances ci-dessous (figure AIII.4).

```

/**                               Tolérances                               **/
tolerances : dialog {
  label = "Modification des parametres";
  : edit_box {
    label = "Tolerance de &Linearite";
    key = "gp_tol_lin";
    edit_width = 9;
  }
  : edit_box {
    label = "Tolerance de &Planeite";
    key = "gp_tol_pla";
    edit_width = 9;
  }
  : edit_box {
    label = "Tolerance de &Chainage";
    key = "gp_tol_chain";
    edit_width = 9;
  }
  : edit_box {
    label = "&TCu";
    key = "gp_tcu";
    edit_width = 9;
  }
}
ok_cancel_help;
  errtile;
}
/*****                               Fin tolérances                               *****/

```

Figure AIII.3 Extrait d'un fichier de définition de boîtes de dialogues.

#### a- Discrétisation et tolérances

La boîte de dialogue ci-dessous permet d'introduire et de modifier les valeurs des tolérances de discrétisation, de chaînage et de conception utile.

Paramètre	Valeur
Tolerance de Linearite	0.0250
Tolerance de Planeite	0.0250
Tolerance de Chainage	0.0250
TCu	0.0250

Buttons: OK, Annuler, Aide

Figure AIII.4 Boîte de dialogue pour la modification des valeurs des tolérances.

### b- Usinage d'ébauche et finition

Les paramètres d'usinage sont introduits à partir de boîtes de dialogues organisées comme suit :

- **Paramètres d'outils** : commune à tous les modes d'usinage (Figure AIII.5), permet de spécifier le type et les dimensions d'outil ainsi que le point d'outil piloté.

Informations outil			
Type			Point piloté
<input type="radio"/> Cylindrique	<input checked="" type="radio"/> Hémisphérique	<input type="radio"/> Tortue	<input type="radio"/> Bout <input checked="" type="radio"/> Centre
Dimensions			
Diamètre (mm)	10.0000	Rayon du bout (mm)	5.0000
		Longueur d'outil (mm)	50.0000

Figure AIII.5 Boîte de dialogue des paramètres d'outil.

- **Trajectoires d'évidement en zigzag** : permet de définir les paramètres de l'usinage en zigzag, à savoir le critère de calcul du pas de plongée, le coefficient de recouvrement ainsi que l'activation ou non de l'optimisation des trajectoires (Figure AIII.6).

Trajectoires d'évidement			
type de plongée		Optimisation	
<input type="radio"/> Delta Z maxi	Delta Z maxi (mm)	<input type="checkbox"/> Optim. traject.	
	10.0000		
<input checked="" type="radio"/> P. utile	P. utile (Kw)	Paramètres d'évidement	
	7.5000	Coef. de Recouvrement	0.2500

Figure AIII.6 Boîte de dialogue des paramètres d'usinage en zigzag.

- **Trajectoires de contournage** : permet de définir les paramètres de l'usinage de contournage par lignes de niveau, à savoir le critère de calcul du pas de plongé, la tolérance de conception utile  $T_{cu}$ , la surépaisseur éventuelle, le plan de sécurité, ainsi que l'activation ou non de l'élimination des interférences (Figure AIII.7).



Trajectoires de contourage			
Plan de sécurité			
Plan de Sécurité / Zmax (mm)			50.0000
Type de plongée			
<input checked="" type="radio"/> Delta Z imposé	Delta Z max (mm)		5.0000
<input type="radio"/> ISF	ISF (mm)		2.5000
		Tolérances	
		Tolérance de Surface Totu (mm)	0.0250
		Surépaisseur (mm)	0.0000
		Options	
		<input checked="" type="checkbox"/> Élimin Interférences	
		<input type="checkbox"/> Execution pas à pas	
		<input type="checkbox"/> Contourne	

Figure AIII.7 Boite de dialogue des paramètres d'usinage en zigzag.

- **Trajectoires d'ébauche automatique** : regroupe les trois boites de dialogue précédentes (Figure AIII.8).

* USINAGE D'ÉBAUCHE AUTOMATIQUE *					
Informations outil					
Type			Point pilote		
<input type="radio"/> Cyclindrique	<input checked="" type="radio"/> Hélicoconique	<input type="radio"/> Torique	<input type="radio"/> Bout	<input checked="" type="radio"/> Centre	
Dimensions					
Diamètre (mm)	10.0000	Rayon de l'axe (mm)	0.0000	Longueur d'outil (mm)	50.0000
Trajectoires de contourage					
Plan de sécurité					
Plan de Sécurité / Zmax (mm)			50.0000		
Type de plongée					
<input checked="" type="radio"/> Delta Z imposé	Delta Z max (mm)	5.0000			
<input type="radio"/> ISF	ISF (mm)	2.5000			
			Tolérances		
			Tolérance de Surface Totu (mm)		
			0.0250		
			Surépaisseur (mm)		
			0.0000		
			Options		
			<input checked="" type="checkbox"/> Élimin Interférences		
			<input type="checkbox"/> Execution pas à pas		
			<input type="checkbox"/> Contourne		
Trajectoires d'évidement					
Type de plongée					
<input type="radio"/> Delta Z max	ΔZ max (mm)	10.0000			
<input checked="" type="radio"/> E. utile	Pu (kW)	7.5000			
			Optimisation		
			<input type="checkbox"/> Optim. tra.		
			Paramètres d'évidement		
			Coef de Recouvrement		
			0.2500		
		OK		Annuler	

Figure AIII.8 Boite de dialogue des paramètres d'ébauche automatique.

- *Trajectoires de finition par plan parallèles* : permet de définir les paramètres de l'usinage de finition par plan parallèles, à savoir le critère de calcul du pas transversal, le type de balayage, la tolérance de conception utile  $T_{cu}$ , la tolérance de crête, le plan de sécurité, ainsi que l'activation ou non de l'élimination des interférences (Figure AIII.9).

**USINAGE DE FINITION PAR PLANS PARALLELES**

Informations outil

Type

Cylindrique  Hémisphérique  Lonquin

Point pilote

Bout  Centre

Dimensions

Diamètre (mm)  Rayon de crête (mm)  Longueur d'outil (mm)

Génération des trajectoires d'outil

Plan de sécurité

Plan de Sécurité / Zmax (mm)

Pas transversal

Pas imposé  hauteur de crête max

Balayage

Unidirectionnel  Bidirectionnel

Options

Éliminer interférences

Exécution pas à pas

Contourner

Tolérances

Tolérance de Surface  $T_{cu}$  (mm)

Pas transversal imposé/max (mm)

Tolérance de crête h (mm)

OK Arrêter

Figure AIII.9 Boîte de dialogue des paramètres de finition par plans parallèles.

**III.3- Sortie fichier CL APT**

```
PARTNO/0
PPRINT/USISURF tool library number 22
PPRINT/USISURF tool type number 1
PPRINT/Description: 10mm Ball Rough
PPRINT/Taper angle: 0.000000
PPRINT/Num flutes : 2
PPRINT/Material : 0
CUTTER/10.000000,5.000000,0.000000,5.000000,0.000000,0.000000,0.000000
PPRINT/Offset from surface:0.50000
PPRINT/Programmed to CENTER of tool
PPRINT/USISURF material library number 14
PPRINT/Program units: MILLIMETER
ARCSLP/AUTO
SEQNO/OFF
LOADTL/22,LENGTH,40.00000,0SETNO,22
SPINDL/RPM,477,CLW
COOLNT/FLOOD
MULTAX/OFF
RAPID
GOTO/5.000000,-0.000000,65.000000
GOTO/5.000000,-0.000000,15.000000
.
.
.
GOTO/10.000000,20.000000,10.000000
RAPID
GOTO/10.000000,20.000000,65.000000
END
FINI
```

*Figure AIII.10 Extrait d'un fichier de sortie APT.*