



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE SAAD DAHLAB DE BLIDA
FACULTE DES SCIENCES
DEPARTEMENT D'INFORMATIQUE

CDTA

MEMOIRE DE FIN D'ETUDES

Pour l'obtention

d'un Diplôme de Master en Informatique

Option : Systèmes Informatiques et Réseaux

THÈME :

**Combinaison des Outils Hémisphériques,
Cylindriques et Toriques pour la Finition
des Surfaces Complexes sur des Fraiseuses
à 05-Axes**

Réalisé par :

M^{elle}. BELLEILI Hafsa

M^{elle}. CHARCHAL Latifa

Soutenu devant :

Mr. BEY Mohamed	CDTA,	Encadreur
Mr. BENDIFALLAH Hassène	CDTA,	Encadreur
Mme. TCHANTCHANE Zahida	CDTA,	Encadreur
Mr. KAMECHE Abdallah Hicham	USDB,	Promoteur
Mme. HADJ HENNI Malika	USDB,	Présidente
Mme. DJEDDAR Afrah	USDB,	Examinatrice

2018/2019

Résumé :

Ce travail s'insère dans le cadre de développement d'un environnement de fabrication de surfaces gauches initié par l'équipe « CFAO » du Centre de Développement des Technologies Avancées « CDTA ».

L'objectif de ce travail est le développement d'un module logiciel générique, graphique et interactif permettant la détermination de la combinaison optimale des trois formes d'outils cylindriques, toriques et hémisphériques évitant des interférences et les collisions, lors de la finition des surfaces complexes définies par leurs modèles STL sur des fraiseuses numériques à 05-axes en utilisant la stratégie d'usinage « Z-Constant ».

Mots Clés : Combinaison, Surface Complexe, Outils, Interférence, Collision, Finition, Modèle STL, Fraiseuse 05-axes, Z-Constant.

ملخص :

هذا العمل يندرج في إطار تطوير بيئة انتاج القطع ذات الأسطح المعقدة الذي بدأ على مستوى فريق CFAO التابع لمركز تطوير التكنولوجيات المتقدمة (CDTA) .
الهدف من هذا العمل هو تطوير وحدة برمجية عامة وتفاعلية لتحديد التركيبة الأمثل لأشكال الأدوات الثلاثة "أسطوانى، حلقي، نصف كروي" التي تسمح بتجنب التداخلات والتصادمات، عند الانتهاء من وضع اللمسات الأخيرة للأسطح ذات الأشكال المعقدة، على ماكينات 5 محاور و هذا بالإعتماد على استراتيجية التصنيع "ثابت Z".
الكلمات المفتاحية تركيبة الأشكال، السطح المعقد، الأدوات، التداخل، التصادم، ماكينات 5 محاور، نموذج STL ، "ثابت Z".

Abstract:

This work is a part of the software developed by "CAD/CAM" team of the Center for the Development of Advanced Technologies "CDTA" dedicated to the machining of free form surfaces on CNC milling machines.

The objective of this work is the development of a generic, graphical and interactive software for the determination of the optimal combination of three tools forms cylindrical, toroidal and hemispherical cutters avoiding interferences and collisions for finishing free form surfaces defined by their STL models on 05-axis CNC milling machines using the machining strategy « Z-Constant ».

Keywords: Combination, Complex Surface, Tools, Interference, Collision, Finishing, STL Model, 05-axis Machining, « Z-Constant ».

REMERCIEMENTS

Louanges à Dieu, le Miséricordieux qui nous a permis de terminer ce travail en nous donnant la patience, l'abnégation et le courage pour mener à terme cette étude.

Les travaux présentés dans ce mémoire pour sanctionner la fin des études en Master 2 ont été effectués au CDTA « Centre de Développement des Technologies Avancées », sous la direction de Mr. Bey Mohamed, qu'il trouve ici tous nos remerciements et gratitude pour toute l'attention qu'il nous a portée, par sa présence tout au long du stage, ses orientations et surtout sa patience et son dévouement pour ce résultat.

Nous remercions Mr. Bendifallah Hassène et Mme. Tchantchane Zahida pour leurs collaborations et propositions.

Nous tenons à remercier vivement Mr. Kamèche Abdellah Hicham pour avoir accepté de diriger cette étude, la confiance qu'il nous a accordé et pour ses conseils, son orientation, les diverses corrections et encouragements qu'il nous apporta pendant notre recherche.

Nous tenons à remercier les membres du jury pour l'intérêt accordé à notre travail en l'examinant minutieusement.

Et enfin, nous tenons aussi à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce mémoire.

Merci.

Dédicaces

Je dédie ce modeste travail à :

A la mémoire de ma mère

Qui m'a toujours poussé et motivé dans mes études, j'aurais tant aimé que vous êtes présente. Que Dieu ait vos âmes dans sa sainte miséricorde.

À mon cher père

Aucune dédicace ne saurait exprimer mes respects, ma reconnaissance et mon profond amour. Puisse Dieu vous préserver et vous procurer santé et bonheur.

A ma chère grand-mère

Qui m'a accompagné par ses prières, sa douceur, puisse Dieu lui prêter longue vie et beaucoup de santé et de bonheur.

À mes frères

Mohamed, Maamar, Ismail, Billel: en témoignage de mon affection fraternelle, de ma profonde tendresse et reconnaissance, je vous souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, vous protège et vous garde pour moi.

A ma sœur Meriem

Merci d'être toujours à mes côtés, par votre présence, par votre amour, pour donner du goût et du sens à notre vie de famille. Que ce travail vous témoigne de ma sincère affectation.

A mes nièces et mes neveux

Ibrahim, Romaiassa, Douaa, Nihad, Riyad, Rihab, Ritadj, Adem

Une spéciale dédicace à mon binôme « Charchal Latifa »

Qui a été très patiente avec moi, en témoignage de l'amitié qui nous a unies et des souvenirs de tous les moments que nous avons passé ensemble, je te souhaite un avenir plein de joie, de bonheur de santé et de réussite.

Veillez trouver dans ce travail l'expression de mon grand respect.....

Hafsa

Dédicaces

A mes chers parents

Aucune dédicace ne saurait exprimer l'amour, l'estime, le dévouement et le respect que j'ai toujours pour vous et rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être.

Ce travail est le fruit de vos sacrifices que vous avez consentis le long de ces années.

J'implore Dieu, tout puissant, de vous accorder une bonne santé, une longue vie et beaucoup de bonheur.

A mon frère Djamel

Les mots sont insuffisants pour exprimer ma profonde estime.

Je vous dédie ce travail en témoignage de mon attachement indéfectible.

Que Dieu te protège, t'accorde santé, succès et plein de bonheur dans ta vie.

A ma sœur Sihem

Aucun mot ne peut exprimer mon amour et ma gratitude de t'avoir comme sœur.

Je te souhaite beaucoup de succès, prospérité et une vie pleine de joie et de bonheur que Dieu vous garde pour moi.

A la mémoire de mes grands parents

J'aurais tant aimé que vous soyez présents.

Que Dieu ait vos âmes dans sa sainte miséricorde.

A ma famille

A mes chères tantes, mes oncles et mes cousines.

Votre confiance en moi, vos encouragements, vos prières sont ce qui me pousse toujours à suivre la voie de l'excellence et à réaliser mes rêves.

A ma nièce Yasmine.

Je te souhaite beaucoup de succès, bonheur et que Dieu te bénisse ma petite puce adorée.

A mon binôme Hafsa

On a partagé tous ces moments de stress, de panique, de délire ensemble qui aujourd'hui se transforme en un univers partagé plein de joie.

Je te souhaite un avenir plein de bonheur de santé et de réussite.

Latifa

Table des matières

Introduction Générale	2
Chapitre 1 : Généralités sur le Processus d’Usinage des Surfaces Gauches en 05-Axes	
Introduction	6
1. Processus de production de pièces mécanique	6
2. Etude des surfaces	8
2.1. Définition des surfaces complexes	8
2.1.1. Forme non paramétrique :	8
2.1.2. Forme paramétrique	9
2.2. Types de représentations géométriques	10
2.2.1. Représentations implicites	10
2.2.2. Représentations paramétriques :	10
2.3. Plan tangent	11
2.4. Vecteur normal	12
2.5. Courbure des surfaces	12
2.6. Formes locales	13
2.7. Représentations discrètes	14
2.7.1. Vecteur normal et plan tangent	14
2.7.2. Classification des sommets	16
3. Formats d’échange de données	17
3.1. Modèle STL (Standard Tessellation Language):.....	17
3.2. Structure d’un fichier STL	18
4. Usinages des surfaces gauches	19
4.1. Usinage des pièces complexes	20
4.1.1. Usinage en 03-axes	20
4.1.2. Usinage en 05-axes	21
4.1.3. Phases d’usinage	21
4.2. Stratégies d’usinage en finition	22
4.3. Machine-outil à commande numérique « MONC »	23
4.3.1. Définition des axes pour une fraiseuse numérique à 05-axes	23
4.3.2. Outil de coupe en fraisage.....	24

4.3.3. Différentes parties de l'outil	24
4.3.4. Types d'outils de coupe:	24
4.3.5. Positionnement et orientation de l'outil en 05-axes.....	25
4.4. Problèmes d'interférences	27
4.4.1. Evitement des interférences et des collisions :.....	27
5. Conclusion :	28

Chapitre 2 : Etude Conceptuelle

Introduction.....	30
1. Solution proposée	30
1.1. Récupération des paramètres.....	30
1.2. Création des plans d'usinage	30
1.3. Création des contours d'usinage	32
1.3.1 Détermination des segments d'intersection	33
1.4. Création des cellules de points et des triangles	34
1.5. Enrichissement du modèle STL	35
1.6. Affectation des points aux cellules	36
1.7. Positionnement et outil optimum de chaque point de contact	37
1.8. Détermination des zones d'interférences et de collisions	38
1.9. Tests d'interférences et de de collisions	39
1.10. Correction des interférences et des collisions	42
1.11. Identification de l'outil optimum de chaque point de contact	43
1.12. Affectation de l'outil optimum à chaque contour	45
1.13. Calcul du nombre d'outils à utiliser	47
1.14. Sélection d'une combinaison optimale d'outils	48
1.15. Génération et simulation du trajet d'outils	49
2. Modélisation UML	50
2.1. Diagramme de cas d'utilisation.....	51
2.2. Diagrammes de séquences	51
2.3. Diagramme de classes	56
Conclusion.....	62

Chapitre 3: Implémentation Informatique et Validation

Introduction	64
1. Présentation des langages utilisés	64

1.1. Présentation du langage C++	64
1.2. Présentation d'OpenGL	65
1.3. Présentation d'Embarcadero C++ Builder 10 Seattle	65
2. Présentation de l'environnement de « CFAO »	66
3. Présentation de l'application	67
3.1. Lecture des paramètres STL	67
3.2. Lecture des paramètres des sous régions	68
3.3. Création des passes.....	69
3.4. Création des cellules et enrichissement	70
3.5. Positionnement des outils	71
3.6. Outil optimum.....	73
3.7. Nombre d'outils	74
3.8. Simulation du trajet	75
4. Test et validation :	76
Conclusion.....	87
Conclusion Générale	88
Références bibliographiques	90

Tables des figures

Chapitre 1 : Généralités sur le Processus d'Usinage des Surfaces Gauches en 05-Axes

Figure 1. Forme explicite d'une surface.....	8
Figure 2. Forme implicite d'une surface.....	9
Figure 3. Localisation d'un point sur une surface paramétrique.....	9
Figure 4. Représentation implicite.....	10
Figure 5. Surface paramétrique.....	11
Figure 6. Plan tangent à une surface paramétrée en un point.....	11
Figure 7. Vecteur normal d'une surface paramétrée en point.....	12
Figure 8. Courbures d'une surface.....	13
Figure 9. Différentes formes locales.....	13
Figure 10. Détail d'un maillage.....	14
Figure 11. Normale d'un point.....	15
Figure 12. Plan tangent à un sommet S.....	15
Figure 13. Formes locales d'un sommet.....	17
Figure 14. Représentation CAO, représentation STL et paramètres d'un triangle.....	18
Figure 15. Syntaxe d'un fichier STL.....	18
Figure 16. Processus d'usinage.....	19
Figure 17. Vitesse d'avance et vitesse de coupe.....	20
Figure 18. Usinage en 03-axes.....	21
Figure 19. Usinage en 05-axes.....	21
Figure 20. Phases d'usinage.....	22
Figure 21. Stratégies de finition des surfaces gauches.....	23
Figure 22. Stratégies de finition des surfaces gauches.....	24
Figure 23. Formes des fraises de finition.....	25
Figure 24. Positionnement des différentes formes d'outils.....	26
Figure 25. Orientation de l'outil.....	26
Figure 26. Types d'interférences.....	27

Chapitre 2 : Etude Conceptuelle

Figure 1. Contours de la pièce.....	32
Figure 2. Intersection d'un plan avec un triangle.....	34
Figure 3. Création des cellules.....	35
Figure 4. Enrichissement du modèle STL.....	36
Figure 5. Affectation des points aux cellules.....	36
Figure 6. Outil cylindrique.....	37
Figure 7. Outil torique.....	37
Figure 8. Outil hémisphérique.....	38
Figure 9. Zones d'interférences et de collisions.....	39
Figure 10. Collision avec corps-outil.....	40
Figure 11. Interférence outil cylindrique et outil hémisphérique.....	40
Figure 12. Interférence outil torique.....	42
Figure 13. Correction des interférences et des collisions.....	43
Figure 14. Points d'engagements/dégagement.....	50
Figure 15. Diagramme de cas d'utilisation général.....	51
Figure 16. Création des passes.....	52
Figure 17. Création des contours.....	52
Figure 18. Détermination des sommets et des segments d'intersection.....	53
Figure 19. Positionnement de l'outil.....	54
Figure 20. Identification de l'outil optimum en un point de contact.....	55
Figure 21. Calcul du nombre d'outils pour chaque forme d'outil.....	56
Figure 22. Diagramme de classes.....	57
Figure 23. Classe « Parametre_STL_5_Axes ».....	58
Figure 24. Classe« SOMMET_5_Axes ».....	59
Figure 25. Classe « Triangle_5_Axes ».....	59
Figure 26. Classe « Outil_5_axes ».....	60
Figure 27. Classe « Cellule_5_Axes ».....	60
Figure 28. Classe « Region_5_Axes ».....	61
Figure 29. Classe « Contour_5_Axes ».....	61
Figure 30. Classe « Passe_5_Axes ».....	61

Chapitre 3 : Implémentation Informatique et Validation

Figure 1. Environnement de « CFAO ».	66
Figure 2. Lancement du module logiciel développé.	66
Figure 3. Onglets de l'application développée.	67
Figure 4. Onglet « Lecture des sommets et des triangles ».	68
Figure 5. Onglet« Lecture des paramètres des sous régions».	69
Figure 6. Onglet « création des passes ».	70
Figure 7. Onglet « Cellules et enrichissement ».	71
Figure 8. Onglet « Positionnement des outils ».	72
Figure 9. Onglet « Outil optimum ».	73
Figure 10. Onglet « Nombre d'outils ».	75
Figure 11. Onglet « Simulation du trajet ».	76
Figure 12. Exemple de test.	76
Figure 13. Lecture du fichier de la pièce.	77
Figure 14. Paramètres visibles après la lecture du fichier.	78
Figure 15. Lecture du fichier pièce.	78
Figure 16. Régions de la surface.	78
Figure 17. Visualisation des paramètres de la région « selle de cheval ».	79
Figure 18. Passes d'une région « Selle De Cheval».	80
Figure 19. Visualisation des paramètres d'une région.	80
Figure 20. Contours d'une région.	81
Figure 21. Création des cellules de sommets.	81
Figure 22. Triangles enrichis.	82
Figure 23. Enrichissement du modèle STL.	82
Figure 24. Sommets des cellules.	83
Figure 25. Visualisation du point de contact et des paramètres d'outil.	83
Figure 26. Enveloppe de la partie corps d'outil et cellules de chevauchement.	83
Figure 27. Enveloppe de la partie active d'outil et cellules de chevauchement.	84
Figure 28. Point de contact avec interférences et collisions.	84
Figure 29. Affectation de l'outil optimum au point de contact.	85
Figure 30. Affectation de l'outil optimum a un contour.	85
Figure 31. Simulation du trajet d'outil.	87

Liste des organigrammes

Organigramme 1. Démarche proposée.	31
Organigramme 2. Création des plans d'usinage.....	32
Organigramme 3. Création des contours.	33
Organigramme 4. Identification des zones d'interférences et collisions.....	39
Organigramme 5. Affectation de l'outil optimum à chaque point de contact.	45
Organigramme 6. Outil optimum d'un contour.	46

INTRODUCTION GENERALE

Tout système mécanique est composé d'un ensemble de pièces liées entre elles par des contraintes géométriques et/ou technologiques pour répondre à des fonctionnalités imposées dans les cahiers des charges. Avec l'évolution des moyens de production, en particulier les machines-outils à commande numérique (MOCN), le niveau de qualité de ces pièces a augmenté d'une façon considérable, ce qui impose une attention particulière dans leur fabrication. En raison de leurs formes géométriques très complexes et des problèmes d'accessibilité, ces pièces sont usinées sur des MOCN multiaxes de 03-axes à 05-axes, appelées fraiseuses, par la génération des trajets d'outils en passant par les trois étapes de fabrication, l'ébauchage, la demi-finition et la finition. L'obtention de ces pièces avec les caractéristiques prescrites dans le cahier des charges nécessite parfois l'utilisation successive de différents procédés de fabrication.

Dans l'industrie, certaines pièces ont des formes géométriques très complexes telles que aubes de turbines, hélices des navires, rotors, etc. ne peuvent être usinées sur des machines 03-axes. Cette contrainte exige le passage à l'usinage en 05-axes. De plus, la combinaison des différentes formes d'outils est indispensable pour répondre aux contraintes de fonctionnalité exigées. Lors de l'usinage, à chaque point de contact, une infinité d'orientations de l'axe de l'outil sont possibles et l'outil est toujours tangent à la surface à usiner. En plus des trois degrés de liberté de translation, les deux degrés de liberté supplémentaires (deux rotations) augmentent les possibilités d'orientation de l'outil et d'accès aux différentes régions de la surface, mais en même temps ils augmentent les risques d'interférences et de collisions.

SITUATION DU PROBLEME:

Aujourd'hui, la machine sous toutes ses formes est extrêmement sophistiquée. Elle a évolué pour répondre au développement des techniques de conception des pièces. Ces pièces prennent parfois des formes géométriques de base très simples telles que les cubes, les sphères, etc., mais d'autres fois, les formes deviennent très complexes pour des raisons de performance ou d'apparence. Le plus souvent, ces formes complexes ne sont pas explicitées

par des formules mathématiques standards, d'où leurs nom de surfaces libres (surfaces gauches). C'est la raison pour laquelle, des modèles géométriques ont été développés pour faciliter la conception de ces pièces. Avec l'intégration de l'outil informatique, un nombre important de logiciels de Conception Assistée par Ordinateur « CAO » sont développés et commercialisés pour aider les concepteurs dans leurs tâches de conception. Une fois la pièce conçue, il est possible d'échanger les informations contenues dans le modèle avec d'autres systèmes de conception en utilisant des standards d'échange de données tels que STL, STEP, IGES, DWG, etc. Dans certains cas où la géométrie de la pièce est tellement complexe que sa conception dans un logiciel de CAO devienne couteuse ou impossible, le processus du Reverse Engineering est utilisé. Dans ce cas, une maquette (prototype) est réalisée manuellement puis scannée par un dispositif spécifique pour récupérer un nuage de points très dense. Ce dernier est traité pour le convertir en un modèle triangulé (modèle STL).

La mise en production de ces pièces nécessite le recours à une machines spéciale dont sa caractéristique principale est sa commande numérique intégrée. Elle permet d'automatiser les trois phases du processus d'usinage à savoir l'ébauchage, la demi-finition et la finition. La génération des trajets d'outils associés à chaque opération exige la prise en compte de plusieurs paramètres tels que les modèles des surfaces (continus, discrets), les outils d'usinage (formes et dimensions), la cinématique de la machine (type et nombre d'axes), les stratégies d'usinage en fonction de l'opération d'usinage considérée, les précisions exigées (hauteur de crête, erreur de flèche, etc.), les problèmes d'usinage (interférences, collisions), etc. Le nombre important des paramètres à considérer et leurs interdépendances rendent la tâche de génération d'un trajet d'outils sains et optimums très complexe.

Pendant l'usinage, l'outil suit une trajectoire complexe dans l'espace 3D. De ce fait, des problèmes d'interférences, entre la partie active de l'outil et la surface à usiner, ainsi que des problèmes de collisions, entre les différentes parties de l'environnement d'usinage (machine, outils, pièce, etc.), peuvent surgir et donc altérer la qualité des surfaces usinées. Pour cela, il est impératif d'éliminer ces problèmes avant l'exécution effective de l'usinage. Lors de l'usinage en 5-axes, en chaque point de contact l'outil est tangent à la surface à usiner, possède une infinité d'orientations de son axe. Les deux rotations supplémentaires augmentent les possibilités d'orientation et d'accès de l'outil aux différentes régions de la surface, mais en même temps augmentent considérablement les risques d'interférences et de collisions. Parmi les orientations possibles, il est indispensable de choisir l'orientation la plus optimale évitant ces problèmes et optimisant l'usinage.

Le présent projet s'inscrit dans le cadre du programme triennal 2018-2020 de recherche de l'équipe Conception et Fabrication Assistées par Ordinateur « CFAO », de la Division Productique et Robotique « DPR » du Centre de Développement des Technologies Avancées « CDTA », intitulé « Production des Surfaces de Formes Libres sur des Fraiseuses Numériques à 05-axes ». L'objectif de ce projet est la proposition de solutions logicielles permettant l'automatisation et l'optimisation du processus de production des pièces de formes quelconques sur des fraiseuses numériques à 05-axes.

OBJECTIF DU TRAVAIL :

Dans ce projet, nous nous intéressons à la combinaison des trois formes d'outils « Cylindriques, Hémisphériques et Toriques » lors de la finition des surfaces complexes, définies par leurs modèles « STL », sur des fraiseuses numériques à 05-axes en considérant les formes locales des surfaces et les temps d'usinage. L'objectif étant la détermination des orientations optimales des outils affectés à des portions de surfaces appelées régions (formes et dimensions) afin de déterminer les points de passage permettant d'éliminer les interférences et les collisions. Il s'agit de concevoir, de développer et d'intégrer à l'environnement de production des surfaces complexes développé par l'équipe « CFAO » du « CDTA », un module logiciel générique, graphique et interactif pour la détermination de la combinaison optimale des trois formes d'outils permettant l'évitement des interférences et des collisions sous certaines contraintes géométriques et technologiques.

STRUCTURATION DU MEMOIRE :

Le présent mémoire est composé des chapitres suivants :

- Le premier chapitre est consacré à l'étude du processus d'usinage des surfaces complexes.
- L'étude conceptuelle de notre application logicielle, la démarche de résolution et les solutions proposées sont menées dans le deuxième chapitre.
- Le dernier chapitre est réservé à la phase de tests et de validations des résultats.

CHAPITRE 1

GENERALITES SUR LE PROCESSUS D'USINAGE DES SURFACES GAUCHES EN 05-AXES

Introduction.

1. Processus de production des pièces mécaniques.
2. Etude des surfaces.
3. Formats d'échanges de données.
4. Usinage des surfaces gauches.

Conclusion.

Introduction :

Le monde de la conception et de la fabrication mécanique est très vaste et commence par la plus petite pièce utilisée par un chirurgien-dentiste à la plus grande pièce d'un navire. Chaque système mécanique est composé de pièces où chacune est composée de surfaces avec des caractéristiques géométriques et mécaniques spécifiques pour répondre aux exigences techniques et aux conditions de fonctionnement.

La surface d'une pièce (objet solide) est la partie de ce solide qui le limite dans l'espace et le sépare du milieu environnant. Elle assure des fonctions appropriées telles que l'aptitude aux frottements et la résistance aux contraintes mécaniques. Parmi les surfaces les plus répandues dans le domaine de l'industrie manufacturière les surfaces complexes appelées généralement « surfaces de formes libres » ou « surfaces gauches ». Ces surfaces sont très utilisées dans de nombreuses applications industrielles telles que les pièces d'automobiles, les coques de navires, les pièces en aérospace, etc. Elles sont caractérisées par une topologie complexe composée d'un ensemble de régions de surfaces concaves, convexes ou à double courbure (selle de cheval). Le processus de réalisation de ces surfaces est un processus industriel qui a subi de grandes évolutions ces dernières décennies par l'introduction de la « CFAO » « Conception et Fabrication Assistées par Ordinateur » et de la commande numérique dans les machines de production. Ce processus est formé d'une suite d'activités qui passe par plusieurs étapes successives et complémentaires à savoir la conception, la simulation, la production et le contrôle de la pièce réalisée.

La suite de ce chapitre est organisée en trois parties. Dans la première partie, une présentation du processus de production des pièces mécaniques. La deuxième partie est consacrée à l'étude des méthodes de représentation et de conception des surfaces gauches. La dernière partie est réservée à la présentation du processus de production des surfaces gauches en 05-axes et les moyens utilisés.

1. Processus de production de pièces mécanique :

Le processus de production des pièces mécaniques est né des phénomènes suivants :

- **Besoin** : la nécessité est mère de l'invention grâce au besoin. Il y'a la naissance de l'industrie [2].
- **Naissance de l'idée** : c'est l'analyse des besoins. Elle donne naissance à une idée qui conduit à la définition des solutions [2].

- **Conception « CAO » :**
 - C'est l'étude, la recherche et l'élaboration du cahier des charges.
 - Dans cette étape, les solutions trouvées sont conçues et schématisées en respectant les critères imposés.
 - Choix de la matière qui convient pour fabriquer la pièce désirée.

La performance des solutions dépend de l'efficacité et de la puissance des moyens utilisés. Ainsi, le recours à des méthodes de conception très puissantes telles que la « CAO » (Conception Assistée par Ordinateur) qui rassemble des outils informatiques pour réaliser une modélisation d'un objet géométrique en 3D. Parmi ses avantages :

- Accroître la productivité et améliorer la qualité de la conception.
- Valider par simulation des solutions adoptées avant leurs réalisations.

Le modèle « CAO » peut être ensuite exporté vers des logiciels de « FAO » (Fabrication Assistée par Ordinateur) afin de créer les programmes de pilotage des machines-outils à commande numérique [3].

- **Fabrication « FAO » :**

La Fabrication Assistée par Ordinateur « FAO » est le maillon intermédiaire de la chaîne numérique dans la production de pièces. Le but de la « FAO » est la génération du fichier contenant le programme de pilotage d'une machine-outil à commande numérique « MOCN ». Ce fichier décrit avec grande précision les mouvements que doit exécuter les éléments mobiles de la machine-outil en particulier les axes pour usiner une pièce donnée.

Dans un système homogène où les modules de « CAO » et de « FAO » sont complètement intégrés dans un seul environnement, les informations circulent facilement et directement entre les deux modules. Dans le cas où ces deux modules ne sont pas intégrés dans le même environnement, la circulation des données nécessite l'utilisation des translateurs (formats d'échange de données). Plusieurs formats d'échange de données sont utilisés tels que IGES, STEP, STL, DWG etc. Le choix d'un des formats dépend du besoin, des contraintes de l'application et du choix de l'utilisation [3].

- **Choix de l'outil :**

Le choix de l'outil comprend la forme, les dimensions et le matériau. Ainsi, en fonction des résultats escomptés, la pièce est usinée en fonction de [2] :

- ✓ La matière à traiter.

- ✓ Niveau de complexité de la pièce.

- **Usinage :**

L'usinage définit une famille de techniques de fabrication de pièces mécaniques entre autres le fraisage, l'alésage, le mortaisage, le perçage, le pliage, etc. L'usinage de pièces mécaniques s'effectue sur des machines à 3-axes pour les travaux sur surfaces planes, mais cède progressivement le chemin à l'usinage en 5-axes pour des surfaces à profil complexe [1]. Le processus d'usinage de ces surfaces passe par trois opérations :

- **Ebauchage** : pour enlever le maximum de matière.
- **Demi-finition** : pour s'approcher de la forme finale.
- **Finition** : pour obtenir la forme finale avec un bon état de surface.

2. Etude des surfaces :

2.1. Définition des surfaces complexes :

Les surfaces complexes, appelées aussi surfaces gauches, sont caractérisées par une topologie complexe composée d'un ensemble de surfaces concaves, convexes ou à double courbures. Les surfaces sont classées en deux grandes catégories.

2.1.1. Forme non paramétrique :

Les surfaces non-paramétriques sont représentées sous deux formes différentes [4] :

- **Forme explicite** : elle est définie par une fonction à deux (2) variables :

$$Z = F(X, Y) \quad (1)$$

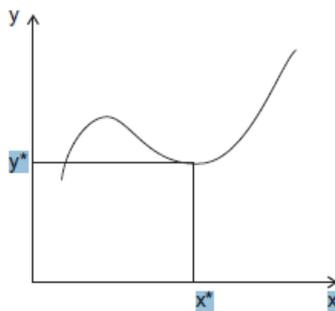


Figure 1. Forme explicite d'une surface.

Cette formulation ne permet pas de décrire des surfaces fermées ou qui se croisent plusieurs fois. Cette formulation ne satisfait pas les conditions de diversité.

- **Forme implicite** : ces formes sont définies par une fonction de trois variables :

$$F(X, Y, Z)=0 \quad (2)$$

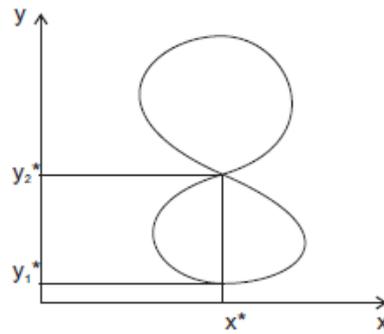


Figure 2. Forme implicite d'une surface.

La forme implicite ne permet pas de représenter toutes les surfaces, mais elle n'a pas la limitation de la représentation explicite.

2.1.2. Forme paramétrique :

Les surfaces paramétriques sont très utilisées dans la modélisation de surface. Elles sont définies par un ensemble de trois fonctions réelles $X()$, $Y()$ et $Z()$.

Ces fonction dépendent de deux paramètres u et v qui n'ont aucune signification physique.

Une surface paramétrique est donnée par la formulation suivante

$$F(u,v)=(X(u,v),Y(u,v),Z(u,v)) \quad (3)$$

Ces fonctions, dépendent de deux (02) paramètres u et v appartenant à l'intervalle $[0,1]$. Ainsi, chaque point $F(u, v)$ de la surface paramétrique, appartient à un carré dans le plan paramétrique qui est défini par les sommets $(0, 0)$, $(0, 1)$, $(1, 0)$ et $(1, 1)$ [3].

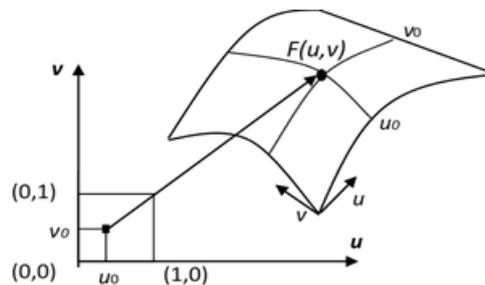


Figure 3. Localisation d'un point sur une surface paramétrique.

2.2. Types de représentations géométriques :

Trois méthodes de représentation de surfaces gauches sont à distinguer: [4]

- Représentations implicites.
- Représentations paramétriques.
- Représentations discrètes.

2.2.1. Représentations implicites:

Cette représentation repose sur la définition d'un champ de potentiel engendrée par une fonction définie en tout point de l'espace R^3 . Habituellement, la surface est considérée comme étant l'ensemble des points de potentiel nul, tandis que l'intérieur correspond aux potentiels négatifs. Les propriétés différentielles de la fonction utilisée permettent en outre de définir les normales à la surface de l'objet représenté.

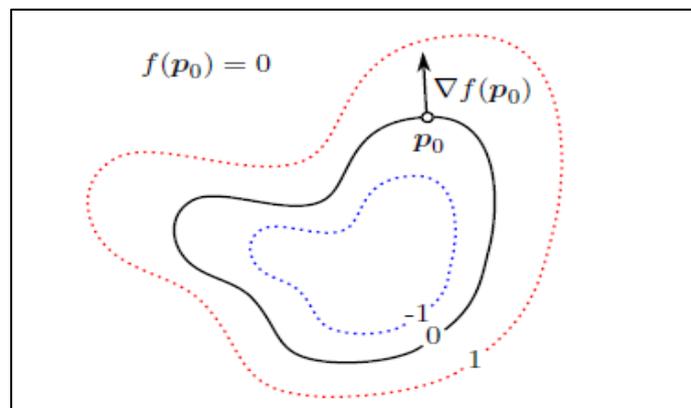


Figure 4. Représentation implicite.

2.2.2. Représentations paramétriques :

Une surface paramétrique est une surface dans l'espace euclidien R^3 définie par une fonction paramétrique avec deux paramètres.

$$F: D \subset R^2 \rightarrow R^3 \quad (4)$$

Elle s'écrit :

$$(u, v) \in D \rightarrow F(u, v) = \begin{cases} X(u, v) \\ Y(u, v) \\ Z(u, v) \end{cases} \quad (5)$$

Les réels u et v sont des paramètres qui appartiennent à l'intervalle $[0, 1]$.

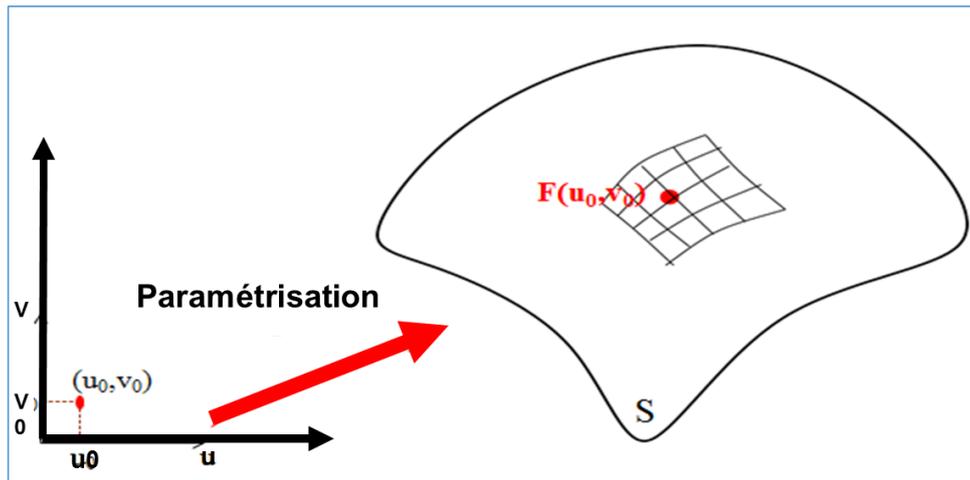


Figure 5. Surface paramétrique.

2.3. Plan tangent :

C'est le vecteur dont les coordonnées sont les dérivées partielles de F par rapport à \mathbf{u} en considérant \mathbf{v} constant et vis-versa.

$$F(u, v) = (X(u, v), Y(u, v), Z(u, v)) \quad (6)$$

Les dérivées suivant les directions u et v sont données par :

$$F'_u(u, v) = \left(\frac{\delta}{\delta u} X(u, v), \frac{\delta}{\delta u} Y(u, v), \frac{\delta}{\delta u} Z(u, v) \right) \quad (7)$$

$$F'_v(u, v) = \left(\frac{\delta}{\delta v} X(u, v), \frac{\delta}{\delta v} Y(u, v), \frac{\delta}{\delta v} Z(u, v) \right) \quad (8)$$

Ces vecteurs sont tangents à la surface en un point M_0 . Le plan défini par M_0 et les vecteurs $F'_u(u_0, v_0)$ et $F'_v(u_0, v_0)$ est appelé plan tangent à la surface en M_0 [5].

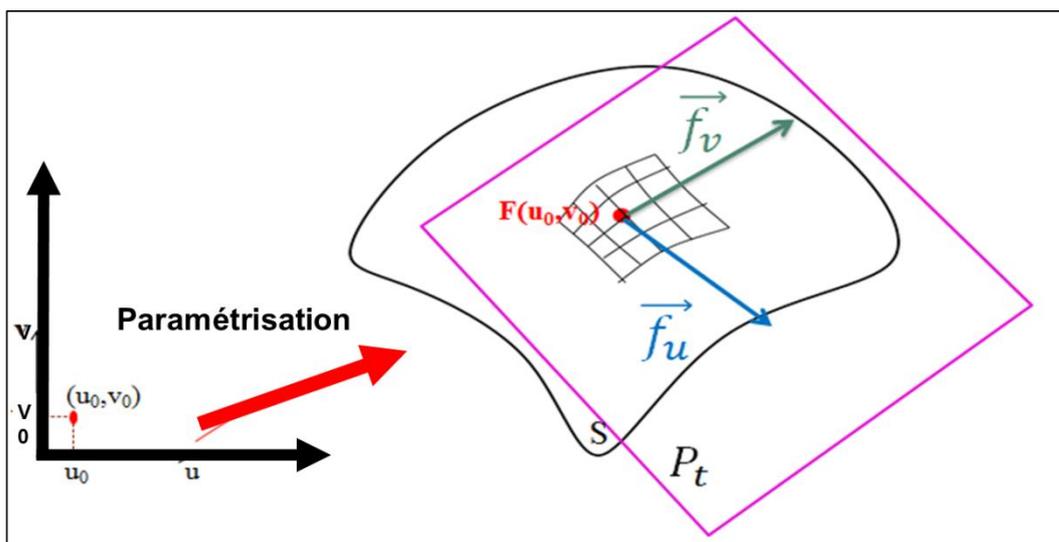


Figure 6. Plan tangent à une surface paramétrée en un point.

2.4. Vecteur normal :

Le vecteur normal d'une surface paramétrée en point est le vecteur non nul orthogonal au plan tangent en ce point. Ce vecteur est obtenu de deux façons différentes :

- Le produit vectoriel de deux vecteurs directeurs non colinéaires du plan tangent.
- A partir d'une équation cartésienne du plan. Si l'équation du plan est $ax+by+cz=d$, alors un vecteur normal du plan est le vecteur de composantes (a, b, c) [6].

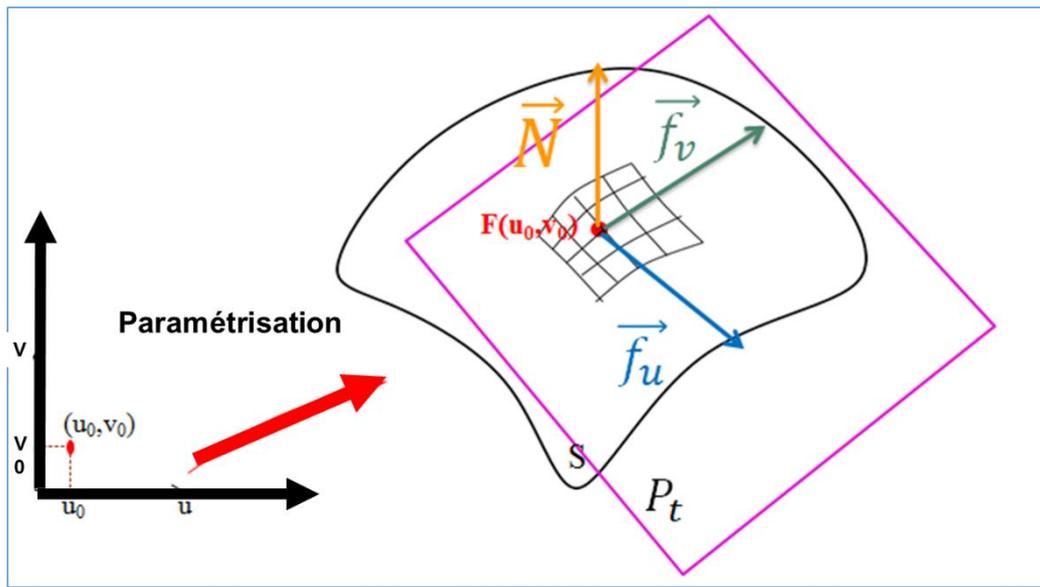


Figure 7. Vecteur normal d'une surface paramétrée en point.

2.5. Courbure des surfaces :

La courbure mesure la manière dont une courbe ou un arc géométrique s'éloigne localement d'une ligne droite. En n'importe quel point sur une surface paramétrique, il y a un nombre infini de courbes ayant chacune une valeur différente. Parmi ces courbures, il existe deux courbures principales appelées courbures maximum et minimum notées k_1 et k_2 respectivement. Elles sont obtenues en coupant la surface par des plans perpendiculaires au plan tangent au point considéré [7]. Elles permettent de déterminer la forme locale de la surface dans le voisinage d'un point de coordonnées paramétriques u et v .

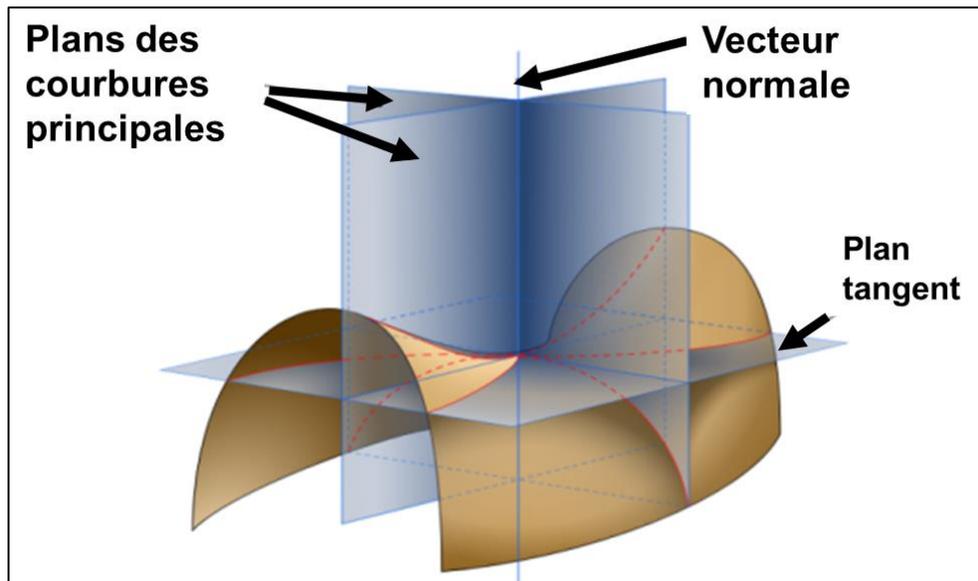


Figure 8. Courbures d'une surface.

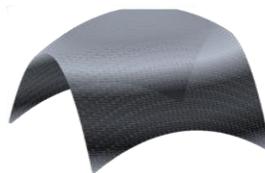
2.6. Formes locales :

De la géométrie différentielle, la forme locale au voisinage d'un point peut être une de six formes possibles en fonction des valeurs des courbures principales k_1 et k_2 [8] :

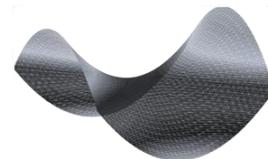
- 1 Si $k_1 > 0$ et $k_2 > 0$: le point est elliptique (Figure 9-a).
- 2 Si $k_1 < 0$ et $k_2 < 0$: le point est elliptique (Figure 9-b).
- 3 Si $k_1 < 0$ et $k_2 > 0$ le point est hyperbolique (Figure 9-c).
- 4 Si $k_1 = 0$ et $k_2 > 0$: le point est parabolique (Figure 9-d).
- 5 Si $k_1 < 0$ et $k_2 = 0$: le point est parabolique (Figure 9-e).
- 6 Si $k_1 = 0$ et $k_2 = 0$: le point est plan (Figure 9-f).



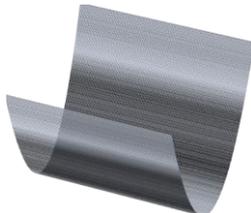
a. Concave.



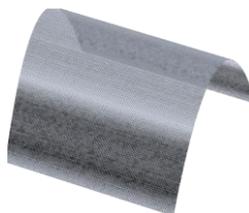
b. Convexe.



c. Selle de cheval.



d. Concave développable.



e. Convexe développable.



f. Plane.

Figure 9. Différentes formes locales.

2.7. Représentations discrètes :

Les modèles polyédriques sont générés par construction de surfaces à partir de nuages de points numérisés ou bien par facettisation (discrétisation) de modèles « CAO » paramétriques décrivant des formes complexes. Un maillage est constitué d'éléments géométriques simples tels que les sommets, les arêtes et les faces. Les sommets connectés les uns aux autres par des faces (ou facettes) de forme polygonale. Plus le nombre de facettes du maillage augmente, plus la représentation est fine et fidèle à une surface complexe. La plupart des maillages sont triangulaires ou quadrangulaires. Quand toutes les faces sont des triangles, le maillage est triangulaire.

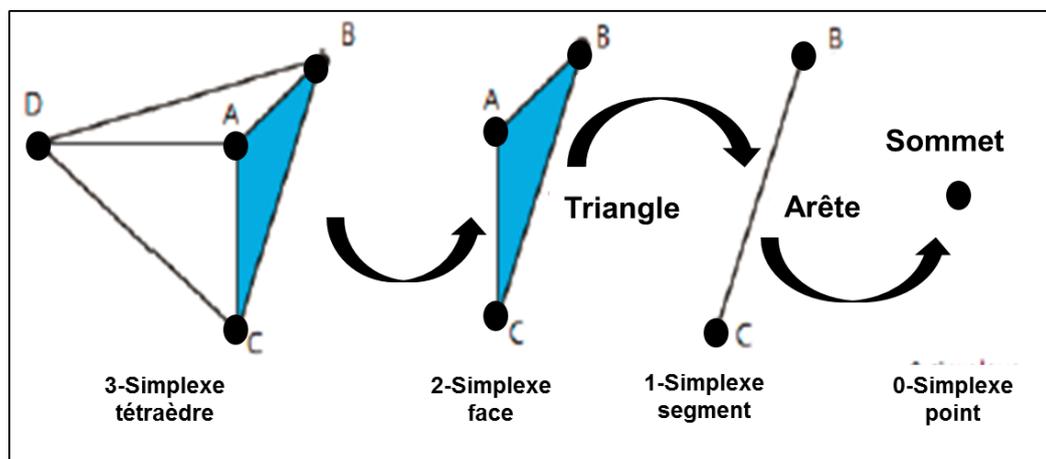


Figure 10. Détail d'un maillage.

2.7.1. Vecteur normal et plan tangent :

2.7.1.1. Vecteur normal :

Le vecteur normal est une propriété géométrique locale d'une surface et spécifique à un point donné. L'estimation du vecteur normal en un point de données dans un ensemble de données du nuage de points implique au moins les trois étapes principales suivantes:

A. Identification des points voisins applicables :

Trop de points voisins utilisés pour l'estimation du vecteur normal dégradent la caractéristique locale du vecteur normal et peu de points peuvent ne pas suffire à représenter de manière satisfaisante la géométrie locale.

B. Estimation du vecteur normal :

Une des méthodes d'estimation du vecteur normal en un point P consiste à calculer la moyenne pondérée des vecteurs normaux de triangles de mailles locales de P.

$$\vec{N} = \frac{\sum_{i=1}^k w_i \cdot \vec{n}_i}{\sum_{i=1}^k w_i} \quad (9)$$

Où n_i sont les vecteurs normaux des triangles partageant le même sommet et W_i sont les facteurs de pondération. Ces facteurs peuvent être de trois modes :

- Pondération par les aires des triangles. Dans ce cas, W_i représente l'aire du triangle.
- Pondération par les angles des triangles. Dans ce cas, W_i représente l'angle au sommet du triangle.
- W_i est pris égal à 1. Dans ce cas, c'est la moyenne des vecteurs normaux des triangles partageant ce sommet [9].

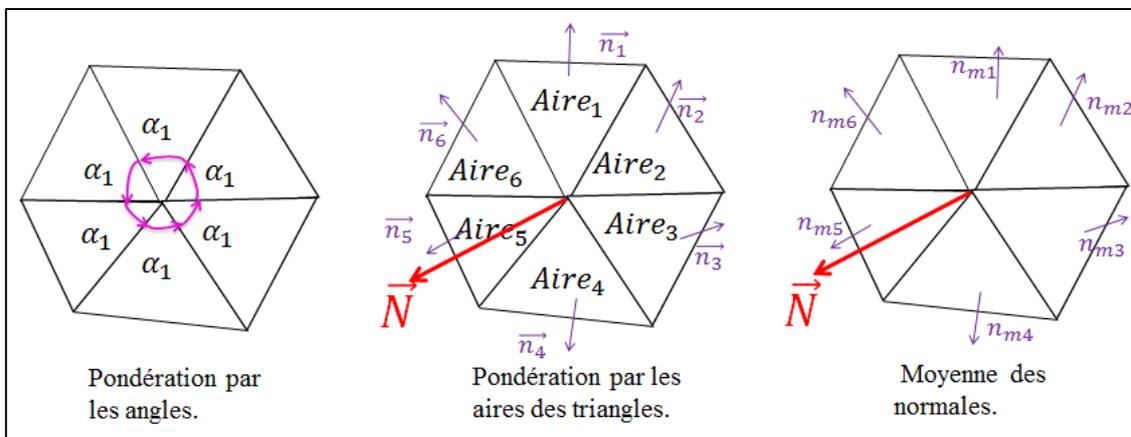


Figure 11. Normale d'un point.

C. *Établissement des directions interne et externe du vecteur normal.*

- **Plan tangent** : Le plan tangent à un sommet S est ici considéré comme étant le plan passant par S est orthogonal à la normale \vec{N}_S définie précédemment .

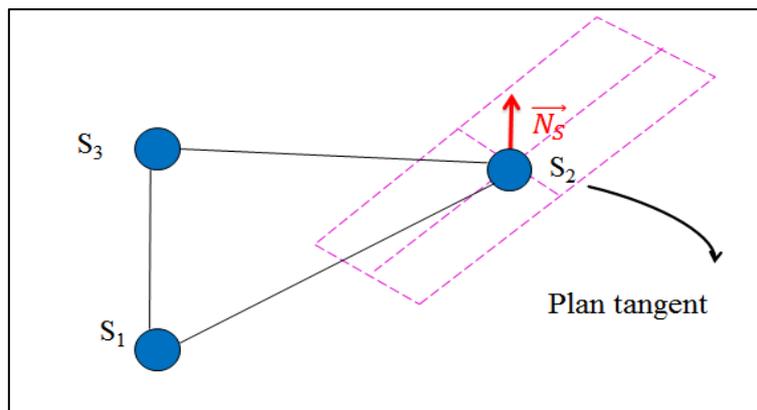


Figure 12. Plan tangent à un sommet S.

2.7.2. Classification des sommets :

Pour pouvoir classer les sommets, une approche basée sur des considérations purement géométriques est adoptée. Elle consiste à considérer la position des voisins au sommet considéré par rapport au plan tangent en ce point. Chaque sommet peut être assimilé à l'une des six (06) formes suivantes [10] :

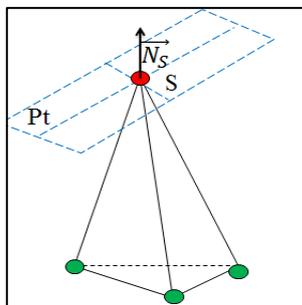
1. Un sommet est convexe si tous ses voisins sont en dessous du plan tangent.
2. Un sommet est concave si tous ses points voisins sont en dessus du plan tangent.
3. Un sommet est plat si ses voisins appartiennent au plan tangent
4. Un sommet est en selle de cheval s'il est à l'intérieur de l'enveloppe convexe de ses voisins.
5. Un sommet est convexe développable si une partie de ses voisins sont en dessous du plan tangent et une autre partie appartient au plan tangent.
6. Un sommet est concave développable si une partie de ses points voisins sont en dessus du plan tangent et une partie appartient au plan tangent

Afin de déterminer la forme locale d'un point, l'équation du plan tangent doit être calculée. Cette équation est donnée par :

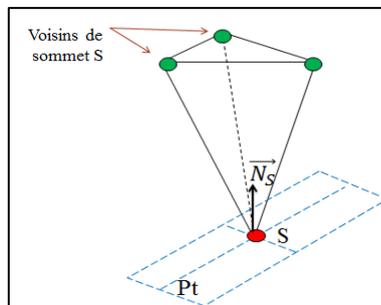
$$A.X + B.Y + C.Z + D = 0 \quad (10)$$

Pour un point de coordonnées X_0 , Y_0 et Z_0 , et de composantes du vecteur normal N_x , N_y et N_z , les constantes du plan tangent sont :

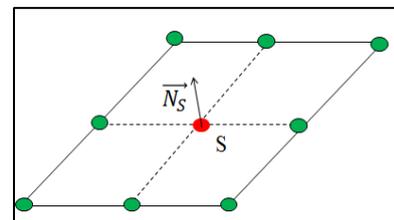
$$\begin{aligned} A &= N_x \\ B &= N_y \\ C &= N_z \\ D &= -(AX_0 + BY_0 + CZ_0) \end{aligned} \quad (11)$$



a. Convexe.



b. Concave.



c. Plat.

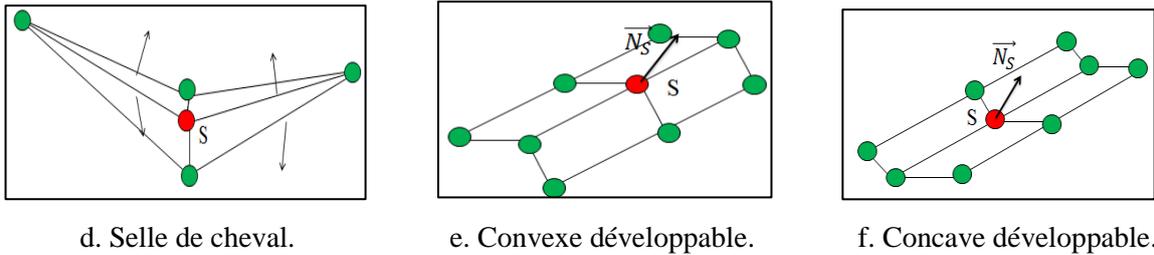


Figure 13. Formes locales d'un sommet.

Pour déterminer la position d'un point voisin par rapport au plan tangent en un point donné, le point extrémité du vecteur normal en ce point est pris comme un point de référence.

La procédure de ce processus suit les étapes suivantes :

- Calculer les coordonnées de l'extrémité du vecteur normal (X_0+N_x ; Y_0+N_y , Z_0+N_z).
- Remplacer ces coordonnées dans l'équation du plan tangent et récupérer le signe.
- Pour chaque point voisin, remplacer ces coordonnées dans l'équation du plan tangent et récupérer le signe. Si les deux signes sont identiques, le point est au-dessus du plan tangent. Sinon, le point est au-dessous du plan tangent [10].

3. Formats d'échange de données :

C'est la convention qui permet d'échanger des données entre divers programmes ou logiciels par l'intermédiaire d'un fichier. Plusieurs formats d'échange existent, les formats natifs, neutres, et les passerelles tels que STEP, IGES, STL, DXF, DWG etc. Le choix d'un de ces formats dépend du besoin et de l'utilisation. Malgré les nombreux standards existants, aucune solution universelle n'existe pour transférer un modèle CAO d'un système à un autre. Les erreurs de conversion ou les pertes d'informations lors du transfert sont fréquentes [11].

3.1. Modèle STL (Standard Tessellation Language):

Le format « STL » est largement utilisé pour le prototypage rapide et la fabrication assistée par ordinateur. Il a été lancé par 3D système en 1989. Le fichier « STL » décrit un objet par sa surface externe. Cette surface est nécessairement fermée et définie par une série de triangles. Chaque triangle est défini par les coordonnées cartésiennes (x , y , z) dans un trièdre direct, de son vecteur normal « n » orienté vers l'extérieur et de ses trois sommets ordonnés dans le sens trigonométrique [11]. Chaque triangle doit partager deux (02) sommets avec chacun des triangles le juxtaposant.

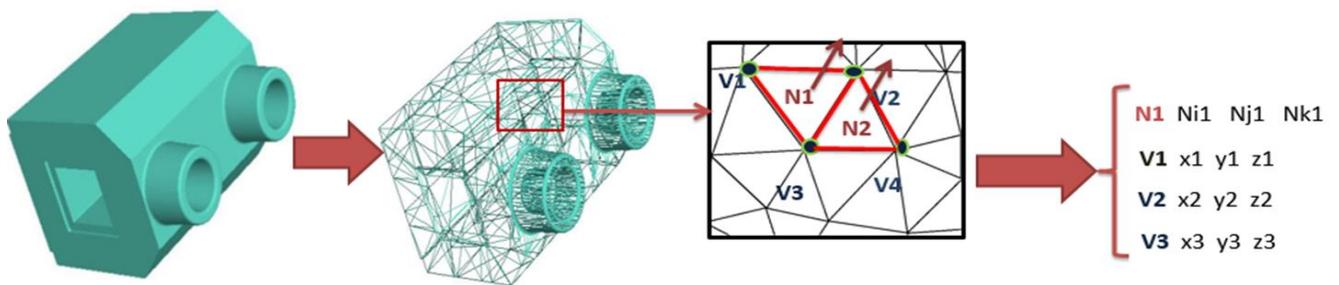


Figure 14. Représentation CAO, représentation STL et paramètres d'un triangle.

3.2. Structure d'un fichier STL :

Ils existent deux formats de fichier « STL », un format ASCII et un format binaire (Figure 15).

- **Format STL ASCII :**

Le format ASCII est spécifié comme un fichier ASCII (caractère imprimable) qui est plus descriptif et lisible mais gros de taille [3].

- Le fichier ASCII commence par une ligne précédée par le mot réservé « **Solid** ».
- Suivi par les coordonnées des triangles (la normale =facet normal et les sommets=vertex de chaque e triangle). Cette partie est répétée une ou plusieurs fois.
- La fin de fichier est indiquée par« **endsolid** » [3].

	Bytes	Data type	Description
<code>solid name</code>	80	ASCII	Header. No data significance.
<code>facet normal n_i n_j n_k</code>	4	unsigned long integer	Number of facets in file
<code>outer loop</code>			
<code>vertex v1 v1 v1</code>	4	float	<i>i</i> for normal
<code>vertex v2 v2 v2</code>	4	float	<i>j</i>
<code>vertex v3 v3 v3</code>	4	float	<i>k</i>
<code>endloop</code>			
<code>endfacet</code>			
<code>endsolid name</code>			

a. ASCII

b. binaire

Figure 15. Syntaxe d'un fichier STL.

Le tableau suivant résume les avantages et les inconvénients du format STL.

Tableau 1. Avantages et les inconvénients de formats STL.

Avantages	Inconvénients
<ul style="list-style-type: none"> • Format de fichier compact et simple. • Compatible avec de nombreux logiciels de modelage 3D. • Très répandu dans l'industrie des prototypes. 	<ul style="list-style-type: none"> • Supprime les informations de connectivité. • Erreurs topologiques pour géométrie complexe. • Fichier volumineux.

4. Usinages des surfaces gauches :

La conception et la fabrication constituent deux fonctions essentielles du processus de réalisation d'un produit manufacturier. La tendance actuelle consiste à utiliser des processus commandés numériquement appelés les systèmes « CFAO ». Ce processus doit permettre de garantir la fidélité entre la pièce produite et les spécifications fonctionnelles. Le processus de production d'une pièce passe par trois étapes (Figure 16) [11]:

- **Conception du modèle de la pièce :** la pièce est conçue sur ordinateur grâce aux outils de « CAO » appropriés en fonction des besoins de l'utilisateur final.
- **Génération des trajectoires d'usinage :** selon les outils utilisés, un ensemble de trajectoires sur toute la pièce est défini. Ces trajectoires indiquent les chemins suivis par l'outil pour usiner la pièce.
- **Usinage de la pièce :** l'usinage de la pièce effectif est alors réalisé en traduisant les trajectoires générées en un ensemble d'instructions exécutables par la commande numérique de la machine-outil. Ces instructions appelées aussi « G-Code » définissent les positions et les orientations des outils susceptibles de balayer toute la surface à usiner sans endommager ni la machine, ni les outils ni la pièce elle-même

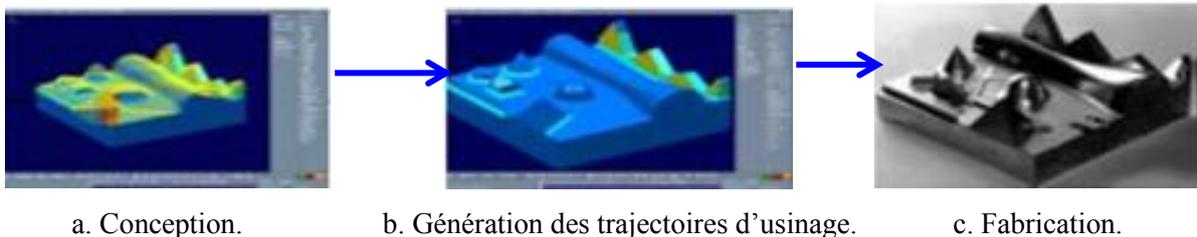


Figure 16. Processus d'usinage.

4.1. Usinage des pièces complexes:

L'usinage est une famille de techniques de fabrication de pièces par enlèvement de matière. Il consiste à enlever de la matière de façon à donner à la pièce brute la forme et les dimensions prescrites dans les dessins techniques à l'aide d'une machine-outil. Cette technique permet l'obtention de pièces d'une grande précision. Lors de l'usinage d'une pièce, l'enlèvement de matière est réalisé par la conjonction de deux mouvements relatifs entre la pièce et l'outil à savoir : le mouvement de coupe (vitesse de coupe) et le mouvement d'avance (vitesse d'avance) (Figure 17). Les procédés d'usinage les plus fréquents sont le tournage et le fraisage. Le premier procédé génère des pièces de révolution et le second génère des pièces prismatiques. Dans ce travail, le fraisage est considéré. Dans ce procédé, l'outil est animé d'un mouvement de rotation (mouvement de coupe) et la pièce est animée d'un mouvement de translation (mouvement d'avance).

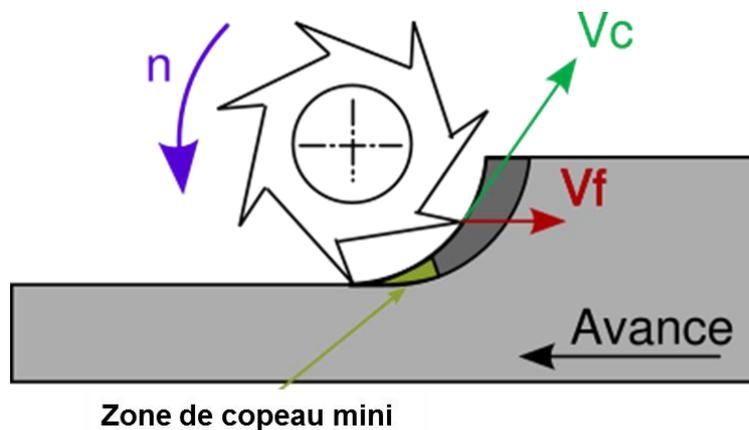


Figure 17. Vitesse d'avance et vitesse de coupe.

Les surfaces gauches sont usinées par fraisage sur des fraiseuses à 03-axes, à 04-axes et à 05-axes en fonction de leurs complexités géométriques [12-13].

4.1.1. Usinage en 03-axes :

Dans cette technique, trois translations simultanées suivant les axes X, Y et Z de la machine sont possibles (Figure 18.a). Lors de la finition en 03-axes, pour chaque point de contact entre l'outil d'usinage et la surface à usiner, il existe une unique position de l'outil tangente à la surface (Figure 18.b). Cette technique est plus appropriée pour des pièces sans parties en contre-dépouille.

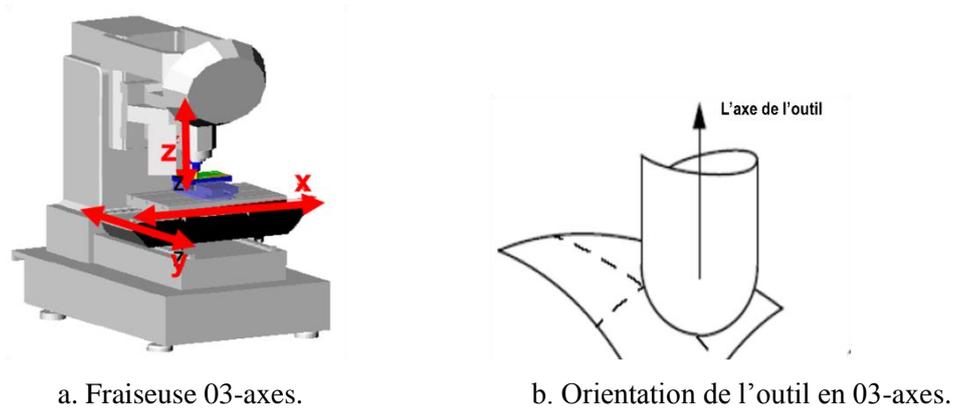


Figure 18. Usinage en 03-axes.

4.1.2. Usinage en 05-axes :

L'usinage en 05-axes fait appel à des machines-outils à cinq (05) degrés de liberté correspondant aux 03 axes linéaires de translation, auxquels s'ajoutent deux axes rotationnels (Figure 19.a). Une telle cinématique permet d'approcher la pièce dans toutes les directions et de la traiter sur cinq côtés en une seule opération (Figure 19.b). Contrairement à l'usinage en 03-axes, cette technique est parfaitement adaptée aux pièces complexes avec des parties en contre-dépouille et aux matériaux durs avec lesquels elle garantit un niveau de précision élevé grâce à l'utilisation d'outils d'usinage plus courts. La vitesse d'usinage est également plus grande, tandis que les vibrations d'outils sont réduites.

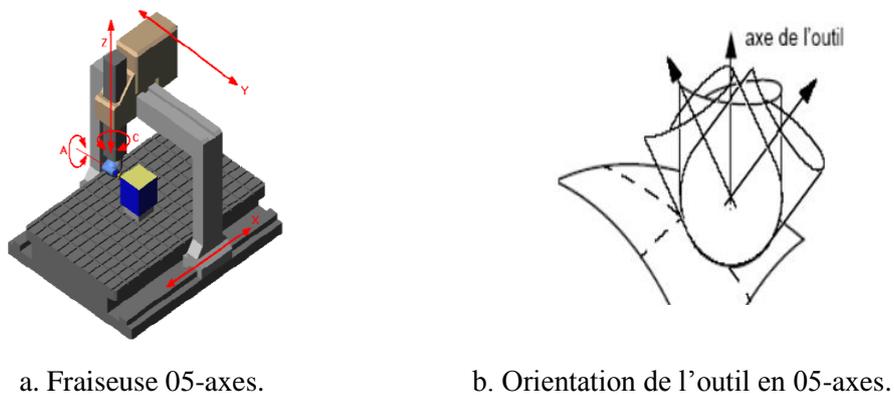


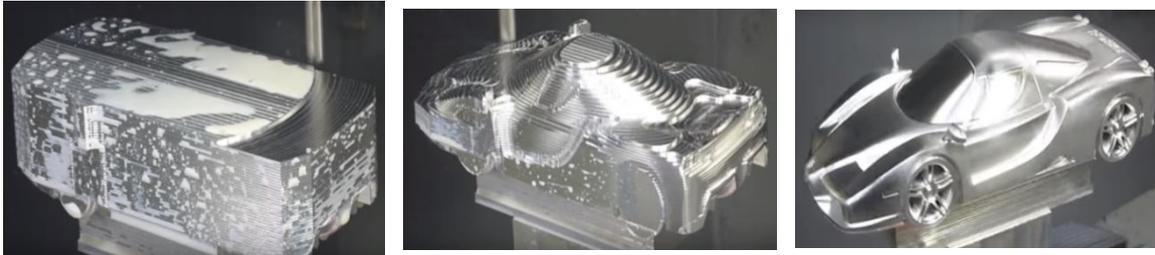
Figure 19. Usinage en 05-axes.

4.1.3. Phases d'usinage :

Afin de transformer une pièce brute de son état initial à son état final (pièce finie), il faut passer par trois phases (Figure 20) [14] :

- **Ébauchage** : pour l'enlèvement d'un maximum de matière en un minimum de temps.

- **Demi-finition** : pour supprimer les irrégularités de la surface laissées après la phase d'ébauchage et avoir une surépaisseur uniforme et donc une forme plus proche de la forme désirée.
- **Finition** : pour enlever la fine couche de matière pour obtenir la forme finale de la pièce conforme aux spécifications techniques et géométriques exigées.



a. Ebauchage.

b. Demi-finition.

c. Finition.

Figure 20. Phases d'usinage.

4.2. Stratégies d'usinage en finition:

Une stratégie d'usinage est une méthodologie utilisée pour générer une série d'opérations pour réaliser une forme donnée. Elle permet d'associer un processus d'usinage à une entité d'usinage, c'est-à-dire, un ensemble d'opérations comprenant la définition des outils, des conditions de coupe et des trajectoires d'usinage. Les stratégies d'usinage permettent de calculer l'ensemble des trajectoires d'outils lors de l'usinage sur machine-outil à commande numérique. Comme il n'y a pas une forme d'outil permettant de générer des surfaces gauches directement. Ces dernières sont usinées par balayage d'un outil selon une direction privilégiée caractérisant le mode de balayage. Le choix d'un mode est basé sur un critère d'optimisation du temps d'usinage ou de la qualité des surfaces à usiner. Les stratégies les plus utilisées pour la finition des surfaces gauches sont les suivantes [15] :

- **Plans parallèles** : elle s'appuie sur des trajectoires d'outil résultantes de l'intersection entre la surface à usiner et un ensemble de plans verticaux parallèles (Figure 21.a).
- **Z-Constant** : elle s'appuie sur des trajectoires d'outil résultantes de l'intersection entre la surface à usiner et un ensemble de plans horizontaux parallèles (Figure 21.b).
- **Isoparamétrique** : elle s'appuie sur le plan paramétrique de la surface pour générer le trajet d'outil avec plusieurs modes (Figure 21.c).

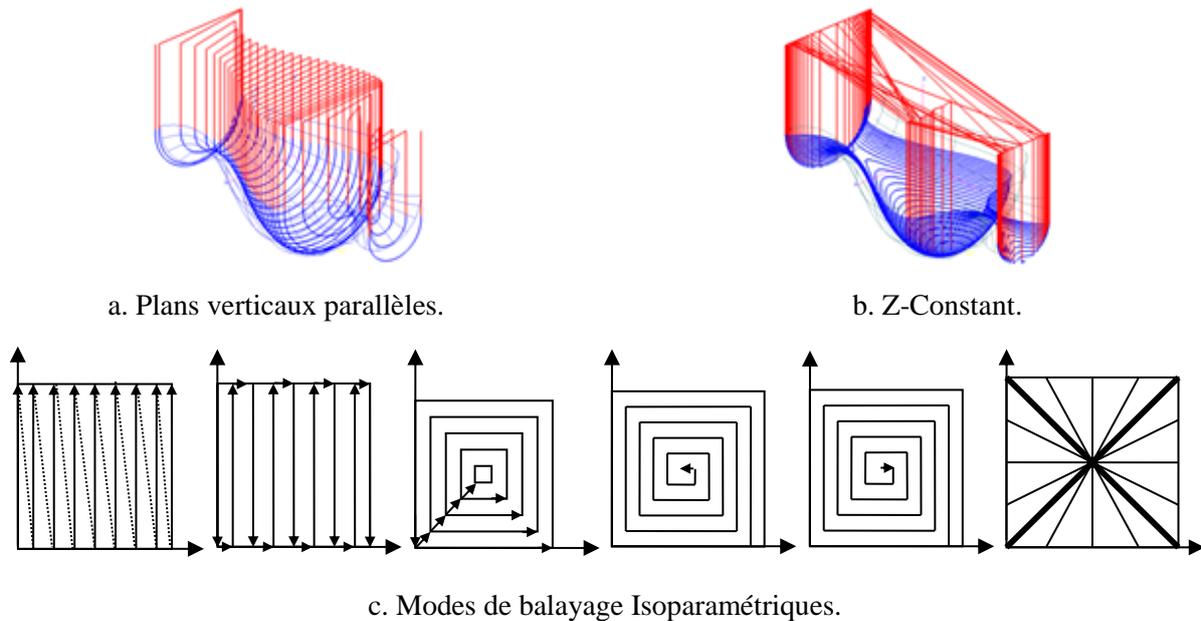


Figure 21. Stratégies de finition des surfaces gauches.

4.3. Machine-outil à commande numérique « MONC » :

Une machine-outil est une machine destinée à exécuter un usinage avec une précision et une puissance adaptées. Elle maintient un outil fixe, mobile, ou tournant et lui imprime un mouvement afin d'usiner ou de déformer une pièce fixée sur une table fixe ou mobile. Une machine-outil à commande numérique « MOCN » désigne la famille des machines-outils dotées d'une commande numérique « CN ». La « CN » correspondant aux matériels et logiciels permettant de donner les instructions de mouvements à tous les organes d'une machine-outil. Dans le monde de l'usinage, deux grandes familles de machines-outils sont utilisées : machines de fraisage et machines de tournage. Ces « MOCN » peuvent être équipées de magasins d'outils où le changement d'outils est automatique [16].

4.3.1. Définition des axes pour une fraiseuse numérique à 05-axes:

Une fraiseuse à commande numérique à 05-axes possède trois axes de translation et deux axes de rotation. Les axes de translation sont parallèles aux directions X, Y et Z du système de coordonnées de la machine et les axes de rotation sont perpendiculaires entre eux. Par convention, l'axe de rotation autour de l'axe des abscisses est appelé axe A, autour de l'axe des ordonnées est appelé axe B et autour de l'axe Z est appelé axe C (Figure 22). Les machines se différencient alors par la position des axes rotatifs. Les trois configurations possibles sont :

- Axes sur la tête: c'est la tête qui comporte les deux axes de rotation. Les axes X, Y et Z étant sur la table ou sur la tête.
- Axes sur table : c'est la table qui comporte les deux axes de rotation.

- Axes répartis sur la tête et sur la table.

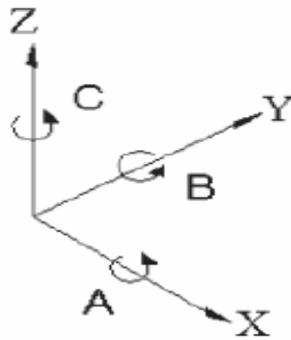


Figure 22. Stratégies de finition des surfaces gauches.

4.3.2. Outil de coupe en fraisage:

L'outil de coupe « fraise » est un organe monté sur la tête de la machine-outil dont l'action se produit lors du contact avec la pièce pour enlever la matière (copeaux) pour produire une surface conforme aux exigences. Il est défini par des critères géométriques (forme et dimensions) et technologiques (conditions de coupe et modes d'attachement) [17].

4.3.3. Différentes parties de l'outil :

Les parties principales d'un outil de coupe sont:

- **Corps** : il assure la liaison de l'outil avec le porte-outil. Sa forme peut être prismatique ou cylindrique.
- **Queue** : pour les fraises, la « queue » est l'élément qui permet la liaison de l'outil avec le porte-outil. Sa forme peut être cylindrique ou conique.
- **Partie active** : c'est la partie qui agit directement sur la pièce pour provoquer un enlèvement de matière. Elle est caractérisée par sa forme.

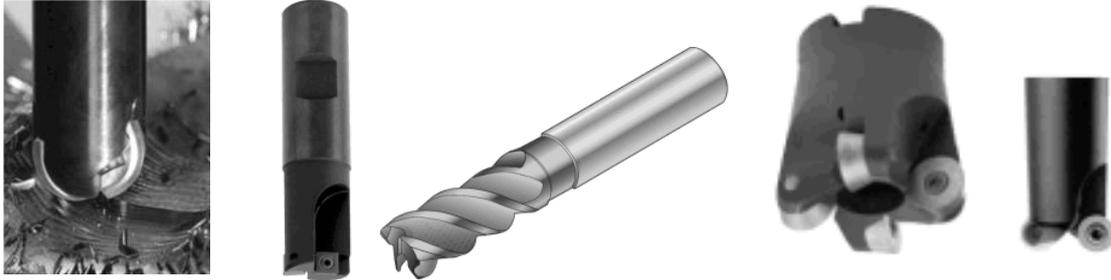
4.3.4. Types d'outils de coupe:

Les outils de finition des surfaces gauches sont définis par une enveloppe de types cylindrique, hémisphérique ou torique [18].

➤ **Fraise hémisphérique** : l'outil hémisphérique est l'outil le plus utilisé vu sa simplicité et sa géométrie qui le rendent plus facile à positionner pour l'usinage (Figure 23.a). Les fraises hémisphériques conviennent pour la finition de toutes les matières.

➤ **Fraise cylindrique** : avec leurs faces planes, elles conviennent aux travaux d'ébauche et de demi-finition (Figure 23.b). Leur avantage majeur réside dans leurs parties actives, le cylindre situé sur un rayon éloigné de l'axe de l'outil dont la vitesse est non nulle.

➤ **Fraise torique** : les fraises toriques constituent un bon compromis entre les fraises cylindriques et les fraises hémisphériques (Figure 23.c). Son avantage majeur est sa partie active, le tore, situé sur un rayon éloigné de l'axe de l'outil dont la vitesse est non nulle.



a. Fraise hémisphérique.

b. Fraise cylindrique.

c. Fraise torique.

Figure 23. Formes des fraises de finition.

4.3.5. Positionnement et orientation de l'outil en 05-axes:

Lors de la finition des surfaces gauches, l'outil utilisé, quel que soit sa forme et ses dimensions, doit être constamment tangent à la surface à usiner en chaque point de contact. Le positionnement d'un outil en finition par rapport à la surface en un point de contact donné dépend du vecteur normal unitaire au point de contact, de l'orientation de l'axe de l'outil et de la forme et des dimensions de l'outil.

Les différentes positions d'un outil hémisphérique sont données par (Figure 24.a) [10] :

$$\begin{cases} \overrightarrow{OC_E} = \overrightarrow{OC_C} + r\vec{n} \\ \overrightarrow{OC_L} = \overrightarrow{OC_E} - r\vec{u} = \overrightarrow{OC_C} + r\vec{n} - r\vec{u} \end{cases} \quad (12)$$

Les différentes positions d'un outil cylindrique sont données par (Figure 24.b) :

$$\begin{cases} \overrightarrow{OC_E} = \overrightarrow{OC_C} + R \frac{\vec{k} \wedge \vec{u}}{\|\vec{k} \wedge \vec{u}\|} \\ \overrightarrow{OC_L} = \overrightarrow{OC_E} = \overrightarrow{OC_C} + R \frac{\vec{k} \wedge \vec{u}}{\|\vec{k} \wedge \vec{u}\|} \end{cases} \quad (13)$$

Avec : $\vec{k} = \frac{\vec{u} \wedge \vec{n}}{\|\vec{u} \wedge \vec{n}\|}$

Les différentes positions d'un outil torique sont données par (Figure 24.c) :

$$\begin{cases} \overrightarrow{OC_E} = \overrightarrow{OC_C} + r\vec{n} + R \frac{\vec{k} \wedge \vec{u}}{\|\vec{k} \wedge \vec{u}\|} \\ \overrightarrow{OC_L} = \overrightarrow{OC_E} - r\vec{u} = \overrightarrow{OC_C} + r\vec{n} - r\vec{u} + R \frac{\vec{k} \wedge \vec{u}}{\|\vec{k} \wedge \vec{u}\|} \end{cases} \quad (14)$$

Avec : $\vec{k} = \frac{\vec{u} \wedge \vec{n}}{\|\vec{u} \wedge \vec{n}\|}$

Avec : C_C le point de contact entre l'outil et la surface, C_E le point centre de l'outil, C_L le point extrémité de l'outil, \vec{n} le vecteur normal unitaire à la surface au point de contact, \vec{u} orientation de l'axe de l'outil, r le rayon de l'outil hémisphérique et le petit rayon de l'outil torique, R le grand rayon de l'outil torique et le rayon de l'outil cylindrique.

Les équations précédentes permettent de calculer soit les coordonnées du centre de l'outil à partir du point de contact, soit le point de contact à partir des coordonnées du centre.

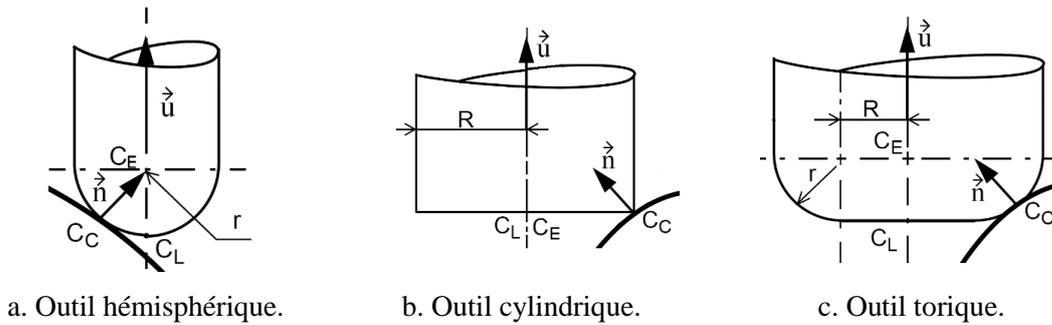


Figure 24. Positionnement des différentes formes d'outils.

Une fois la position de l'extrémité de l'outil déterminée, il reste à définir l'orientation de son axe. A chaque position de l'outil est associé un repère local dont l'origine est le point de contact et les deux directions principales sont la normale à la surface et la direction d'avance f . Dans un premier temps, l'outil est orienté, pour que son axe soit vertical. Par la suite, deux rotations sont appliquées pour orienter l'axe dans la direction appropriée. Les rotations peuvent être différentes. Aucune règle n'est imposée et il est possible de proposer d'autres modes d'orientation de l'axe de l'outil. Parmi ces modes :

- **Premier mode** : les deux rotations sont vers le segment et autour du vecteur normal à la surface au point de contact (Figure 25).
- **Deuxième mode** : les deux rotations sont vers le segment et autour du segment. Généralement, c'est ce mode qui est le plus utilisé en raison de sa simplicité.

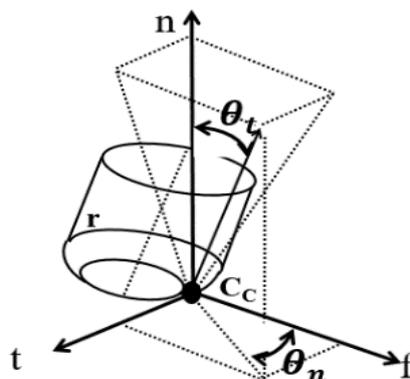


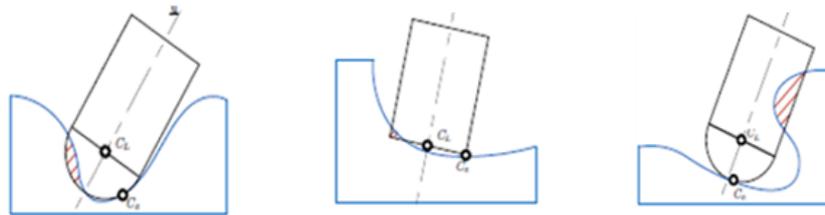
Figure 25. Orientation de l'outil.

4.4. Problèmes d'interférences :

L'usinage en 05-axes est puissant et peut résoudre de nombreux problèmes rencontrés en usinage 03-axes. Malgré cet avantage, les deux rotations supplémentaires augmentent considérablement les risques d'interférences et de collisions. Ces problèmes peuvent apparaître au niveau des positionnements d'outils calculés, lors de l'interpolation de la trajectoire ou encore lors des mouvements hors matière. L'orientation efficace de l'axe de l'outil pour un point de contact outil pièce donné, représente le problème majeur le plus rencontré en usinage 05-axes. Le positionnement et le parcours de l'outil sur la surface ne peuvent pas être dissociés; ils font l'objet d'une optimisation qui consiste le plus souvent à maximiser la largeur usinée avec comme contrainte la non interférence. Les interférences sont en général classées selon trois types : interférences locales, interférences vers l'arrière et interférences globales ou collisions [11].

- **Interférences locales** : elles traduisent un enlèvement de matière excessif par la partie active de l'outil sur la surface à usiner (Figure 26.a).
- **Interférences vers l'arrière** : elles sont des pénétrations intempestives de l'arrière de l'outil dans la surface à usiner (Figure 26.b).
- **Collisions** : elles représentent des collisions entre l'ensemble (corps d'outil, porte outil, broche) et l'ensemble (pièce, porte pièce, etc.) (Figure 26.c).

Ces trois types d'interférences ont un impact différent sur la pièce usinée. Ils peuvent être source de génération de défauts géométriques.



a. Interférence locale. b. Interférence vers l'arrière. c. Collision.

Figure 26. Types d'interférences.

4.4.1. Evitement des interférences et des collisions :

Pour un point de contact C_C le long du trajet d'outil, les paramètres d'orientation de l'outil choisis ne garantissent pas un usinage sans interférences et sans collisions. La gestion des interférences est un problème difficile à résoudre en usinage 05-axes. Parmi les méthodes de détection des interférences existantes, il y a celles qui sont basées sur les caractéristiques locales de la surface à usiner telles que la courbure, et celles qui optent pour une

représentation approchée par des points ou des modèles polyédriques. De plus, la plupart de ces méthodes tentent d'optimiser la position de l'outil hors interférences en termes de temps d'usinage. L'objectif consiste alors à positionner l'outil hors interférence afin d'obtenir un enlèvement de matière maximum. Pour corriger les interférences et les collisions, deux solutions sont utilisées [11] :

- Changement d'outil ou bien l'orientation de l'outil.
- Pour les collisions, il suffit juste de réorienter l'outil, quel que soit sa géométrie.

La maîtrise des interférences et des collisions lors de la génération de trajectoires d'outils est une condition nécessaire pour garantir la conformité de la pièce en termes de géométrie.

5. Conclusion :

Dans ce chapitre, nous avons présenté d'une manière générale les différentes méthodes de modélisation des surfaces gauches en mettant en avant le modèle discret plus particulièrement le modèle « STL » support de notre étude. Par la suite, nous avons explicité le processus d'usinage des surfaces gauches sur des fraiseuses numériques à 05-axes et les différentes stratégies d'usinages. Nous avons terminé ce chapitre par la présentation des machines-outils à commande numérique et les différents outils d'usinages. Le prochain chapitre fera l'objet de la présentation de la conception de notre application.

CHAPITRE 2

ETUDE CONCEPTUELLE

Introduction.

1. Démarche proposée.
2. Modélisation de l'application avec UML.

Conclusion.

Introduction

L'usinage 05-axes a longtemps été considéré comme une technologie de haut niveau, exclusivement réservée aux industries aéronautiques et automobiles. Or, ses nombreux avantages lui valent un intérêt de plus en plus vif des secteurs du moule et de l'outillage. Pendant l'usinage, l'outil suit une trajectoire complexe dans l'espace 3D. De ce fait, des problèmes d'interférences, entre la partie active de l'outil et la surface à usiner, ainsi que des problèmes de collisions, entre les différentes parties de l'environnement d'usinage (machine, outil, pièce, etc.), peuvent surgir et peuvent altérer la qualité d'usinage. Pour cela, il est impératif d'éliminer ces problèmes avant l'exécution effective de l'usinage. Lors de l'usinage en 05-axes, en chaque point de contact, l'outil constamment tangent à la surface à usiner, possède une infinité d'orientations de son axe. Les deux rotations supplémentaires par rapport à l'usinage 03-axes augmentent les possibilités d'orientation et d'accès de l'outil aux différentes régions de la surface, mais en même temps augmentent considérablement les risques d'interférences et de collisions.

1. Solution proposée :

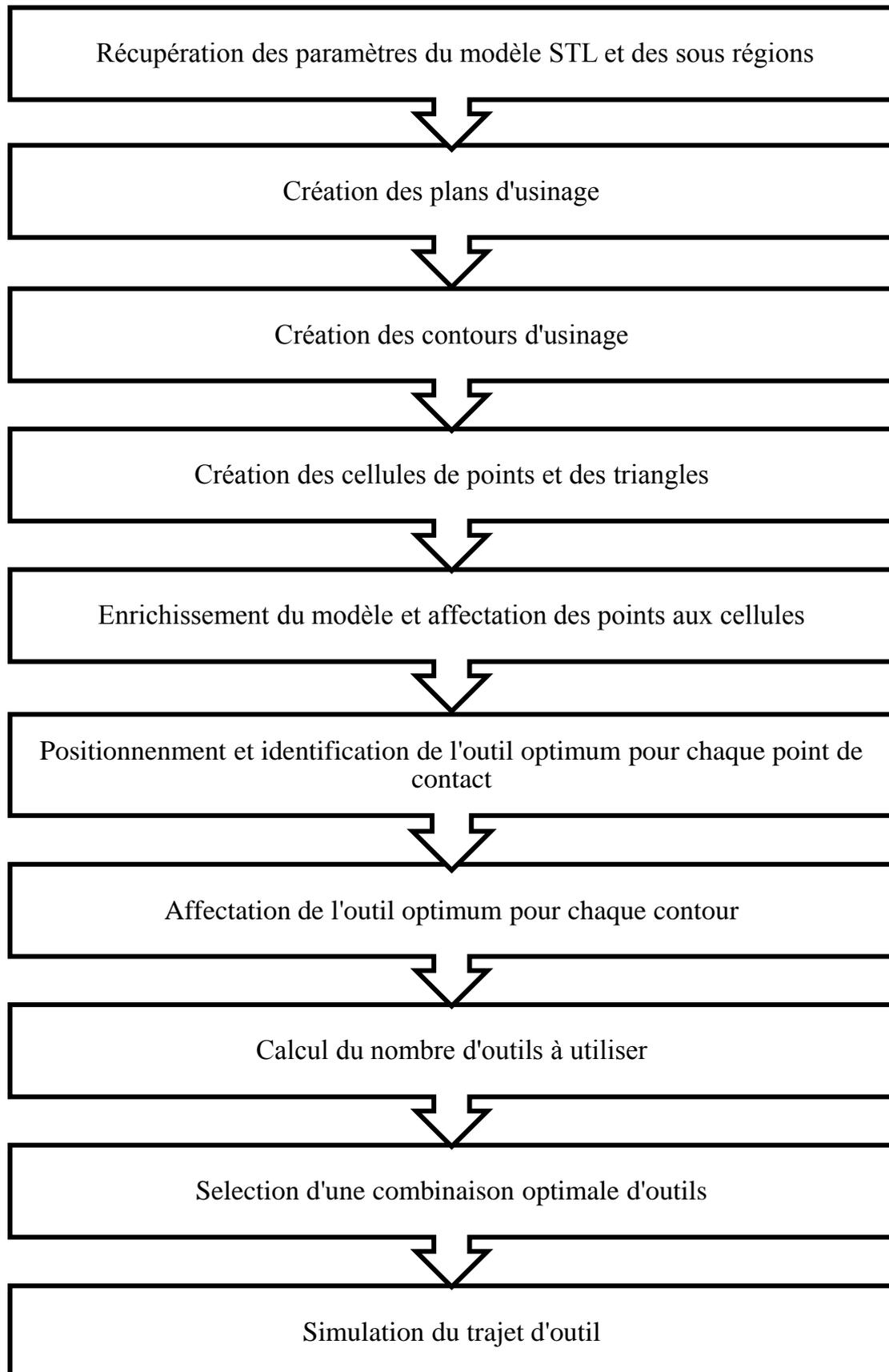
La solution proposée consiste à déterminer la combinaison et les orientations optimales des trois formes d'outils à savoir cylindriques, hémisphériques et toriques lors de la finition des surfaces gauches définies par leurs modèles STL sur des fraiseuses numériques à 05-axes (Organigramme 1).

1.1 Récupération des paramètres du modèle STL et des sous régions :

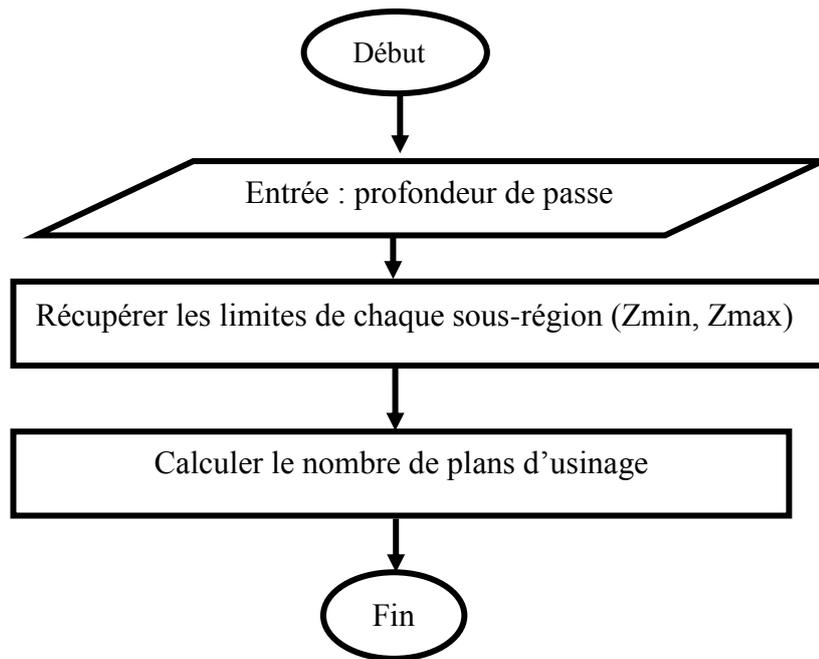
Dans cette étape, le code d'une application développée antérieurement [3] a été modifié par l'intégration de deux fonctions. La première pour sauvegarder les paramètres du modèle STL et les paramètres des sous régions dans un fichier texte. La deuxième pour récupérer tous ces paramètres nécessaires à l'usinage en 05-axes.

1.2 Création des plans d'usinage :

Afin de créer les plans d'usinage, la stratégie d'usinage Z-Constant a été adoptée. Elle consiste à diviser la surface à usiner par des plans horizontaux. Dans ce travail, les plans d'usinage (passes) pour chaque sous-région, sont déterminés en suivant les étapes données par l'Organigramme 2.



Organigramme 1. Démarche proposée.



Organigramme 2. Création des plans d'usinage.

1.3 Création des contours d'usinage :

Cette étape est composée de plusieurs tâches consécutives. Un contour est composé de segments issus de l'intersection de la surface à usiner avec des plans horizontaux. Pour déterminer ces segments, il faut d'abord identifier les points d'intersection entre les triangles du modèle STL de la surface et les plans d'usinage. La surface étant représentée par son modèle STL, l'intersection des plans avec les triangles du modèle de la surface génère des points qui sont reliés entre eux selon un ordre bien déterminé pour créer une chaîne fermée ou ouverte appelée contour (Figure 1) (Organigramme 3).

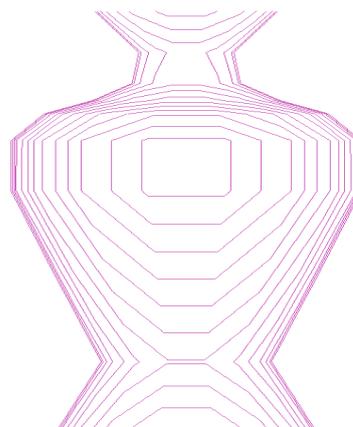
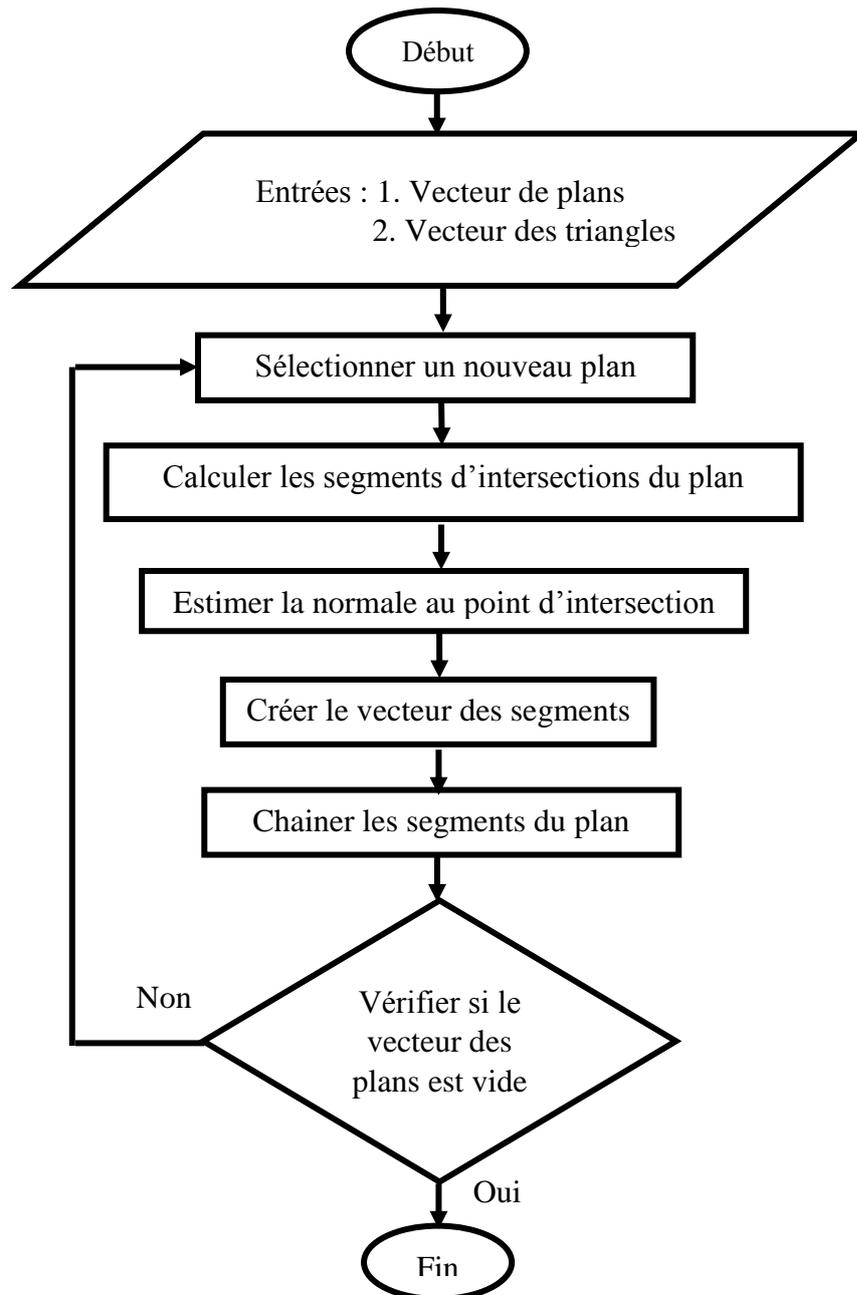


Figure 1. Contours de la pièce.



Organigramme 3. Création des contours.

1.3.1 Détermination des segments d'intersection :

1.3.1.1 Calcul des points d'intersection :

Le calcul des points d'intersection passe par le parcours du vecteur des triangles et pour chaque triangle, l'intersection avec le plan d'usinage sélectionné est calculée. Il en résulte un ou plusieurs (maximum 3) sommets du triangle d'intersection selon le cas. Les cas d'intersection entre un triangle et un plan sont illustrés par la Figure 2.

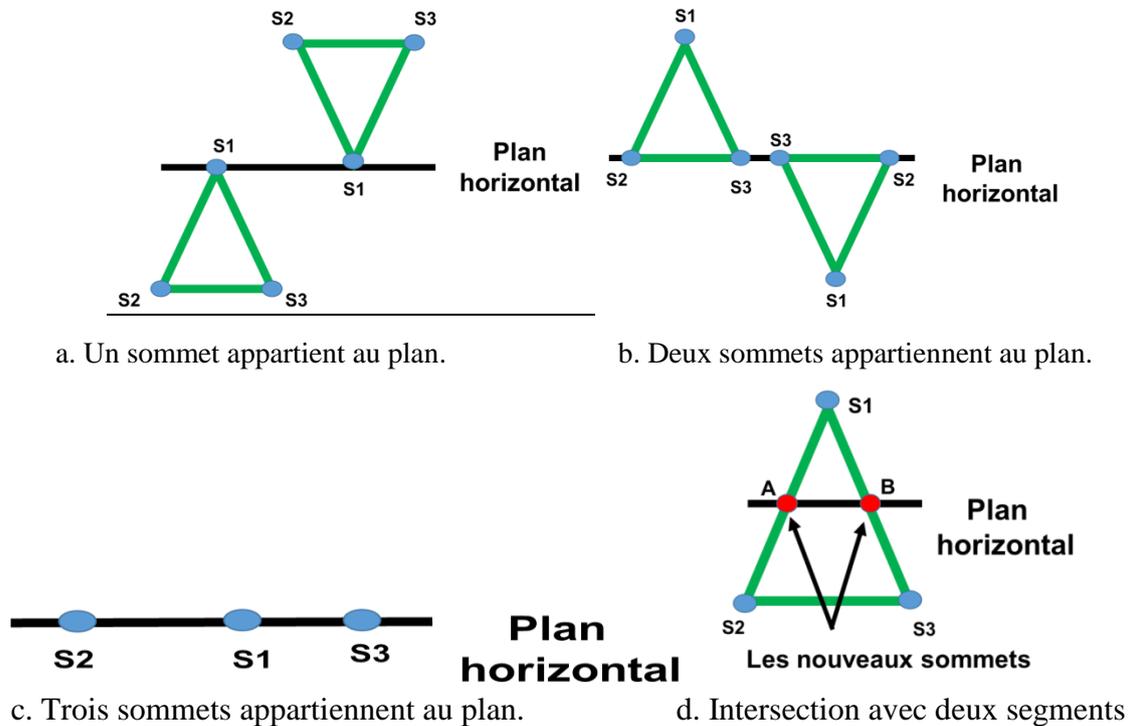


Figure 2. Intersection d'un plan avec un triangle.

1.3.1.2. Estimation de la normale en chaque point de contact :

Si le point d'intersection est un sommet, alors sa normale est déjà estimée dans la première étape (lecture du paramètre STL). Dans le cas où le point d'intersection n'est pas un sommet, alors la normale est calculée en utilisant la pondération par les distances par rapport aux sommets qui forment le segment.

1.4 Création des cellules de points et des triangles :

Le modèle STL est défini par un nombre très important de triangles et de sommets. Donc, la vérification des interférences et des collisions implique le parcours de tous les sommets du modèle. Cette vérification nécessite un temps de calcul très important. Afin de le minimiser, le brut est subdivisé en des blocs parallélépipédiques (cellules) de même taille (Figure 3). La création des cellules nécessite la spécification du nombre de cellules $N_{b_{x}}$, $N_{b_{y}}$ et $N_{b_{z}}$ suivant les trois axes X, Y et Z. A partir de ces données, les pas et les limites de chaque cellule sont calculés.

➤ Calcul des pas suivant les axes X, Y et Z :

$$\text{Pas}_x = \text{longueur du brut} / N_{b_{x}} \quad (1)$$

$$\text{Pas}_y = \text{largeur du brut} / N_{b_{y}} \quad (2)$$

$$\text{Pas}_z = \text{hauteur du brut} / N_{b_{z}} \quad (3)$$

- Calcul des limites de chaque cellule : les calculs sont résumés dans le Tableau 1.

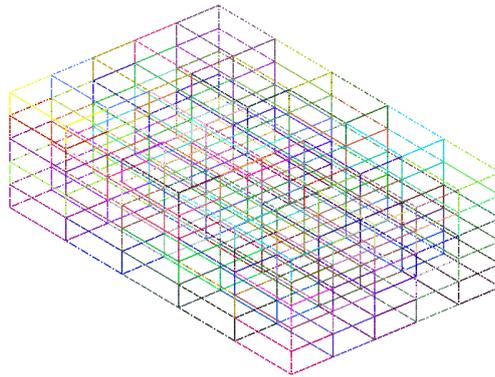


Figure 3. Création des cellules.

Tableau 1. Limites d'une cellule.

	Extrémités minimales	Extrémités maximales
Pour i de 0 à Nbrex	$X_{\min} = \text{Brut_}X_{\min} + i * \text{Pas_}x$	$X_{\max} = X_{\min} + \text{Pas_}x$
Pour j de 0 à Nbrej	$Y_{\min} = \text{Brut_}Y_{\min} + j * \text{Pas_}y$	$Y_{\max} = Y_{\min} + \text{Pas_}y$
Pour k de 0 à Nbrez	$Z_{\min} = \text{Brut_}Z_{\min} + k * \text{Pas_}z$	$Z_{\max} = Z_{\min} + \text{Pas_}z$

1.5 Enrichissement du modèle STL :

Lorsque le modèle STL est grossier, des problèmes d'interférences et de collisions peuvent ne pas être détectés. Ceci est dû au fait que la densité des points n'est pas importante et l'outil peut pénétrer dans la surface sans être repéré. Afin de remédier à ce problème, la densité des points est augmentée. L'approche utilisée consiste à ajouter de nouveaux points dans les triangles en se basant sur les coordonnées barycentriques. Le nombre de points à ajouter pour chaque triangle est déterminé en fonction de l'aire du triangle et la densité des points spécifiée par l'utilisateur. La procédure d'enrichissement se déroule comme suit :

Pour chaque triangle de sommets S_1 , S_2 et S_3 , les coordonnées P d'un point appartenant au triangle sont calculées en utilisant les coordonnées barycentriques par :

$$\overrightarrow{OP} = \alpha * \overrightarrow{OS_1} + \beta * \overrightarrow{OS_2} + \gamma * \overrightarrow{OS_3} \quad (4)$$

Avec:

$$0 \leq \alpha, \beta, \gamma \leq 1$$

$$\alpha + \beta + \gamma = 1$$

$$\alpha = 1 - \sqrt{r_1}$$

$$\beta = \sqrt{r_1 * (1 - r_2)}$$

$$\gamma = 1 - \sqrt{r1}$$

Où r1 et r2 sont des nombres réels aléatoires.

Une fois le traitement terminé, chaque point ajouté est affecté à la cellule qui lui correspond (Figure 4).

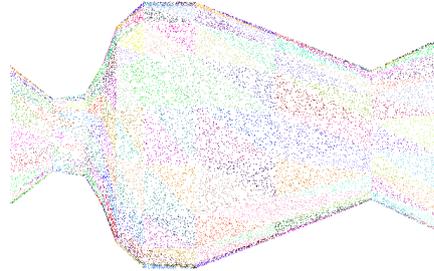


Figure 4. Enrichissement du modèle STL.

1.6 Affectation des points aux cellules :

Après la création des cellules et l'enrichissement du modèle, chaque sommet des triangles du modèle STL est affecté à la cellule correspondante. L'affectation des sommets aux cellules consiste à déterminer les indices de la cellule correspondante (Figure 5). Ils sont calculés en se basant sur les coordonnées (x, y, z) du point considéré et les coordonnées minimales du brut. Les indices de cette cellule sont calculés par :

$$I = \frac{x - X_{\min\text{-brut}}}{\text{Pas}_x} \quad (5)$$

$$J = \frac{y - Y_{\min\text{brut}}}{\text{Pas}_y} \quad (6)$$

$$K = \frac{z - Z_{\min_brut}}{\text{Pas}_z} \quad (7)$$

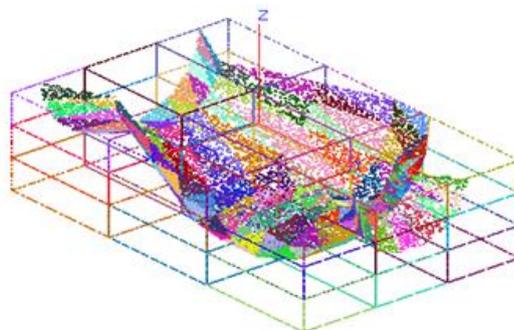


Figure 5. Affectation des points aux cellules.

1.7 Positionnement et outil optimum de chaque point de contact :

L'objectif du projet consiste à déterminer la combinaison et les orientations d'outils optimaux sans interférences et sans collisions. Cet objectif exige le passage par les étapes suivantes :

- Identification de l'outil optimum pour un point de contact.
- Détermination de l'outil optimum pour un contour.
- Sélection d'une combinaison d'outils parmi les outils des contours.

Le positionnement d'un outil consiste à déterminer le point centre OC_E et le point extrémité d'outil OC_L . Pour toutes les formes d'outils (hémisphériques, cylindriques et toriques), ces deux points sont données par :

$$OC_E = OC_C + r \cdot \vec{u} + (R - r) \cdot \vec{v} \quad (8)$$

$$OC_L = OC_E - r \cdot \vec{u} \quad (9)$$

Une fois le point extrémité est calculé, l'extrémité maximale d'outil, l'extrémité maximale de la partie active et l'extrémité maximale sont calculées :

$$OC_{LM} = OC_L + L \cdot \vec{u} \quad (10)$$

$$OC_{LA} = OC_L + L_A \cdot \vec{u} \quad (11)$$

$$OC_{LP} = OC_{LM} + L_p \cdot \vec{u} \quad (12)$$

Avec :

$$\vec{v} = \frac{\vec{u} \wedge \vec{n}}{\|\vec{u} \wedge \vec{n}\|} \wedge \vec{u}$$

L : longueur du corps d'outil.

L_A : longueur de la partie active d'outil.

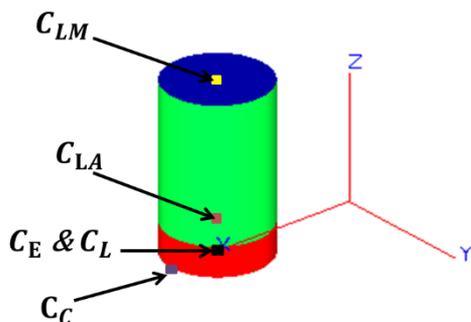


Figure 6. Outil cylindrique.

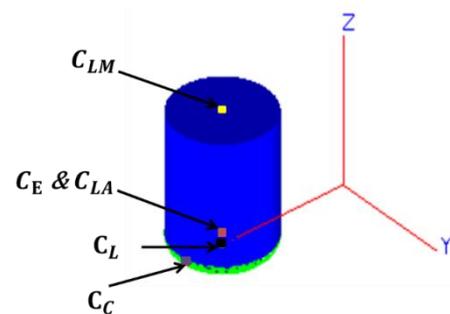


Figure 7. Outil torique.

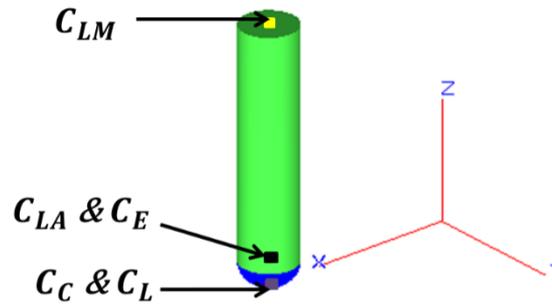


Figure 8. Outil hémisphérique.

Par la suite, les enveloppes des trois parties de l'outil sont calculées pour déterminer les cellules chevauchantes avec ces enveloppes.

- **Enveloppe de la partie corps :** les extrémités minimales et maximales de l'enveloppe de la partie corps sont données par :

$$\begin{cases} L_{\min} = \min(C_{LM} - R, C_{LA} - R) \\ L_{\max} = \max(C_{LM} + R, C_{LA} + R) \end{cases} \quad (13)$$

- **Enveloppe de la partie active :** les extrémités minimales et maximales de l'enveloppe de la partie active sont données par :

$$\begin{cases} L_{\min} = \min(C_{LA} - R, C_L - R) \\ L_{\max} = \max(C_{LA} + R, C_L + R) \end{cases} \quad (14)$$

- **Enveloppe partie porte outil :** les limites de l'enveloppe de la partie porte-outil sont données par :

$$\begin{cases} L_{\min} = \min(C_{LP} - D, C_{LM} - D) \\ L_{\max} = \max(C_{LP} + D, C_{LM} + D) \end{cases} \quad (15)$$

Où D représente le diamètre du porte-outil.

1.8 Détermination des zones d'interférences et de collisions :

L'identification de l'outil optimum en chaque point de contact consiste à trouver l'outil qui ne génère ni interférence ni collision. L'identification des zones d'interférences et de collisions consiste à localiser, pour chaque point de contact, les points des cellules qui se chevauchent avec l'enveloppe de l'outil (Figure 9). L'Organigramme 4 résume les principales étapes suivies pour atteindre cet objectif. Cette façon de faire permet d'éviter le test avec l'ensemble des points des surfaces.

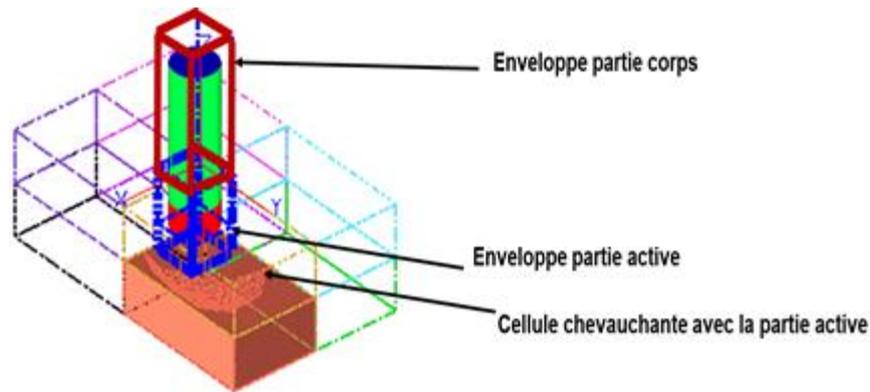
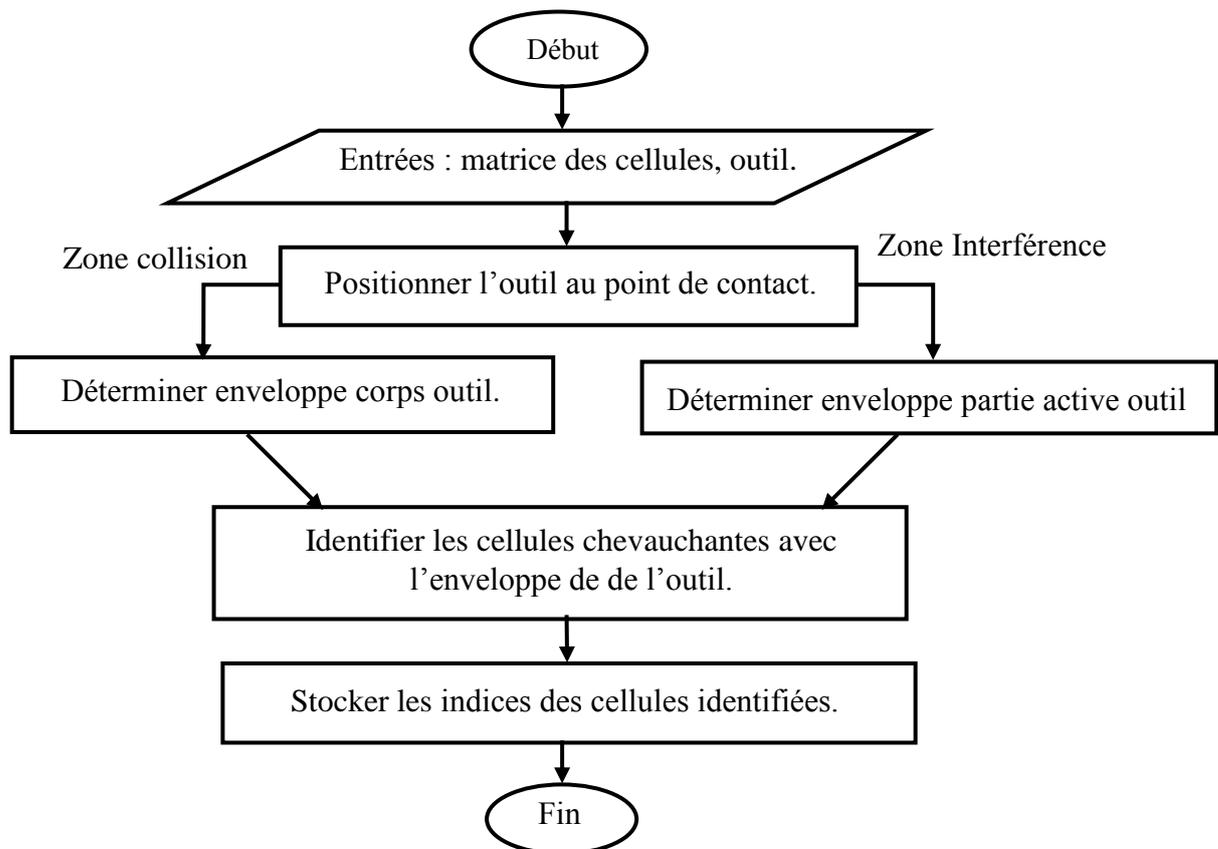


Figure 9. Zones d'interférences et de collisions.



Organigramme 4. Identification des zones d'interférences et collisions.

1.9 Tests d'interférences et de de collisions :

Une fois les zones d'interférences et de collision déterminées, les points de ces zones sont utilisés afin de tester l'existence d'interférences et de collisions pour chaque point de contact.

- **Collision avec le corps-outil :** La collision se produit lorsque des points de la surface à usiner sont situés à l'intérieur du corps de l'outil (partie cylindrique). Comme tous les types d'outils (cylindrique, hémisphérique et torique) ont la même forme du

corps-outil, alors le traitement de test doit être le même pour les trois formes (Figure 10).
Ce traitement suit les étapes suivantes :

- Pour chaque point P de coordonnées X, Y et Z des zones identifiées :
- Calculer la distance D entre le point et l'axe de l'outil par :

$$D = \frac{\|\overline{C_E P} \wedge \vec{u}\|}{\|\vec{u}\|} \quad (16)$$

- Si $D > R$, alors le point ne crée pas de collision.
- Si $D < R$, alors une possibilité de collision existe. Pour cela :
 - Vérifier l'appartenance du point au corps-outil:
 - S'il n'appartient pas, alors il y a une absence de collision.
 - S'il appartient, alors un problème de collision est détecté.

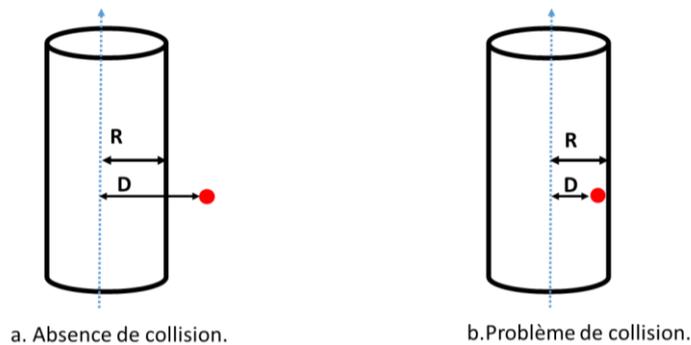


Figure 10. Collision avec corps-outil.

- **Interférence** : l'interférence se produit lorsque des points de la surface appartiennent à la partie active de l'outil.

- **Outil cylindrique et hémisphérique** : le test d'interférence pour l'outil cylindrique et l'outil hémisphérique consiste aussi à calculer la distance entre le point identifié et l'axe d'outil puis la comparer avec le rayon et vérifier l'appartenance à la partie active (Figure 11).

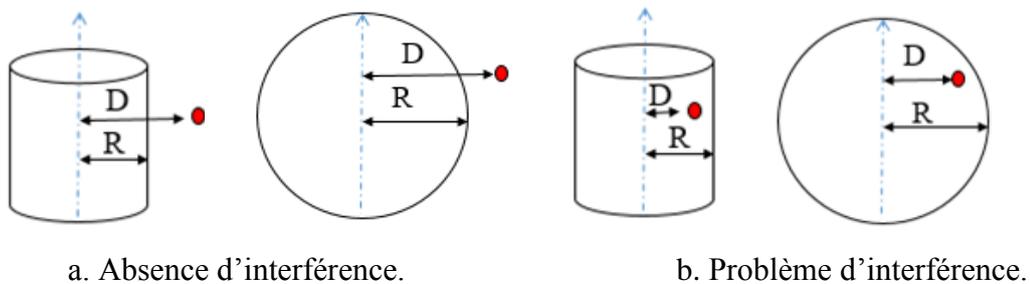


Figure 11. Interférence outil cylindrique et outil hémisphérique.

➤ **Outil torique** : le tore est composé de deux parties; une partie cylindre de rayon R et une de forme en demi tore de rayon r. L'interférence se produit si le point appartient à la partie cylindrique ou au tore. Pour vérifier cette condition, la distance entre le point et l'axe d'outil D est calculée :

- Si $D < R$, une interférence est détectée (interférence au niveau du cylindre).
- Si $D > R$ un risque d'interférence et donc un changement de repère s'impose.

Ce changement de repère consiste à déterminer les coordonnées du point de test dans un repère local lié à l'outil dont l'origine est le point extrémité minimale de l'outil C_L . Son premier vecteur unitaire T_1 est perpendiculaire au vecteur directeur de l'axe de l'outil U. Son deuxième vecteur unitaire T_2 est perpendiculaire au plan formé par les vecteurs U et T_1 . Son troisième vecteur unitaire est le vecteur directeur de l'axe d'outil U. Donc, les coordonnées des points seront exprimées dans le repère (T_1, T_2, U) . Le changement de repère se fait donc du repère global (repère de conception) vers le repère local (repère de vérification lié à l'outil) (Figure 12). Soit un point P de coordonnées (X_1, Y_1, Z_1) exprimées dans un repère global R_1 , ses coordonnées (X_2, Y_2, Z_2) exprimées dans le repère local R_2 sont données par :

$$P_{/R_1} = \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} \quad P_{/R_2} = \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} \quad (17)$$

Le passage du repère R_2 vers le repère R_1 est donné par :

$$P_{/R_1} = M * P_{/R_2} \quad (18)$$

$$M = \begin{bmatrix} T_{1x} & T_{1y} & T_{1z} & x_0 \\ T_{2x} & T_{2y} & T_{2z} & y_0 \\ U_x & U_y & U_z & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

Le passage du repère R_1 vers le repère R_2 est donné par :

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} * M^{-1} \quad (20)$$

Avec :

M^{-1} : matrice inverse de la matrice M.

X_0, Y_0, Z_0 : coordonnées de l'origine du repère R_2 exprimées dans le repère R_1 .

Une fois le changement est effectué, $r(z)$ le nouveau rayon de coupe est donné par :

$$r(z) = R + \sqrt{r^2 - (r^2 - z)^2} \quad (21)$$

- Si $D > r(z)$, absence d'interférence.
- Si $D < r(z)$, problème d'interférence est détecté.

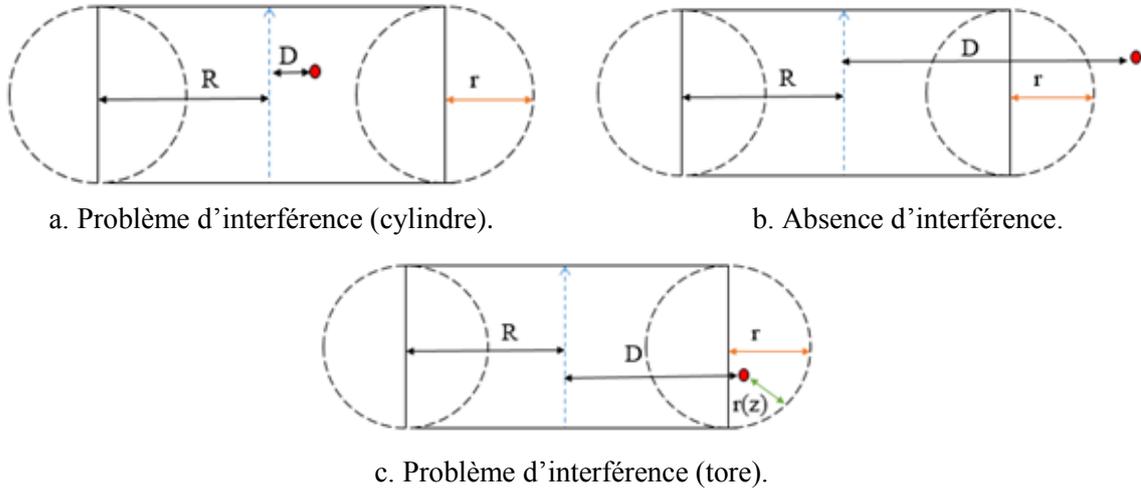


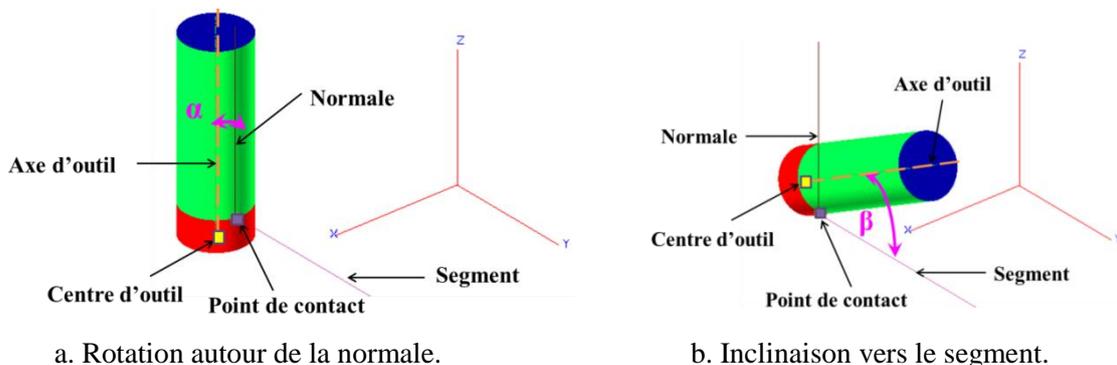
Figure 12. Interférence outil torique.

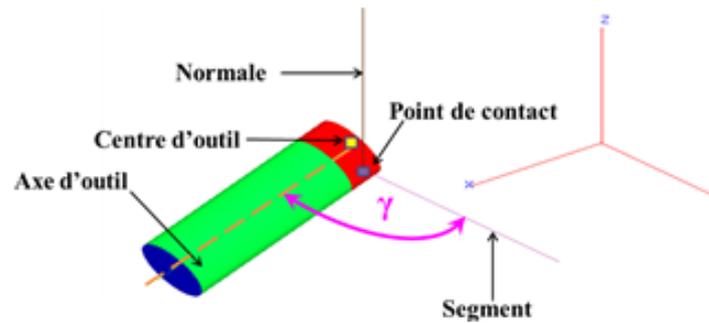
1.10 Correction des interférences et des collisions :

La correction de l'interférence et de la collision consiste à déterminer l'orientation de l'axe de l'outil optimale, c'est-à-dire l'orientation qui ne provoque ni interférences ni collisions. Trois modes d'orientations de l'axe de l'outil sont proposés :

- **Premier mode** : rotation de l'outil autour de la normale (Figure 13. a).
- **Deuxième mode** : inclinaison de l'outil vers le segment (Figure 13. b).
- **Troisième mode** : rotation de l'outil autour de segment (Figure 13. c).

Pour tous les modes et pour chaque point de contact, l'axe de l'outil est positionné initialement parallèle à la normale et en cas de collisions ou d'interférences, l'outil est orienté jusqu'à la détermination d'un angle qui élimine les collisions et les interférences.





c. Rotation autour du segment.

Figure 13. Correction des interférences et des collisions.

1.11 Identification de l'outil optimum de chaque point de contact :

L'affectation de l'outil optimum à chaque point de contact est une étape primordiale pour atteindre l'objectif principal de ce projet. Afin d'identifier un outil optimum pour un point de contact donné, la démarche suivante est suivie :

- a. Déterminer la forme d'outil à utiliser en fonction de la forme de la région :
 - *Forme locale concave, concave développable ou en celle de cheval* : les outils toriques seront choisis en premier lieu. En cas de test négatif, les outils hémisphériques seront choisis.
 - *Forme locale convexe, convexe développable ou plane* : les outils cylindriques seront choisis en premier lieu. En cas de test négatif, les outils hémisphériques seront choisis.
- b. Sélectionner à partir de la liste des outils le plus grand outil.
- c. Positionner l'outil sur ce point.
- d. Déterminer les cellules qui se chevauchent avec les enveloppes de la partie active et la partie corps de l'outil.

Deux cas sont possibles :

- Absence d'interférences et de collisions.
- Existence d'interférences et/ou de collision : dans ce cas, il faut trouver une position qui permet de dégager l'outil de sa position de chevauchement avec la surface. Pour cela, la correction consiste à déterminer l'orientation de l'axe de l'outil optimale, c'est-à-dire l'orientation qui ne provoque ni interférences ni collisions. Pour éviter ces problèmes, trois modes d'orientation de l'axe de l'outil peuvent être combinés :

1. Rotation de l'outil autour de la normale.
2. Inclinaison de l'outil vers le segment dans le sens de l'avance de l'outil.
3. Rotation autour du segment.

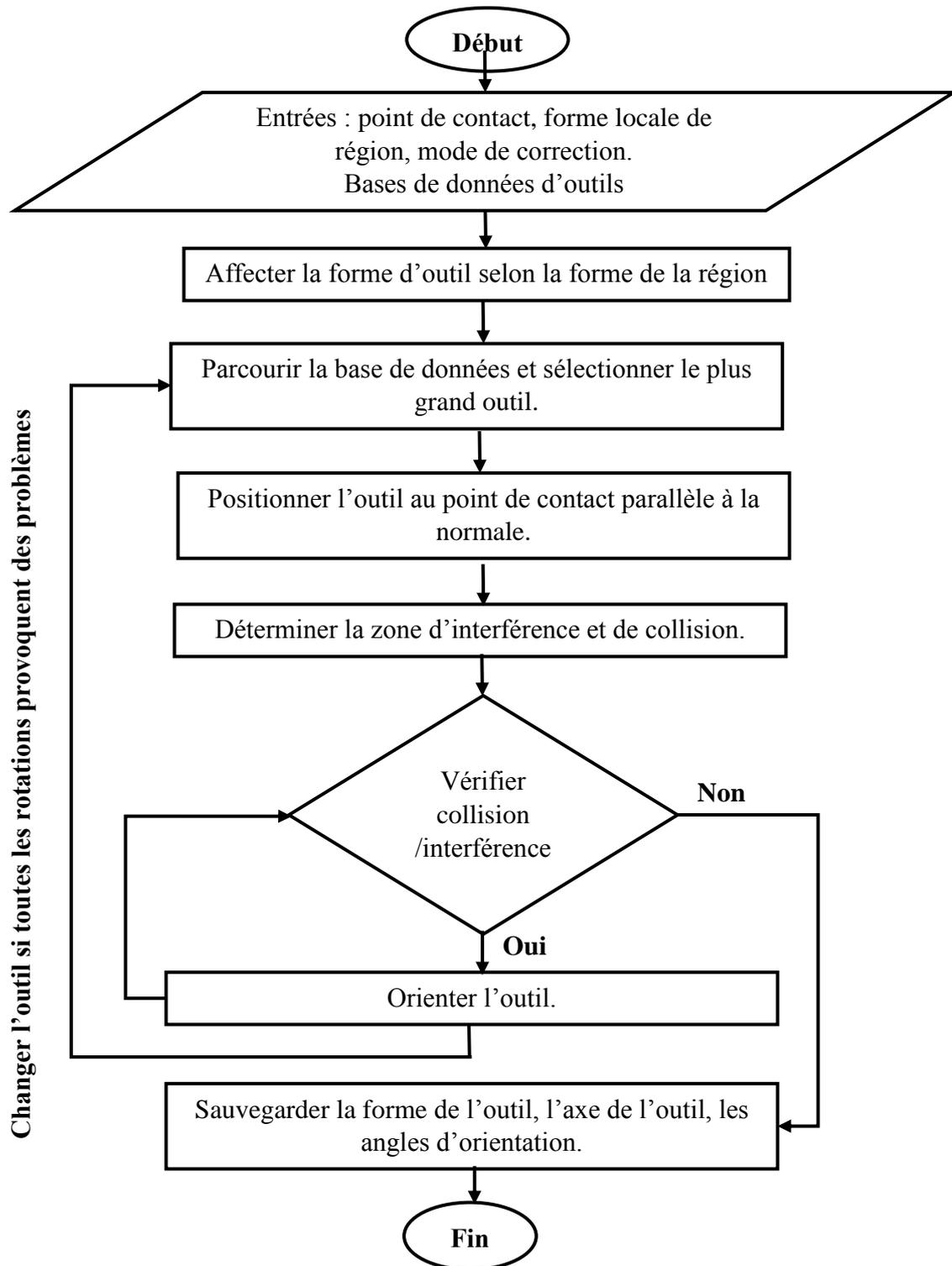
Donc, huit (08) combinaisons possibles :

- 1) Aucune rotation.
- 2) Rotation de l'outil autour de la normale avec un incrément dans un intervalle.
- 3) Inclinaison de l'outil vers le segment dans le sens de l'avance de l'outil avec un incrément dans un intervalle.
- 4) Rotation autour du segment avec un incrément dans un intervalle.
- 5) Rotation de l'outil autour de la normale et pour chaque angle « alpha » l'outil sera incliné vers le segment avec un angle « beta ».
- 6) Rotation de l'outil autour de la normale et pour chaque angle « alpha » l'outil subira une rotation autour du segment avec un angle « gamma ».
- 7) Inclinaison de l'outil vers le segment dans le sens de l'avance de l'outil et pour chaque variation de « beta » il y a une rotation de l'outil autour du même segment (formation de l'angle « gamma »).
- 8) Mode hybride : combinaison de (1), (2) et (3).

Après avoir orienté l'outil selon un mode, ce sommet peut être :

- **Usinable** : dans ce cas, il existe des angles « alpha », « beta » et « gamma » qui positionnent l'outil hors collisions et interférences.
- **Non usinable** : le mode de correction est testé pour toute la liste des outils et aucun ne positionne l'outil hors collisions et interférences.

Si le sommet est usinable, l'outil, son orientation et les angles « alpha », « beta » et « gamma » sont sauvegardés. L'Organigramme 5 résume toute cette démarche.



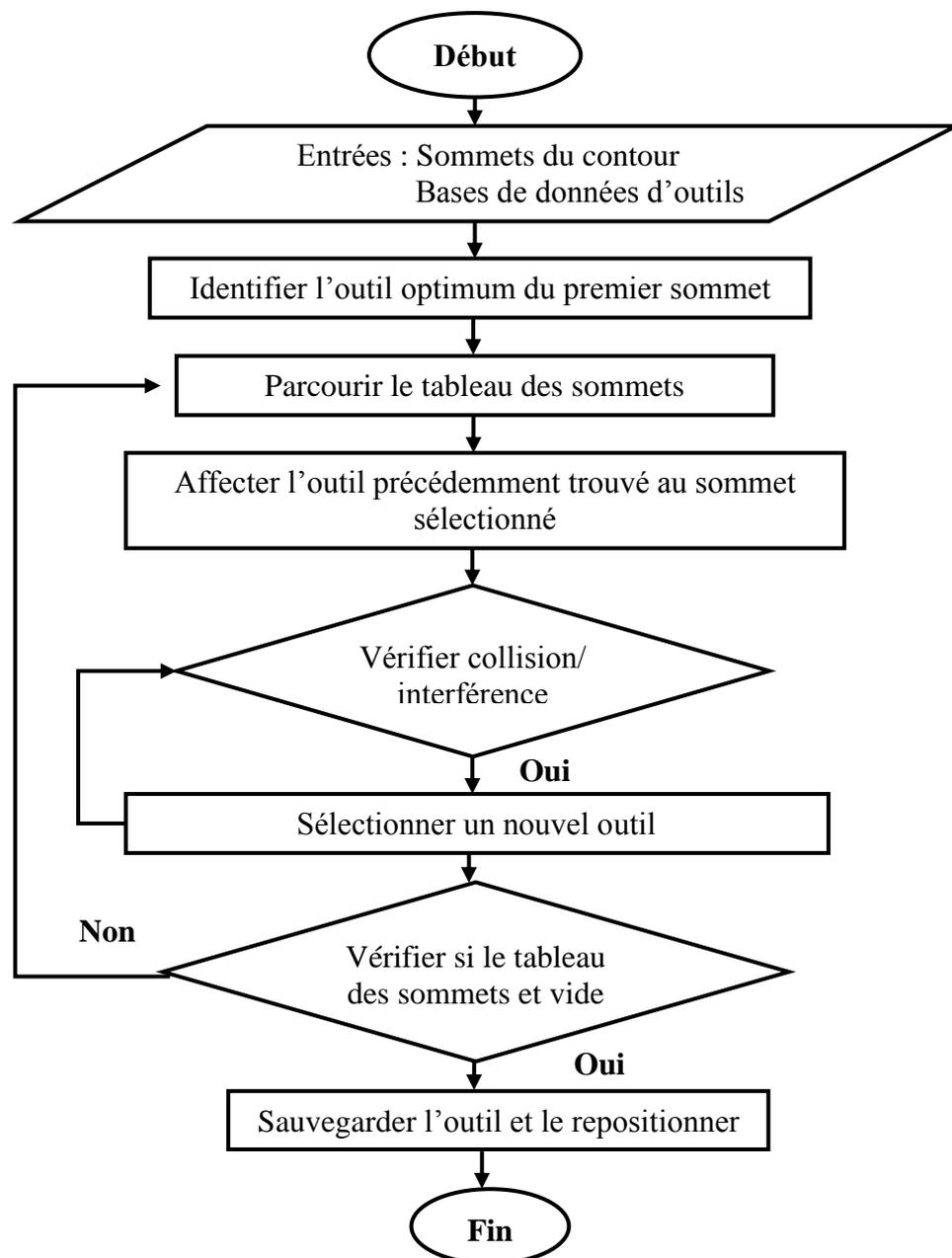
Organigramme 5. Affectation de l'outil optimum à chaque point de contact.

1.12 Affectation de l'outil optimum à chaque contour :

L'outil optimum d'un contour est celui qui élimine l'interférence et la collision en chaque point de ce dernier. Dans le contexte de ce projet, un seul outil est assigné à chaque contour de la manière suivante :

- a. Classer les listes contenant les trois formes d'outil selon un ordre décroissant par rapport au diamètre. Ensuite, la sélection commence par le plus grand outil.
- b. Parcourir la liste des sommets du contour et pour chaque sommet identifier l'outil optimum. Si cet outil est affecté au premier (1^{er}) sommet, alors il est, ainsi que son axe d'orientation et les angles de rotation sont affectés au sommet prochain et le test et refait. Dans le cas contraire, un nouvel outil est affecté au sommet prochain à partir de la liste d'outils établie (Organigramme 6).

Cette procédure est appliquée pour l'ensemble des points de chaque contour. Le dernier outil calculé est affecté au contour.



Organigramme 6. Outil optimum d'un contour.

1.13 Calcul du nombre d'outils à utiliser :

Afin d'optimiser le temps d'usinage d'une pièce de forme complexe en 05-axes, le nombre d'outils à utiliser est calculé en fonction du nombre d'outils du magasin d'outil, la forme locale et l'aire des régions. Pour cela, la démarche suivante est appliquée :

- Fixer le nombre total d'outils dans le magasin.
- Calculer l'aire totale de la surface à usiner.
- Récupérer l'aire des triangles frontières.
- Calculer l'aire de chaque région.
- Sommer l'aire des régions selon deux catégories de formes locales :
 - *Catégorie1* : concave, concave-développable et selle de cheval.
 - *Catégorie2* : convexe, convexe-développable et plane.
- Affecter les formes d'outils en fonction de la forme locale de la région :
 - *Catégorie1* : torique et hémisphérique.
 - *Catégorie2* : cylindrique et hémisphérique.

Ensuite, le nombre d'outils de chaque forme est estimé par la formule suivante :

$$\text{Nombre d'outils} = \% \left[\frac{(A_{reg} - A_{trf})}{A_t} \right] * N \quad (22)$$

Avec :

A_{reg} : aire des régions.

A_{trf} : aire des triangles frontières.

A_t : aire totale.

N : capacité du magasin d'outils.

Après le calcul du nombre d'outils à utiliser de chaque forme (cylindrique, torique, hémisphérique), il a été remarqué qu'il y'a certaines formes d'outils sont ignorées dans le calcul. Pour cette raison, une deuxième approche est proposée pour remédier à ce problème. Cette dernière est basée sur les outils optimums de la surface à usiner.

La formule suivante est appliquée pour estimer le nombre d'outils de chaque forme par rapport au nombre d'outils précédemment calculés.

$$\text{Nombre d'outils} = (Nbr_o * N) / Nbr_{total} \quad (23)$$

Avec :

Nbr_o : nombre d'outils utilisés (cylindrique, torique ou hémisphérique).

Nbr_{total} : nombre total d'outils utilisés.

N : capacité du magasin d'outils.

1.14 Sélection d'une combinaison optimale d'outils :

Après avoir identifié pour chaque contour un seul outil, stratégie visant à utiliser un nombre prédéfini d'outils afin d'optimiser le temps d'usinage et pour avoir une bonne qualité de surface est proposée en se basant sur les critères suivants :

- ✓ Diamètre de l'outil.
- ✓ Nombre d'utilisation de l'outil.

Si le nombre d'outils à utiliser pour usiner la totalité de la surface est égal à N , la procédure suivante est suivie pour déterminer N outils :

- a. Stocker dans une liste tous les outils affectés aux contours.
- b. Classer la liste selon un ordre décroissant par rapport au nombre d'utilisation de chacun.
- c. Affecter à la liste des outils optimisés le plus petit outil selon le diamètre.
- d. Affecter ($N-2$) outils à la liste des outils optimisés les outils les plus utilisés.
- e. Le dernier outil à affecter est le plus petit outil selon le rayon mais le plus utilisé.

Le pseudo-algorithme (Algorithme 1) détaille la démarche proposée afin de sélectionner la combinaison optimale d'outils pour la forme cylindrique.

Entrées : Tableau `tab_outil_utilise_cylindrique` de `nbr_outils_cyl` outils cylindriques utilisés, `nbre_cyl`.

Sorties : Tableau `outils_cyl_opt` de `nbre_cyl` outils cylindriques optimisés.

Variables : `Nbre`, `limite` : entier.

Début

`Nbre` ← `nbre_cyl` ;

Si(`nbre_cyl` >= 1)

Si(`nbr_outils_cyl` <= `nbre_cyl`) // pas d'optimisation

Pour `k` ← 0 à `nbr_outils_cyl` faire

Ajouter_outil(`outils_cyl_opt`, `tab_outil_utilise_cylindrique`[`k`]);

```

FinPour ;
Sinon
Ajouter_plus_petit_outil(outils_cyl_opt,tab_outil_utilse_cylindrique);
Nbre← Nbre-1 ;
quickSort(tab_outil_utilse_cylindrique,0,nbr_outils_cyl);//trie du tableau
Si (Nbre>=1)
    Limite=0 ;
    Si (Limite<Nbre-1)
        Si(tab_outil_utilse_cylindrique[m].getvaleur_inséré== false)
            Ajouter_outil(outils_cyl_opt,tab_outil_utilse_cylindrique[k]);
            Limite← Limite+1 ;
        FinSi ;
    FinSi ;
//insérer le plus petit des restants
Ajouter_plus_petit_outil(outils_cyl_opt,tab_outil_utilse_cylindrique);
FinSi ;
FinSi ;

```

Fin.

Algorithme 1. Sélection de la combinaison optimale d'outils.

1.15 Génération et simulation du trajet d'outils :

Après avoir déterminé les outils optimums de tous les contours de la pièce, il faut ajouter à chaque contour usinable quatre (04) points, deux points au début (p1 et p2) appelés points d'engagement calculés par rapport au premier sommet du contour et deux points à la fin (p3 et p4) appelés points de dégagement qui vont être calculés par rapport au dernier sommet du contour (Figure 14). Leurs coordonnées sont calculées par :

$$p_x = x + d * axe_x \quad (24)$$

$$p_y = y + d * axe_y \quad (25)$$

$$p_z = z + d * axe_z \quad (26)$$

Où :

x, y, z : coordonnées du sommet.

axe_x, axe_y, axe_z : composantes de l'axe d'outil propre au sommet.

d=d1 : pour p1 et p4.

$d=d_2$: pour p2 et p3.

La vitesse d'outil entre le point p1 et p2 est rapide puis lente jusqu'au point p3. Puis, elle redevient rapide.

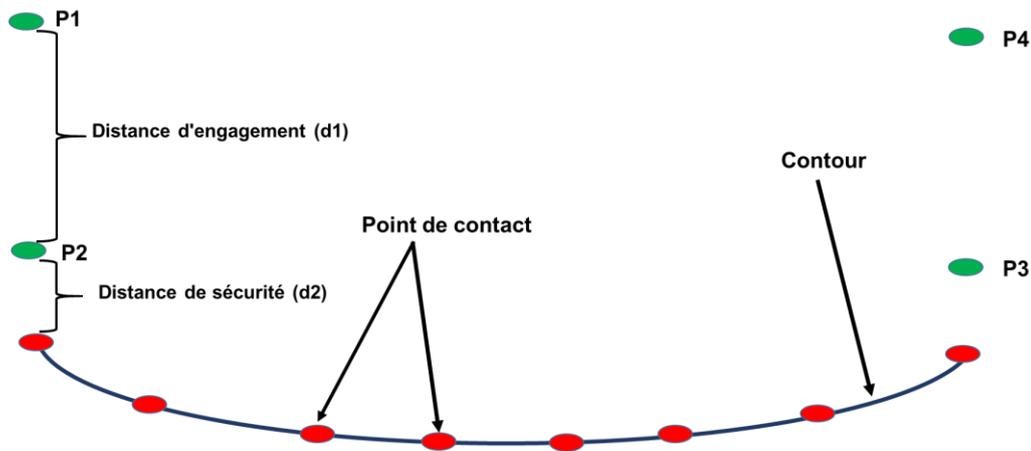


Figure 14. Points d'engagements/dégagement.

Pour réduire le nombre de changement d'outils, tous les contours ayant le même outil sont usinés séquentiellement.

La simulation du trajet d'outils consiste à visualiser graphiquement l'outil dans toutes les positions de passage de l'outil pour n'importe quelle orientation de son axe.

2. Modélisation UML :

Le langage UML «Unified Modeling Language » est un langage de modélisation orientée objet développé en réponse à l'appel à propositions lancé par l'OMG (Object Management Group) dans le but de définir la notation standard pour la modélisation des applications construites à l'aide d'objets. Il est la fusion de plusieurs autres langages tels qu'OMT (Object Modeling Technique) et OOSE (Object Oriented Software Engineering) et Booch. Les principaux auteurs de la notation UML sont Grady Booch, Ivar Jacobson et Jim Rumbaugh. C'est un langage visuel qui permet de modéliser le système par l'intermédiaire des diagrammes ou chacun donne une vision différente du logiciel à développer pour présenter son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées, etc.

Dans ce projet, trois (03) diagrammes sont utilisés : diagramme de cas d'utilisation, diagramme de séquence et diagramme de classe.

2.1. Diagramme de cas d'utilisation :

Les cas d'utilisation (Use Case) permettent de représenter le fonctionnement du système vis-à-vis l'utilisateur. Donc, ils permettent d'exprimer le besoin des utilisateurs d'un système.

Dans ce qui suit, les différentes fonctionnalités de notre système seront présentées. Le diagramme de cas d'utilisation générale qui donne une vue globale du fonctionnement du système est présenté en premier lieu.

- **Diagramme de cas d'utilisation général :** le système est divisé en dix grandes phases (Figure 15)

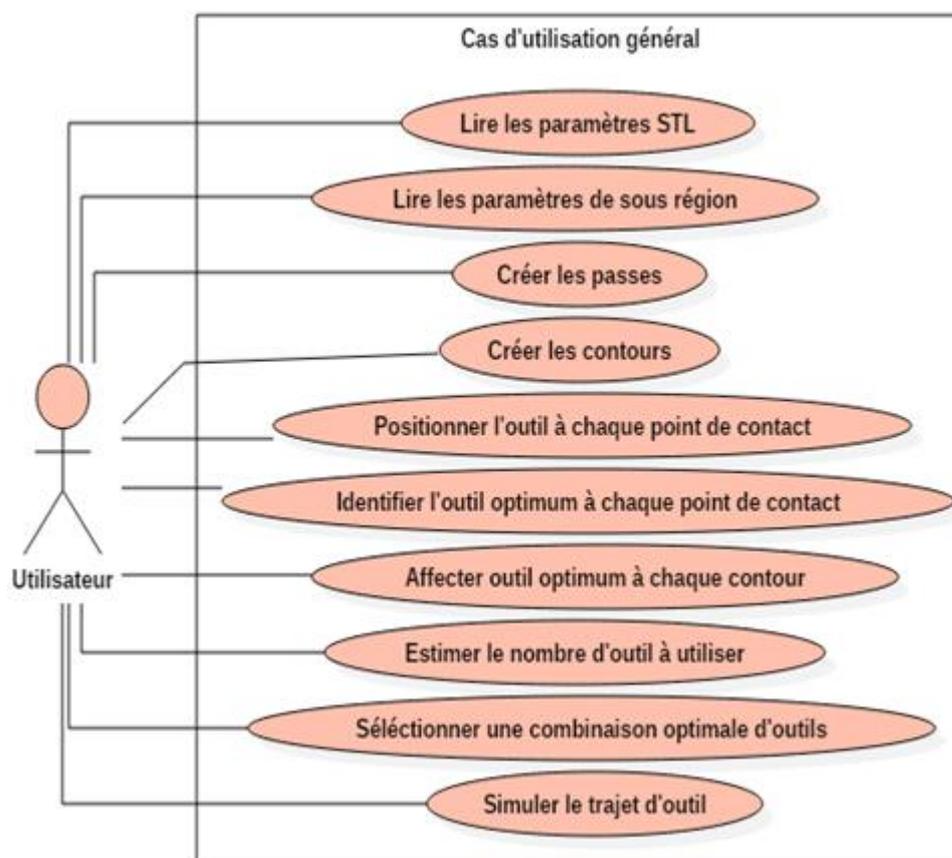


Figure 15. Diagramme de cas d'utilisation général.

2.2. Diagrammes de séquences :

Les diagrammes de séquences sont la représentation graphique des interactions entre les acteurs et le système selon un ordre chronologique dans la formulation UML (Unified Modeling Language). Le diagramme de séquence permet de montrer les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation. Dans un souci de simplification, l'acteur principal est représenté à gauche du diagramme,

et les acteurs secondaires éventuels à droite du système. Le but étant de décrire comment se déroulent les actions entre les acteurs ou objets.

Pour détailler les fonctionnalités principales de notre application, les diagrammes de séquences suivants sont élaborés.

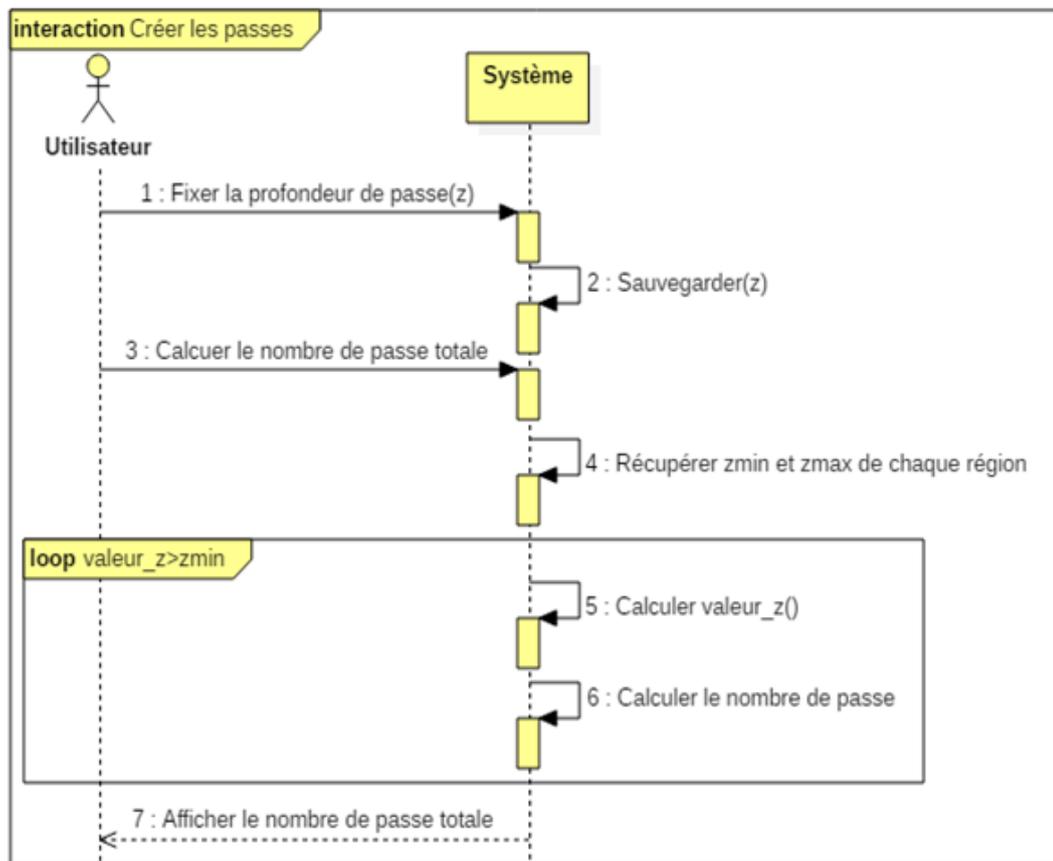


Figure 16. Création des passes.

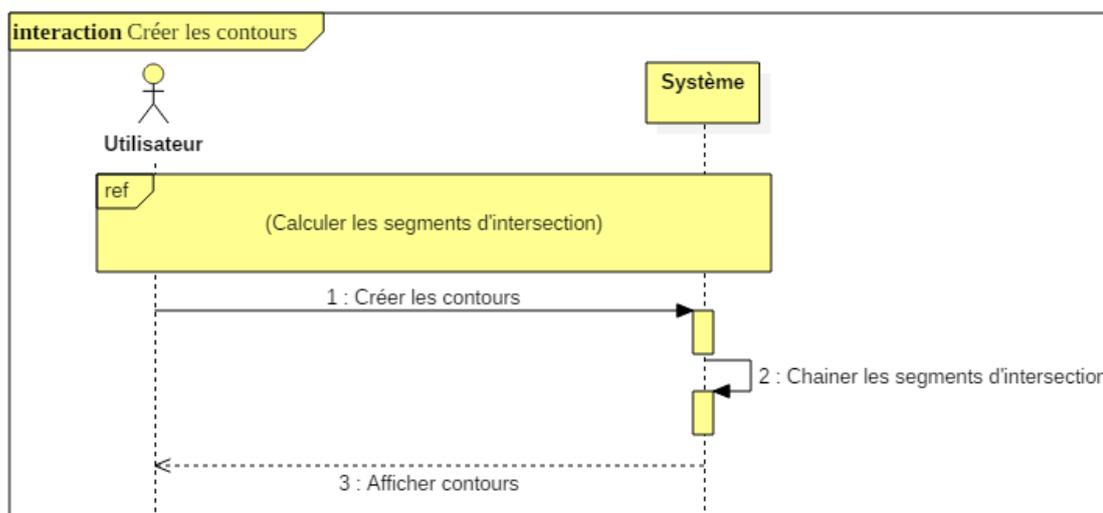


Figure 17. Création des contours.

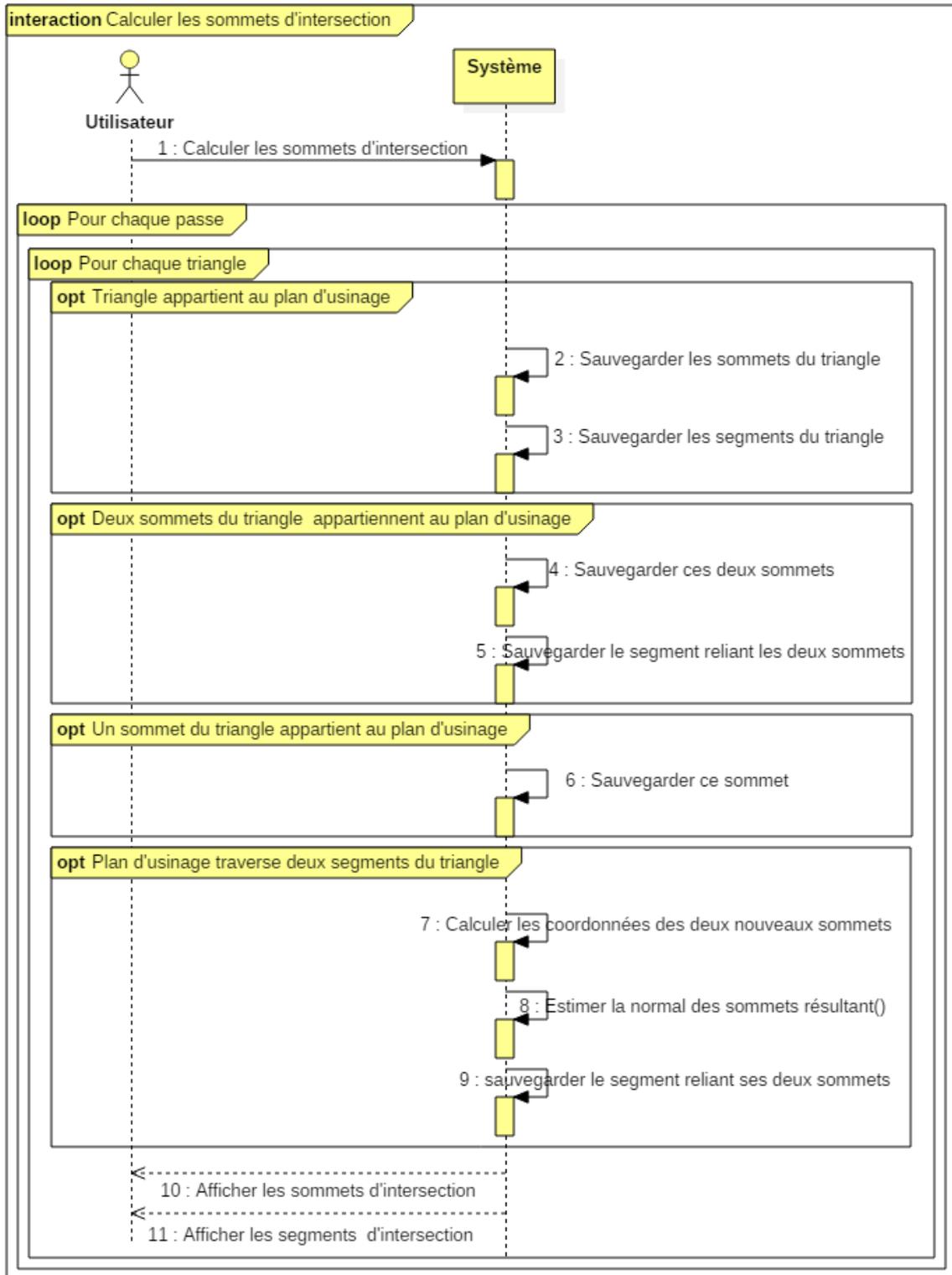


Figure 18. Détermination des sommets et des segments d'intersection.

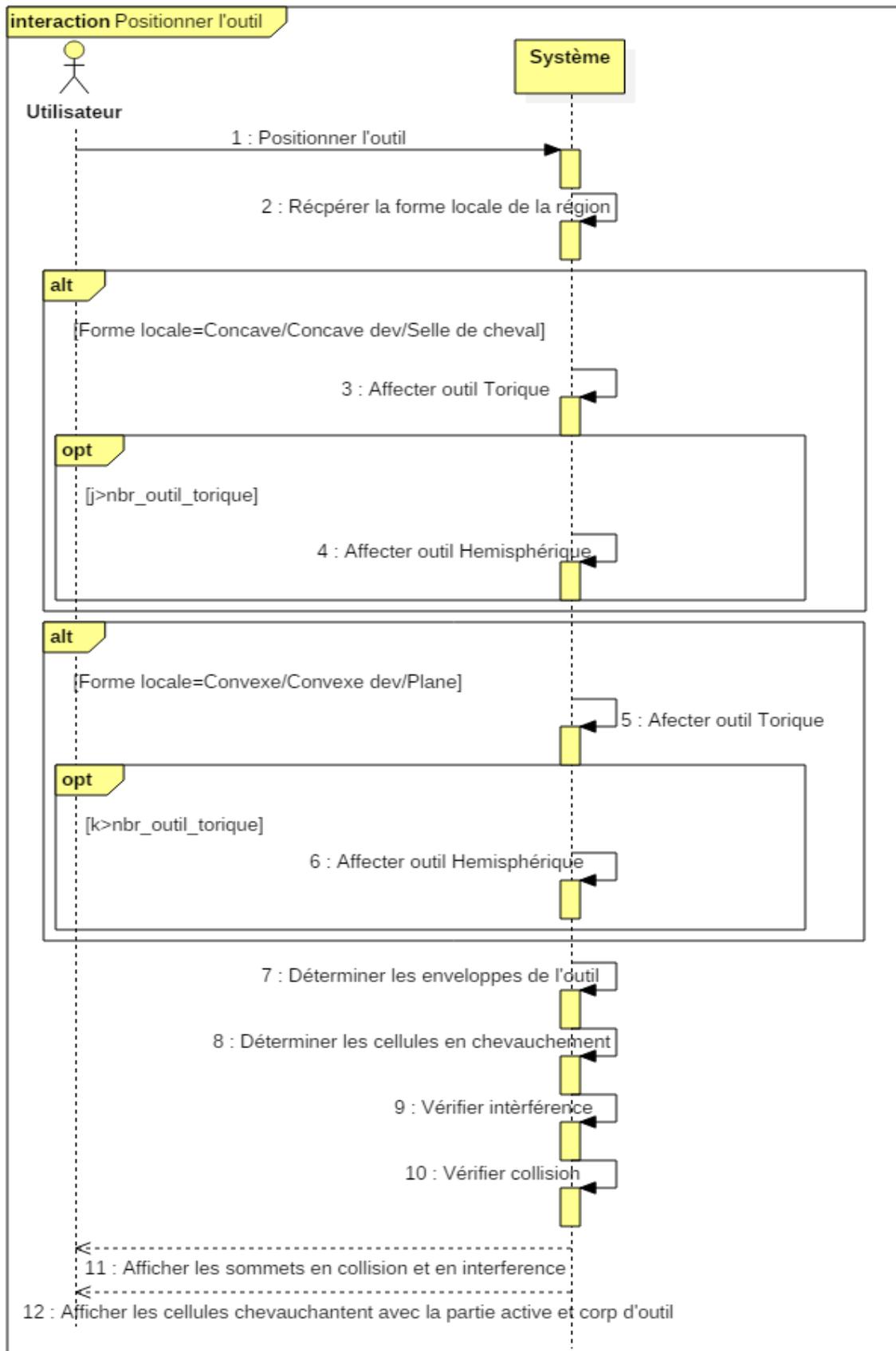


Figure 19. Positionnement de l'outil.

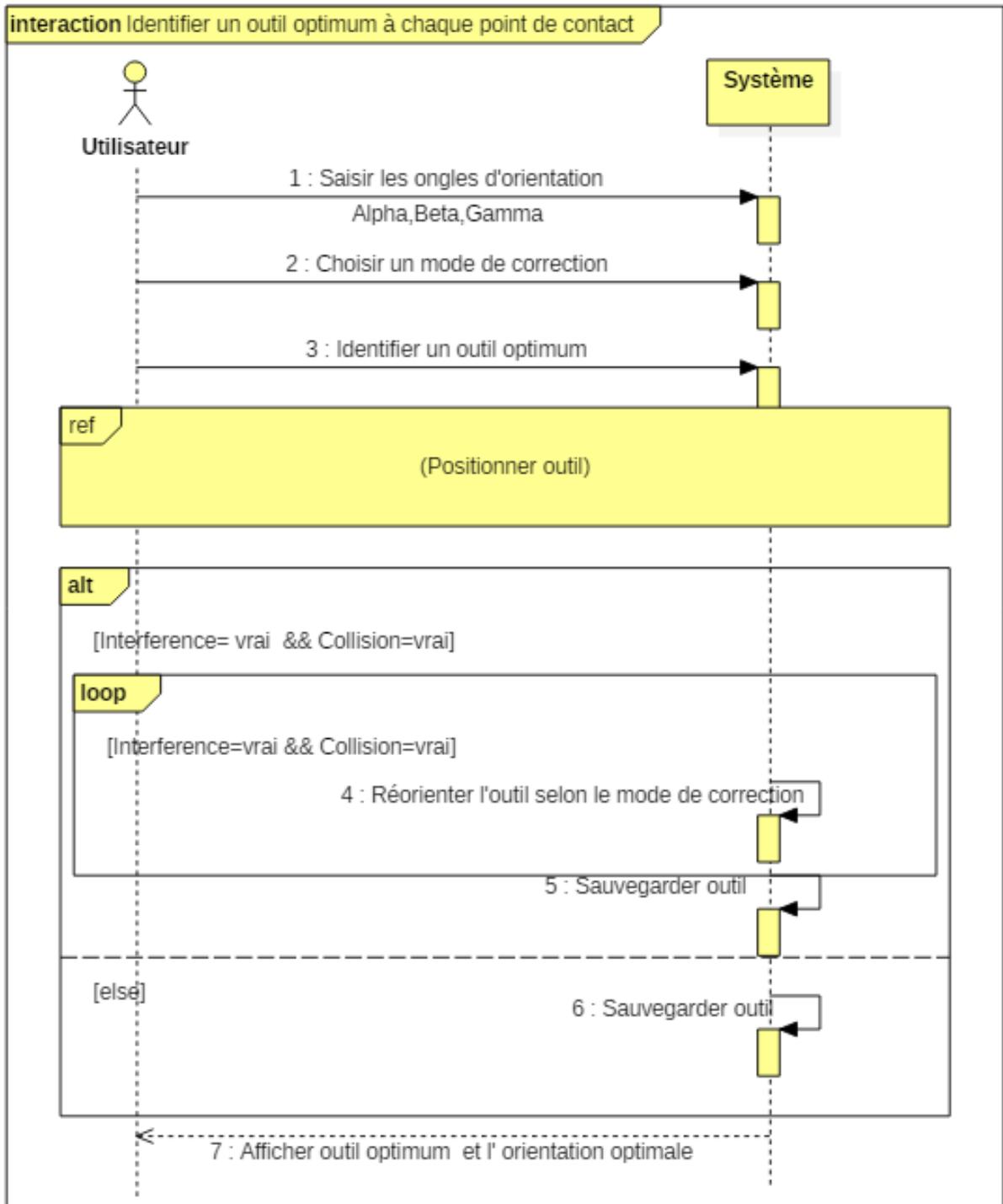


Figure20. Identification de l'outil optimum en un point de contact.

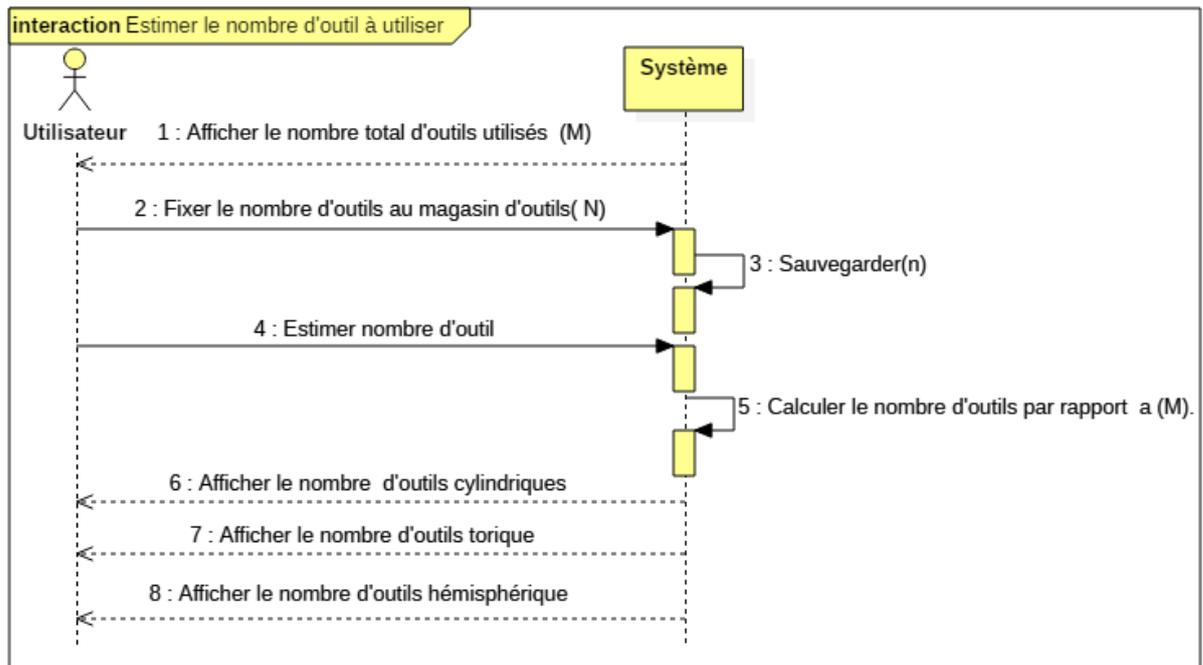


Figure21. Calcul du nombre d'outils pour chaque forme d'outil.

2.3. Diagramme de classes :

Un diagramme de classes décrit le type des objets ou données du système ainsi que les différentes formes de relations statiques qui les relient entre eux. Classiquement, ils existent deux principaux types de relations entre objets :

- Les associations, bien connues des vieux modèles entité/association utilisés dans la conception des bases de données depuis les années 70.
- Les sous-types, particulièrement en vogue en conception orientée objets, puisqu'ils s'expriment très bien à l'aide de l'héritage en programmation

Dans ce qui suit, le diagramme de classes global développé est présenté en tenant compte des relations entre les classes (Figure 22). Ensuite, les principales classes utilisées sont détaillées.

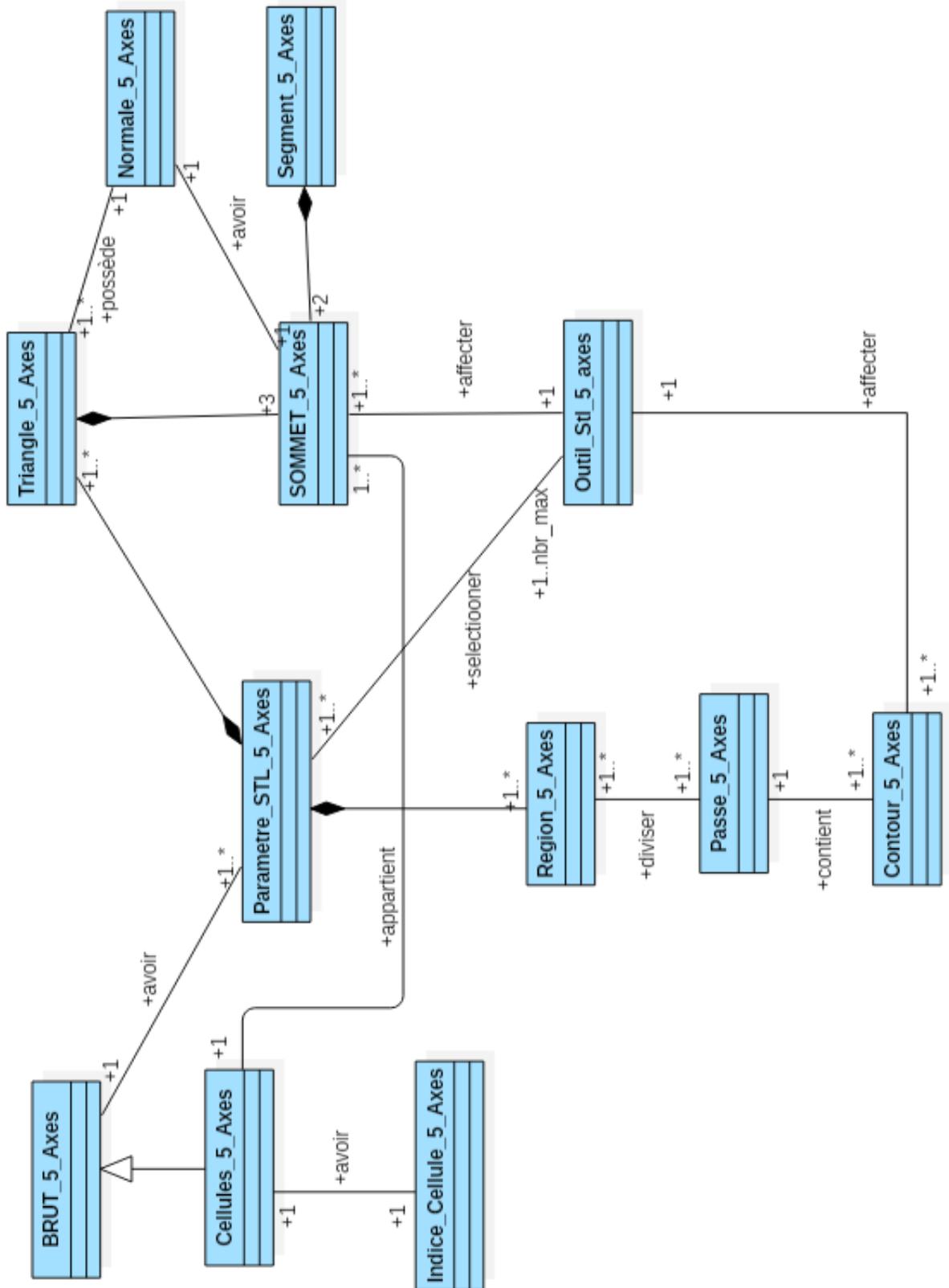


Figure 22. Diagramme de classes.

Les principales classes utilisées sont :

- ✓ **Classe « Parametre_STL_5_Axes »** : c'est la classe principale qui regroupe tous les sommets, triangles du fichier STL, les cellules créées ainsi que les outils (outils utilisés dans toutes les régions et les outils optimums). Les méthodes de cette classe permettent de récupérer les principales informations du fichier STL, les outils de la base de données des outils et des régions (Figure 23).

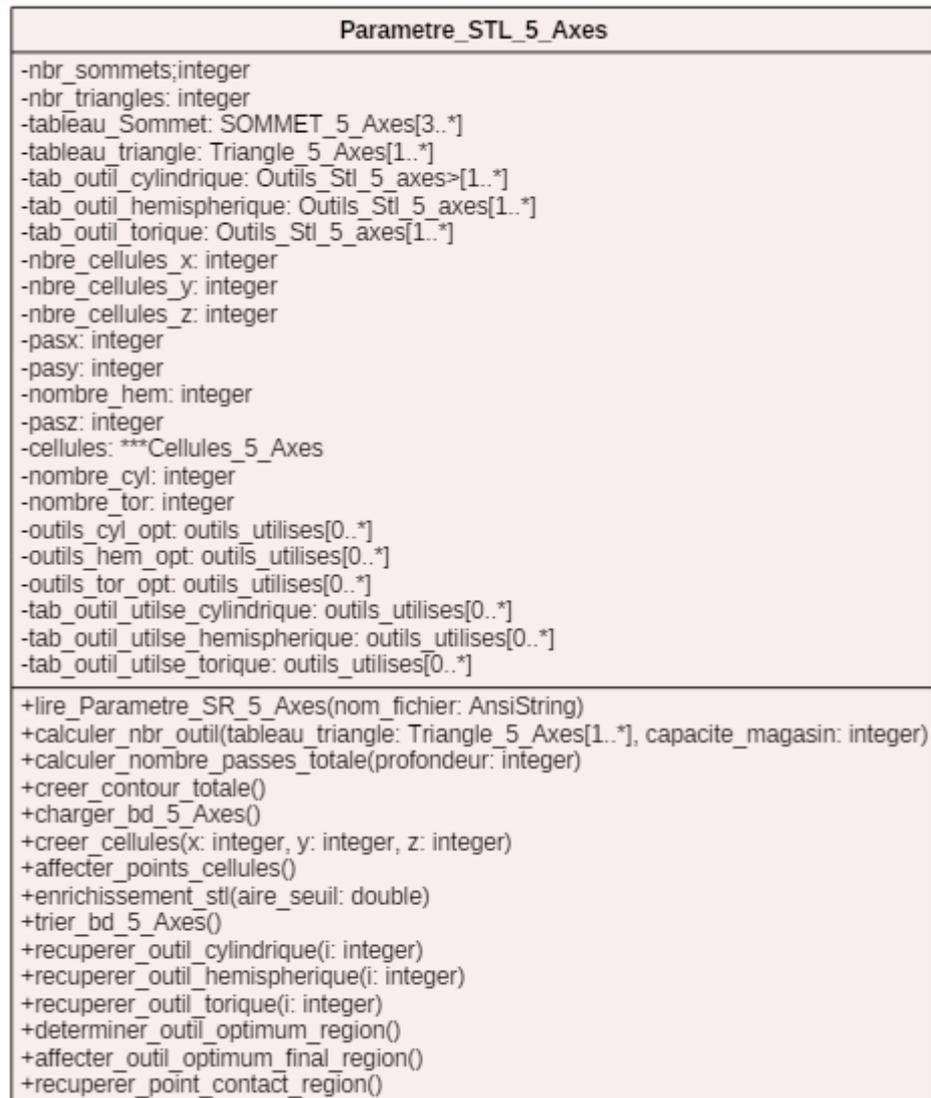


Figure 23. Classe « Parametre_STL_5_Axes ».

- ✓ **Classe « SOMMET_5_Axes »** : c'est la classe qui définit un sommet par ses différentes caractéristiques (Figure 24). Les méthodes de cette classe permettent de déterminer les différents paramètres du sommet.

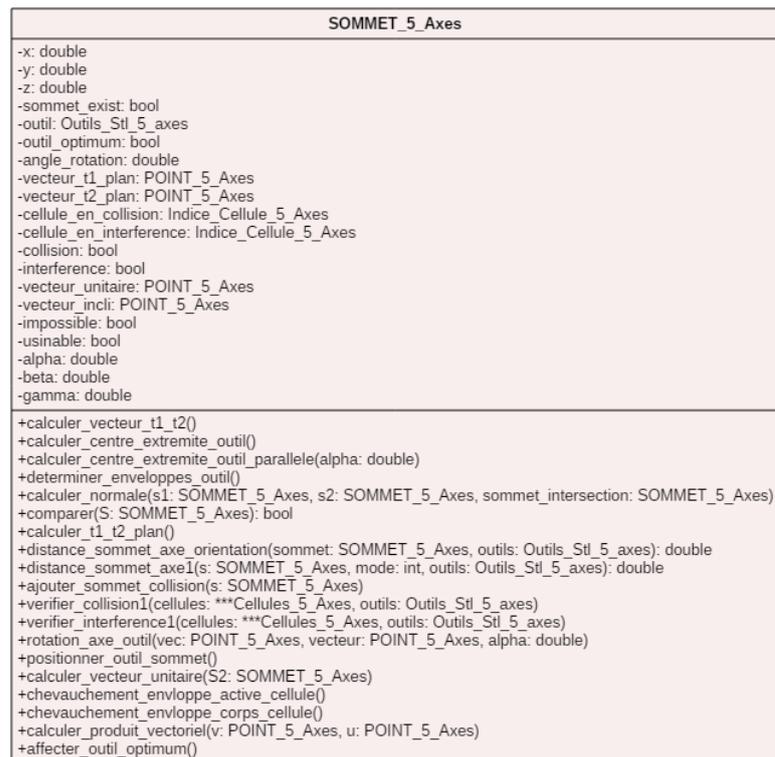


Figure 24. Classe « SOMMET_5_Axes ».

- ✓ **Classe « Triangle_5_Axes »:** c'est une classe qui englobe les différentes caractéristiques des triangles (Figure 25). Ses fonctions sont données pour enrichir un triangle.

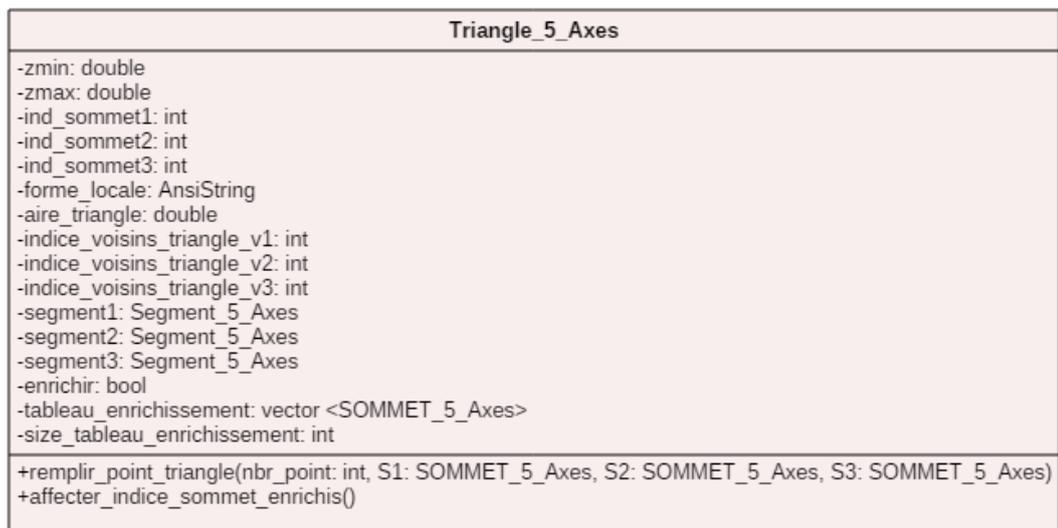


Figure 25. Classe « Triangle_5_Axes ».

- ✓ **Classe « Outil_5_axes »** : elle représente un outil récupéré depuis une base de données (Figure 26). Les fonctions de cette classe sont implémentées afin de déterminer les différentes caractéristiques de l'outil.

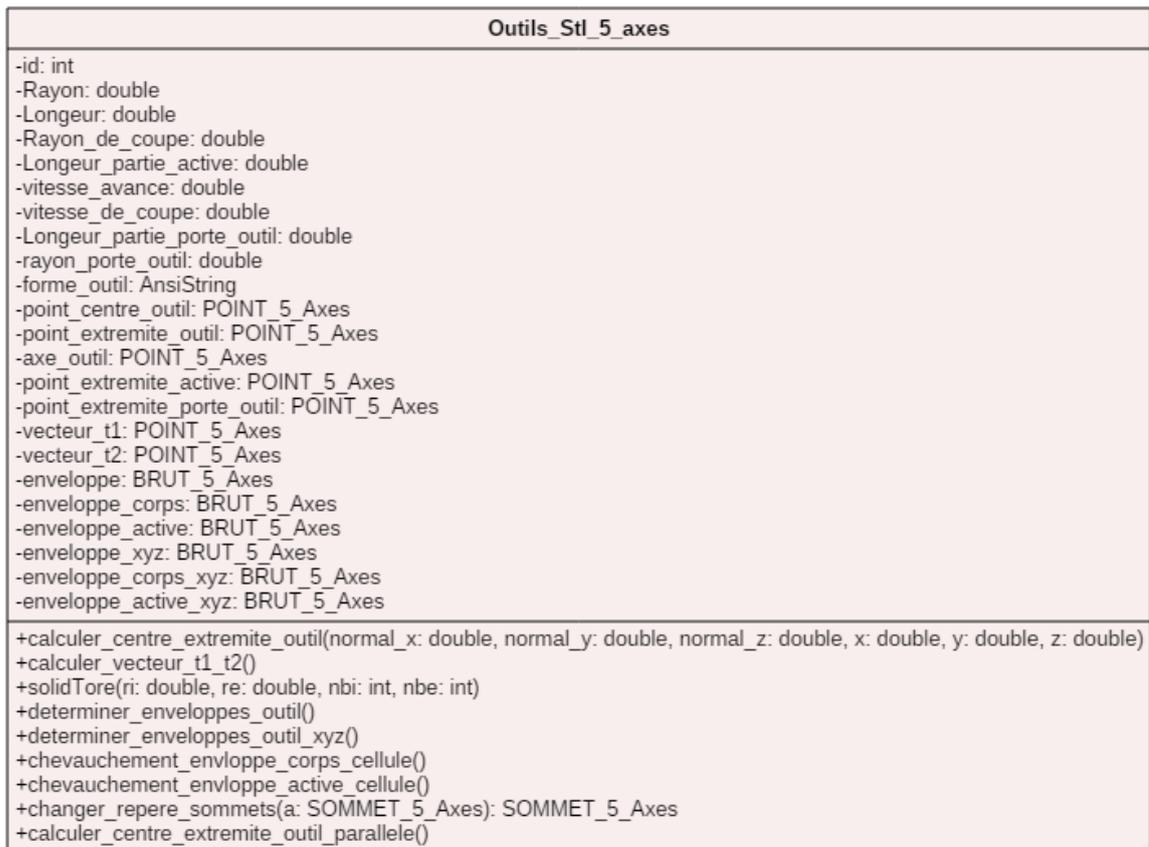


Figure 26. Classe « Outil_5_axes ».

- ✓ **Classe « Cellule_5_Axes »** : représente une cellule par un vecteur d'adresse de sommets (Figure 27) et par deux variables booléennes qui indiquent si cette cellule est en interférence ou en collision. Elle hérite de la classe « brut » comme montre le diagramme global. C'est-à-dire elle prend tous les attributs et méthodes de la classe mère.

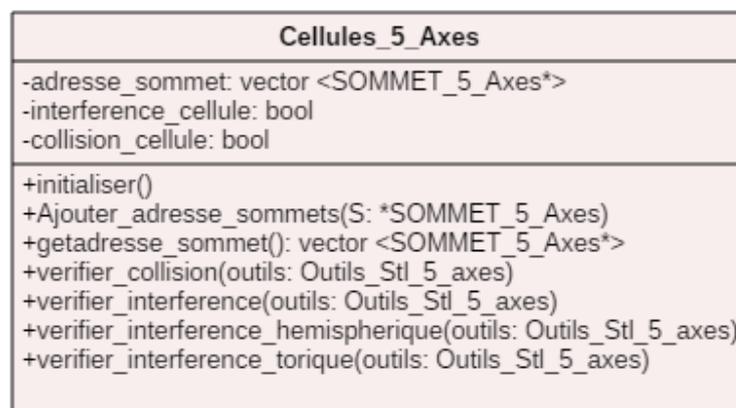


Figure 27. Classe « Cellule_5_Axes ».

- ✓ **Classe « Region_5_Axes »** : elle représente les différentes informations d'une région. Ses attributs et fonctions sont donnés par la Figure 28.

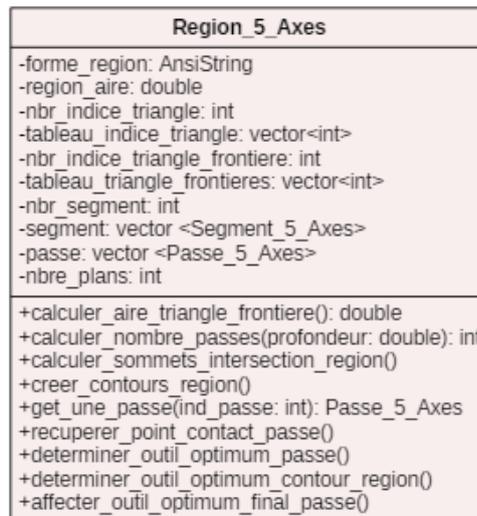


Figure 28. Classe « Region_5_Axes ».

- ✓ **Classe « Contour_5_Axes »** : c'est une classe qui englobe les différentes caractéristiques d'un contour (Figure 29).

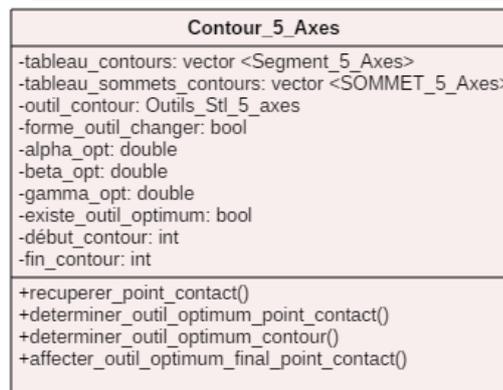


Figure 29. Classe « Contour_5_Axes ».

- ✓ **Classe « Passe_5_Axes**» : elle représente les différentes informations d'une passe (Figure 30).

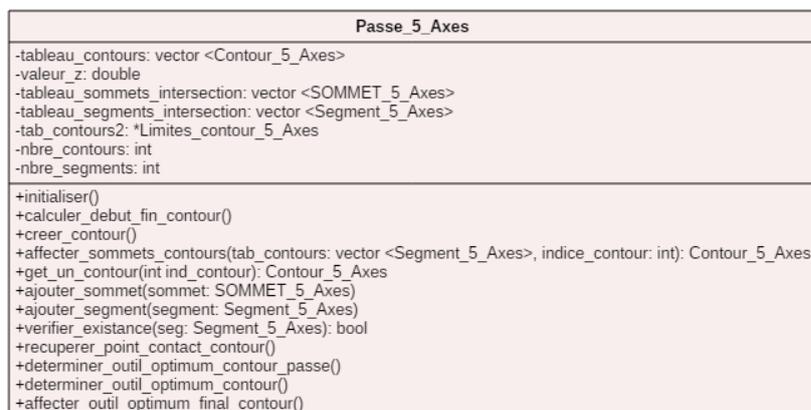


Figure30. Classe « Passe_5_Axes ».

Conclusion :

Dans ce chapitre, nous avons présenté l'architecture globale de notre application logicielle et les problèmes rencontrés avec les différentes solutions proposées pour pouvoir générer un trajet d'outil sans interférences et sans collisions lors de la finition des surfaces complexes définies par des modèles STL sur des fraiseuses numériques 05-axes en utilisant trois formes d'outils cylindrique, torique, hémisphériques.

Le test et la validation de notre application sur des surfaces réelles seront considérés dans le chapitre suivant

CHAPITRE 3

IMPLEMENTATION INFORMATIQUE ET VALIDATION

Introduction.

1. Présentation des langages utilisés.
2. Présentation de l'environnement de « CFA0 ».
3. Présentation de l'application.
4. Test et validation.

Conclusion.

Introduction :

Après avoir présenté la conception de notre système dans le deuxième chapitre, l'objectif de ce chapitre est l'illustration des différentes tâches que notre système peut effectuer. Nous commençons par définir les outils de développement de notre application ainsi qu'une vue générale détaillée en utilisant des captures d'écrans. Par la suite, une validation est réalisée par un ensemble de tests qui ont été préparés pour chaque fonctionnalité de l'application.

1. Présentation des langages utilisés :

Comme notre système est dédié à être intégré dans l'environnement de production des surfaces complexes développé par l'équipe « CFAO » du « CDTA », les outils de développement utilisés lors de la mise en œuvre de notre application (C++, OpenGL) sont les mêmes qu'utilisés par l'équipe « CFAO » pour se conformer à la tendance qui veut que la majorité des systèmes de « CFAO » sont développés par C++ et OpenGL.

1.1. Présentation du langage C++ :

Le langage C++, inventé par Bjarne Stroustrup vers 1983, est une évolution orientée objets du langage C de Brian Kernighan et Denis Ritchie. Il s'est enrichi, au cours de la décennie 1980, parallèlement à la stabilisation et la normalisation de C. Ce langage repose sur les mêmes mécanismes d'écriture et de génération. Il apporte notamment la gestion des exceptions, la gestion des références, la surcharge des opérateurs et les Templates, ...etc. Enfin, une rétrocompatibilité a été gardée où les programmes en C sont compilés sans difficulté avec un compilateur C++.

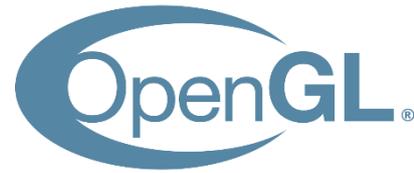


Comme tout langage, C++ dispose d'une bibliothèque standard, c'est-à-dire de fonctions et de classes prédéfinies. Elle comporte notamment de nombreux patrons de classes et de fonctions permettant de mettre en œuvre les structures de données les plus importantes (vecteurs dynamiques, listes chaînées, chaînes, ...etc.) et les algorithmes les plus usuels.

Parmi les environnements de développement AnjutaDevStudio, C++ Builder, Code :: Blocks (open-source), Dev-C++, Eclipse (open-source), Microsoft Visual C++, ...etc. [1].

1.2. Présentation d'OpenGL :

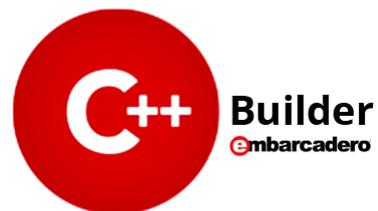
OpenGL (Open Graphics Library) est un ensemble normalisé de fonctions de calcul d'images 2D ou 3D lancé par Silicon Graphics en 1992. Cette interface de programmation est disponible sur de nombreuses plateformes. Elle est utilisée pour des applications allant du jeu vidéo jusqu'à la CAO en passant par la modélisation.



OpenGL permet à un programme de déclarer la géométrie d'objets sous forme de points, de vecteurs, de polygones et de textures. OpenGL effectue ensuite des calculs de projection en vue de déterminer l'image à l'écran, en tenant compte de la distance, de l'orientation, des ombres, de la transparence et du cadrage. L'interface regroupe environ 250 fonctions différentes qui peuvent être utilisées pour afficher des scènes tridimensionnelles complexes à partir de simples primitives géométriques. Du fait de son ouverture, de sa souplesse d'utilisation et de sa disponibilité sur toutes les plates-formes, elle est utilisée par la majorité des applications scientifiques, industrielles ou artistiques 3D et certaines applications 2D vectorielles. Cette bibliothèque est également utilisée dans l'industrie du jeu vidéo où elle est souvent en rivalité avec la bibliothèque de Microsoft Direct3D [2].

1.3. Présentation d'Embarcadero C++ Builder 10 Seattle :

C++ **Builder** est un logiciel de développement rapide d'applications « RAD » conçu par Borland qui reprend les mêmes concepts, la même interface et la même bibliothèque que Delphi en utilisant le langage C++. Il permet de créer rapidement des applications Win32 et Win64 ainsi qu'une interface graphique avec son éditeur de ressources. Il est compatible avec la version de norme ISO C++ de 2011.



Embarcadero C++ Builder 10 Seattle est le moyen le plus rapide de créer et de mettre à jour des applications riches en données, hyper connectées et visuellement engageantes pour Windows 10, Mac, Mobile, IoT et plus encore en utilisant le standard C++. Mettez à jour rapidement et facilement les applications VCL et FMX vers Windows [3].

2. Présentation de l'environnement de « CFAO » :

C'est une plateforme logicielle dédiée à la production numérique des pièces complexes sur fraiseuses numériques 03-axes et 05-axes. Cette plateforme est composée de deux parties : lancement des fonctionnalités et visualisation (Figure 1). Le module logiciel que nous avons développé est intégré à cette plateforme et la Figure 2 montre comment le lancer et l'exécuter.

Partie 2: visualisation

Partie 1: lancement des fonctionnalités

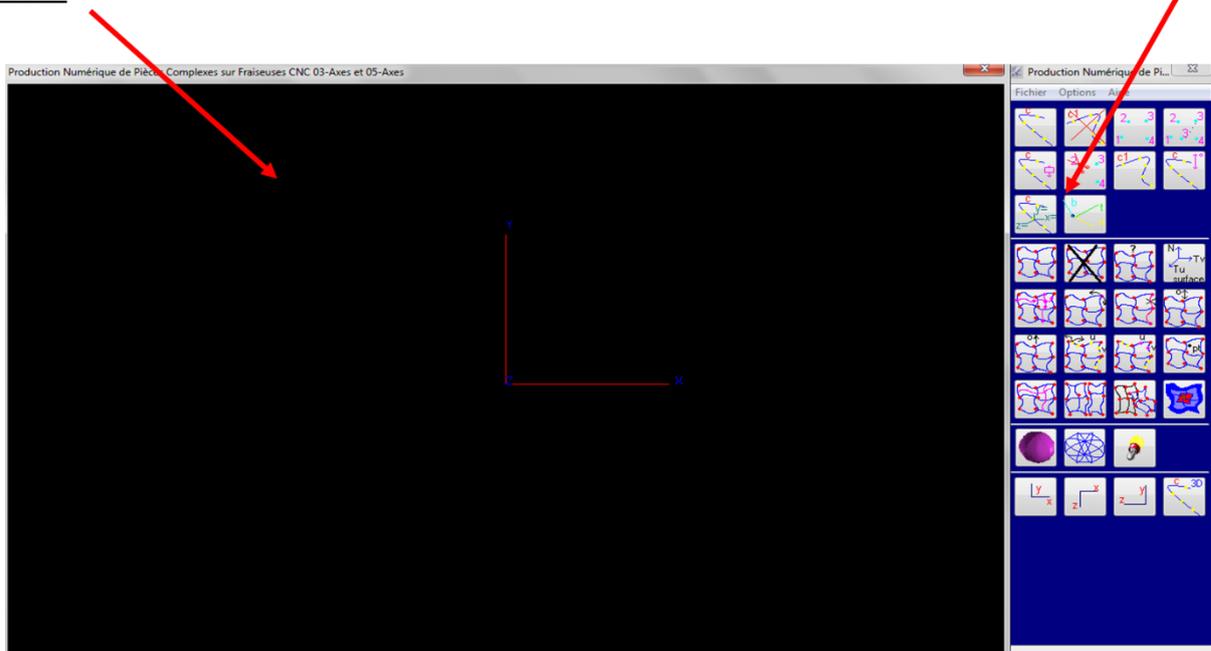


Figure 1. Environnement de « CFAO ».

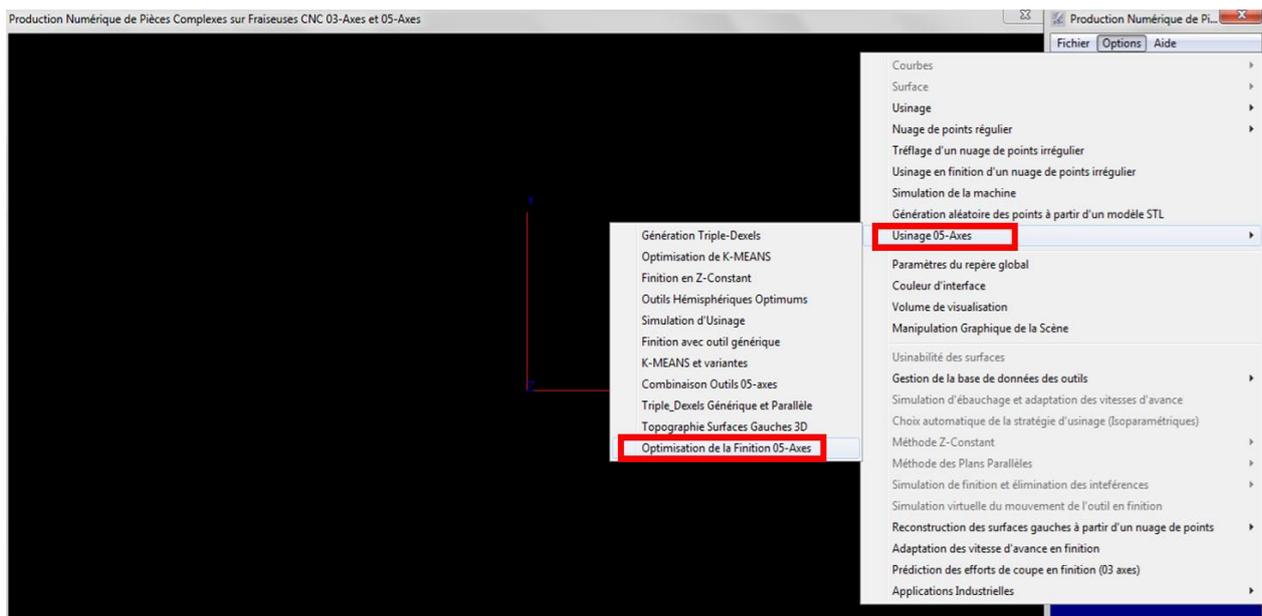


Figure 2. Lancement du module logiciel développé.

3. Présentation de l'application :

L'application développée est composée de plusieurs fonctionnalités complémentaires. Elle est composée de huit (08) onglets (Figure 3).

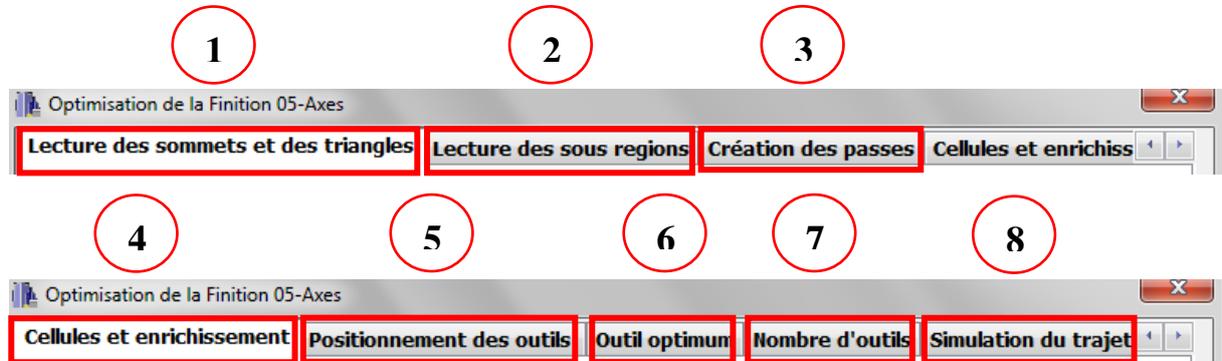


Figure 3. Onglets de l'application développée.

La description détaillée de chaque onglet est réalisée dans les paragraphes suivants.

3.1. Lecture des paramètres STL :

Le premier onglet « **Lecture des sommets et des triangles** » permet la lecture du fichier texte contenant l'ensemble des paramètres des sommets (les coordonnées du sommet et la normale) et des triangles (la normale, l'aire, la forme locale et les indices des sommets).

Par un simple clic sur le bouton « **Lire paramètre STL** » et sélection du fichier à ouvrir, le brut minimum est calculé (Figure 4).

Une fois le fichier est lu, les informations suivantes peuvent être récupérées :

- Dimensions minimales du brut.
- Nombre total de sommets et de triangles.

En plus de ces informations, il est possible de visualiser :

- Brut du modèle en deux modes filaire et rendu.
- Triangles du modèle en deux modes filaire et rendu.
- Sommets du modèle.
- Formes des triangles.
- Normales des sommets du modèle.
- Normales des triangles du modèle.

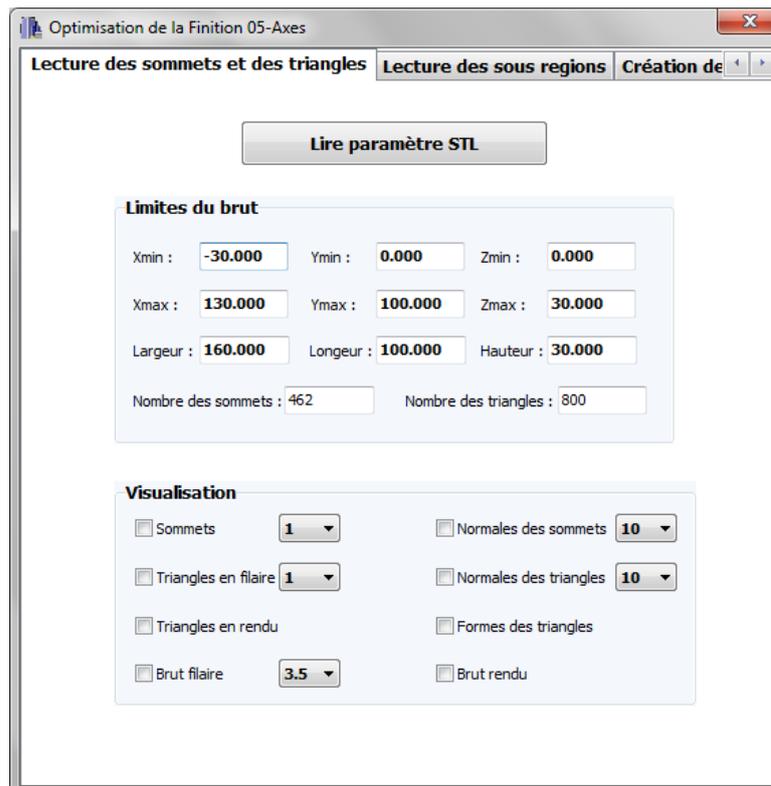


Figure 4. Onglet « Lecture des sommets et des triangles ».

3.2. Lecture des paramètres des sous régions :

Le deuxième onglet « **Lecture des paramètres des sous régions** » permet la lecture du fichier texte contenant l'ensemble des paramètres des sous régions. Par un simple clic sur le bouton « **Lire paramètres sous régions** » et sélection du fichier à ouvrir, les paramètres suivants sont affichés pour chaque sous-région : l'indice, la forme locale, les dimensions minimales de brut, l'aire et le nombre de triangles (Figure 5).

En plus de ces informations, il est possible de visualiser :

- Les sous régions (global et partiel).
- Le brut en filaire de chaque sous-région (global et partiel).
- Les segments frontières (global et partiel).
- Les triangles frontières (global et partiel).

Une fois les paramètres STL et les paramètres des sous régions sont récupérés, l'utilisateur passe au troisième onglet pour la création des passes d'usinage.

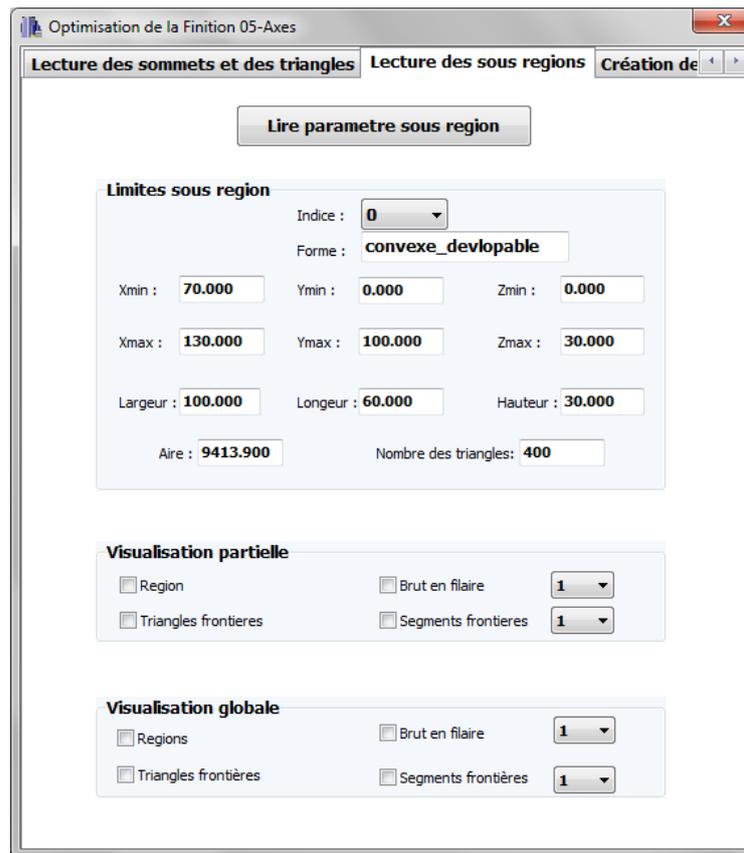


Figure 5. Onglet « Lecture des paramètres des sous régions ».

3.3. Création des passes:

Cet onglet sert à subdiviser la pièce par des plans horizontaux représentant les plans de coupe. Pour cela, l'utilisateur fixe la profondeur de passe, ensuite il clique sur le bouton « **Calculer nombre de passe totale** ». Une fois le traitement est terminé, le nombre de passes total est affiché.

Après avoir créé les passes, en cliquant sur le bouton « **Calculer les sommets d'intersections** », les sommets et les segments d'intersection des passes avec les triangles du modèle sont calculés.

Ensuite, l'utilisateur clique sur le bouton « **Créer contours** » pour chaîner les segments d'intersections issus dans le bouton précédent (Figure 6).

De même, l'utilisateur peut visualiser selon l'indice de la sous régions et l'indice de passe par une visualisation partielle ou bien globale les informations suivantes :

- Nombre de contours par passe.
- Régions.
- Plans d'usinages (passes).
- Sommets d'intersections.

- Segments d'intersections.
- Normale aux sommets d'intersections.
- Contours.
- Chaque contour avec couleur.

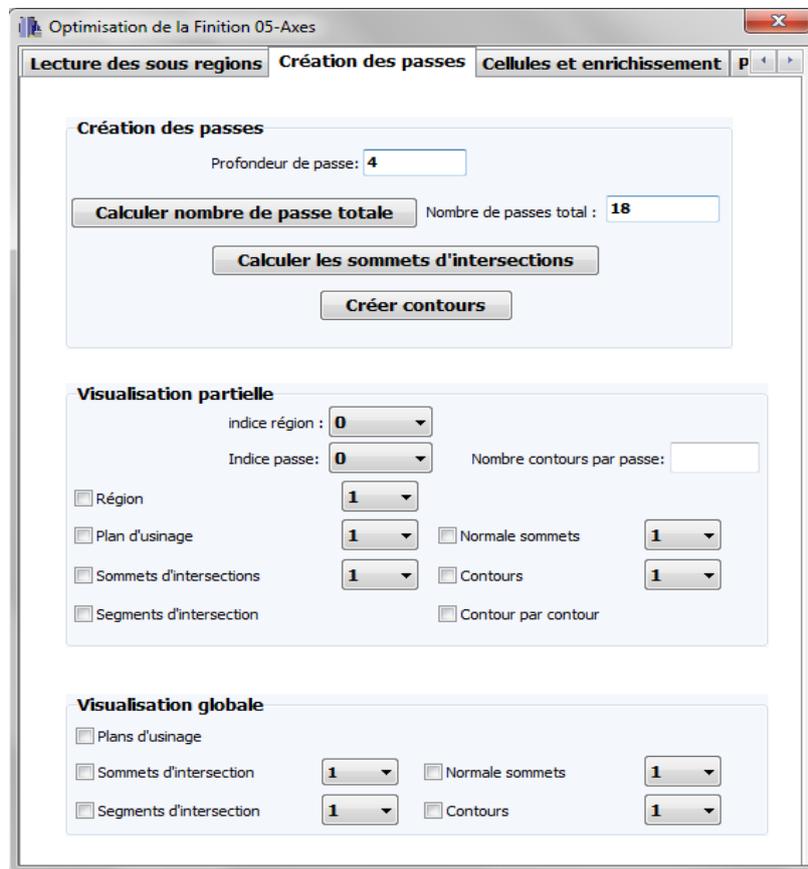


Figure 6. Onglet « création des passes ».

3.4. Création des cellules et enrichissement :

Afin de restructurer le modèle STL, l'utilisateur doit effectuer quatre étapes importantes montrées dans la Figure 7.

La première étape consiste à créer les cellules par la spécification du nombre de cellules suivant les trois axes X, Y et Z et un clic sur le bouton «**Créer les cellules**».

La deuxième étape consiste à déterminer les triangles à enrichir. Pour cela, l'utilisateur fixe le seuil de l'aire minimum et l'aire maximum affichée par le système et un clic sur le bouton «**Triangle enrichis**».

La troisième étape consiste à enrichir le modèle STL par l'ajout d'un ensemble de points pour chaque triangle. Pour cela, l'utilisateur fixe la densité des points à ajouter et clic sur «**Enrichissement du modèle STL**».

La dernière étape consiste à affecter les nouveaux sommets aux cellules par un simple clic sur le bouton « **Affecter point aux cellules** ».

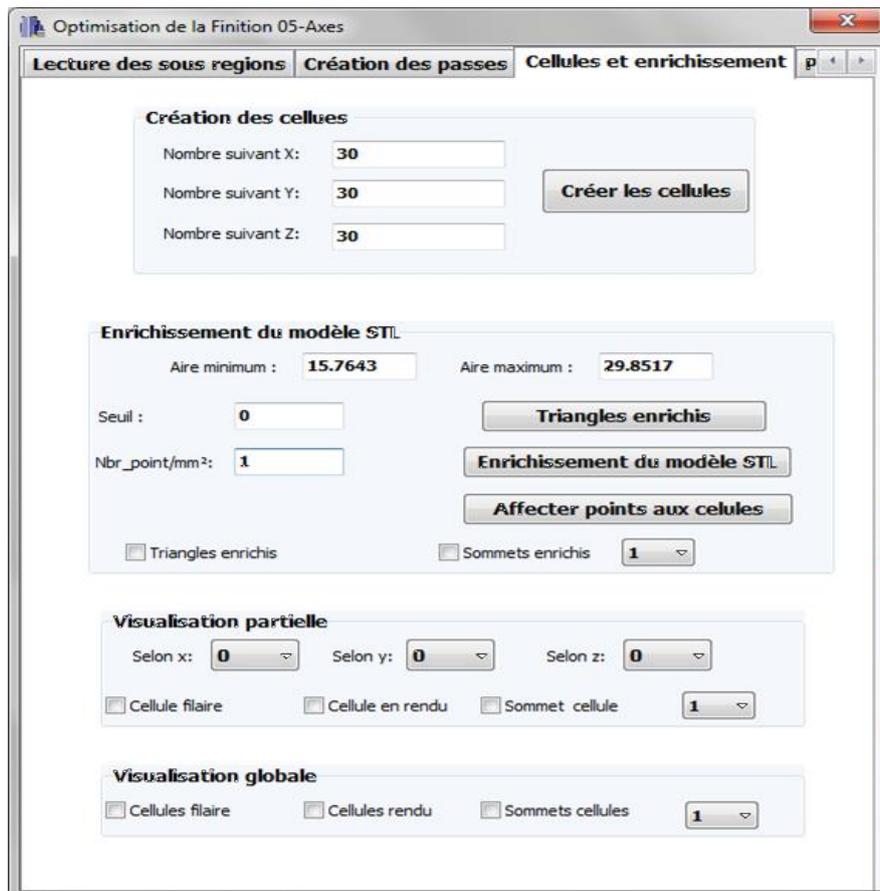


Figure 7. Onglet « Cellules et enrichissement ».

L'utilisateur peut visualiser les informations suivantes :

- Cellules en deux modes filaire et rendu.
- Triangles enrichis.
- Sommet ajoutés.
- Sommets des cellules.

3.5. Positionnement des outils :

Cet onglet sert à positionner une forme d'outil pour chaque point de contact de la sous-région, plan et contour en fonction du choix de l'utilisateur (Figure 8).

Le positionnement se fait selon le mode choisi par l'utilisateur.

- **Mode 1** : l'utilisateur fixe l'axe d'orientation d'outil.
- **Mode 2** : l'axe d'orientation d'outil parallèle à la normale.

Il peut aussi choisir le mode de calcul rapide pour le calcul d'interférences et de collisions. Une fois le positionnement est effectué, il est possible de visualiser :

- Paramètres d'outil.
- Enveloppes d'outil.
- Axe de l'outil.
- Plan en deux modes (rendu et filaire).
- Axes T1 et T2.
- Point de contact.
- Normale au point de contact.
- Contour.
- Sommets des contours.
- Cellules chevauchantes avec les enveloppes d'outil.
- Points en collisions et en interférences avec l'outil.

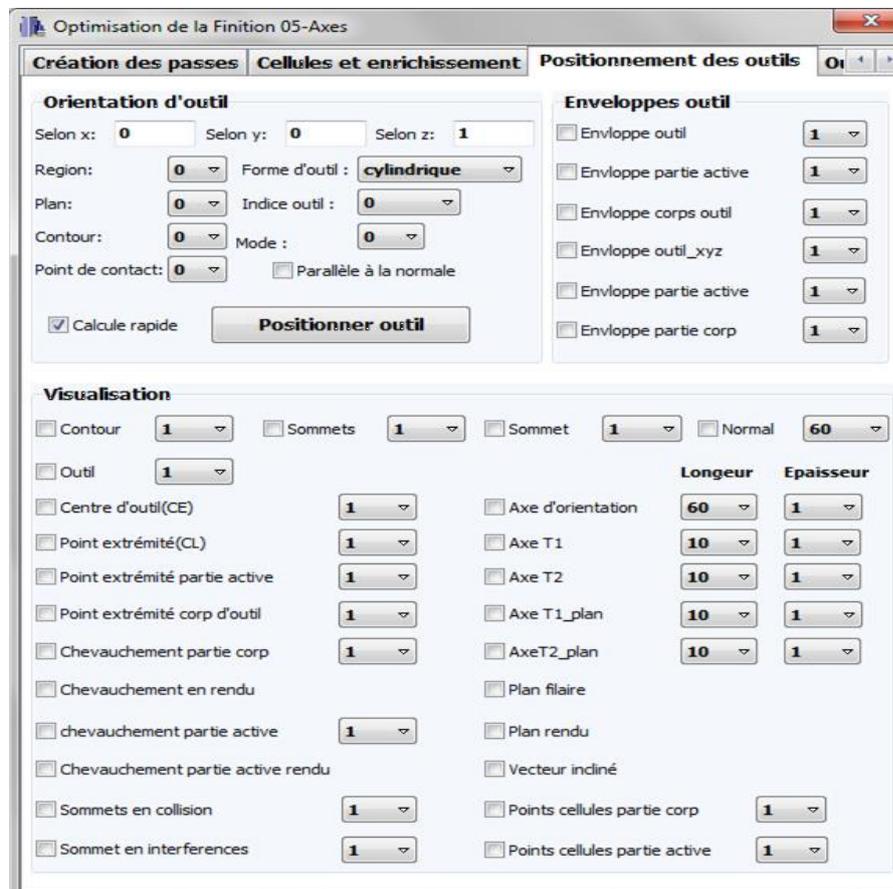


Figure 8. Onglet « Positionnement des outils ».

3.6. Outil optimum:

Cet onglet sert à déterminer l’outil optimum pour un point de contact, contour et les outils optimum pour usiner toute la surface (Figure 9).

Afin de déterminer l’outil optimum pour un point de contact, il suffit que l’utilisateur choisit une région, plan, contour, point de contact et choisit le mode de correction et saisit l’intervalle des angles de rotation, l’incrément et la constante « **cst** » et par clic sur le bouton « **Outil optimum sommet** » l’utilisateur peut visualiser :

- Outil optimum.
- Différents changements de l’outil (rotations et inclinaisons).
- Angles de rotation et d’inclinaison.

Pour identifier l’outil optimum d’un contour il suffit que l’utilisateur choisit une région, plan, contour, ainsi le mode de correction et saisit l’intervalle des angles de rotation, l’incrément et la constante « **cst** » et par un clic sur le bouton « **Outil optimum contour** ». A la fin, la forme et l’indice de l’outil optimum de contour choisi sont affichés.

Le dernier bouton « **Outil optimum contours** » sert à calculer les outils optimaux de toute la surface. Un simple clic sur ce bouton, l’utilisateur peut voir le pourcentage d’avancement de calcul.

The screenshot shows the 'Optimisation de la Finition 05-Axes' software interface. The 'Outil optimum' tab is active. The interface includes the following sections:

- Indices:** Region: 0, Plan: 0, Contour: 0, Point de contact: 0.
- Rotations:** Autour de la normale, Vers le segment, Autour du segment.
- Mode de correction:**
 - α min: 0, α max: 180, Increment: 1
 - β min: 0, β max: 15, Increment: 1
 - γ min: 0, γ max: 90, Increment: 1
 - Alpha: 0.000, Beta: 0.000, Gamma: 90.000
 - cst: 100
 - Autour de la normale, Vers le segment, Autour du segment
- Outil optimum sommet:** Outil optimum sommet button, Outil optimum sommet checkbox.
- Outil optimum contour:** Outil optimum contour button, Forme d'outil, Indice outil, Calcul sans affichage checkbox.
- Outils optimum contours:** Outils optimum contours button, Indice sommet: 11 / 11, Indice outil: 11 / 16, Forme: cylindrique.
- Progression:** Pourcentage d'avancement: 100%

Figure 9. Onglet « Outil optimum ».

Une fois le calcul des outils optimum est terminé, l'utilisateur passe à l'onglet suivant « **Nombre d'outils** »

3.7. Nombre d'outils :

Le but de cet onglet est de déterminer le nombre d'outils optimum à utiliser parmi les outils optimums déjà calculés dans l'onglet précédent et en intégrant aussi la contrainte du nombre maximum d'outils à utiliser (capacité du magasin d'outil de la machine) (Figure 10).

- Bouton « **Déterminer les outils utilisés** » sert à déterminer le nombre d'outils utilisés dans l'onglet précédent.
- Bouton « **Calculer nombre d'outil** » sert à déterminer le nombre d'outils à utiliser selon les outils optimum déjà calculés dans l'onglet précédent et en intégrant aussi la contrainte du nombre maximum d'outils à utiliser (capacité du magasin) saisie par l'utilisateur.
- Bouton « **Déterminer le nombre d'utilisation** » sert à déterminer le nombre d'utilisation de chaque outil.
- Bouton « **Déterminer l'optimisation** » sert à appliquer la stratégie visant à utiliser un nombre prédéfini d'outils afin d'optimiser le temps d'usinage.
- Bouton « **Repositionner les outils optim** » sert à trouver la position qui ne crée ni interférence ni collision pour les nouveaux outils sélectionnés.
- Bouton « **Insérer les point E/S** » sert à ajouter à chaque contour usinable quatre (04) points, deux points au début appelés points d'engagement et deux points à la fin appelés points de dégagement. L'utilisateur saisit la distance d'engagement et la distance de sécurité puis clique sur ce bouton.
- Bouton « **Générer les points de passage d'outils sans optimisation** » sert à définir le trajet d'outils.
- Bouton « **Générer les points de passage d'outils avec optimisation** » sert à définir le trajet d'outils selon la forme et la dimension d'outil.

L'utilisateur peut visualiser les points CL.

Optimisation de la Finition 05-Axes

Positionnement des outils | **Outil optimum** | Nombre d'outils | Simulation du trajet

Nombre d'outils utilisés

Nombre total : 5

Cylindrique : 5 Torique : 0 Hémisphérique : 0

Déterminer les outils utilisés

Outils à utiliser

Nombre d'outils à utiliser : 4

Calculer nombre d'outil

Cylindrique : 4 Torique : 0 Hémisphérique : 0

Déterminer le nombre d'utilisation

Déterminer l'optimisation

Repositionner les outils optim

Trajet d'outil

Distance d'engagement : 100 Distance de sécurité : 10

Insérer les points E/S

Générer les points de passage d'outil sans optimisation

Générer les points de passage d'outil avec optimisation

Visualisation

Points Cl

Figure 10. Onglet « Nombre d'outils ».

3.8. Simulation du trajet :

Cet onglet sert à simuler le trajet d'outils. Un simple clic sur le bouton « **Lancer** » les paramètres d'outils, point CL et l'axe d'outil sont affichés (Figure 11).

- Bouton «**Début** » permet d'afficher l'état initial de l'outil.
- Bouton «**Fin**» permet d'afficher l'état final de l'outil.

L'utilisateur peut :

- Manipuler la vitesse d'affichage.
- Visualiser les points CL.
- Visualiser l'outil.
- Visualiser le trajet d'outil.
- Visualiser le pourcentage d'avancement de simulation.

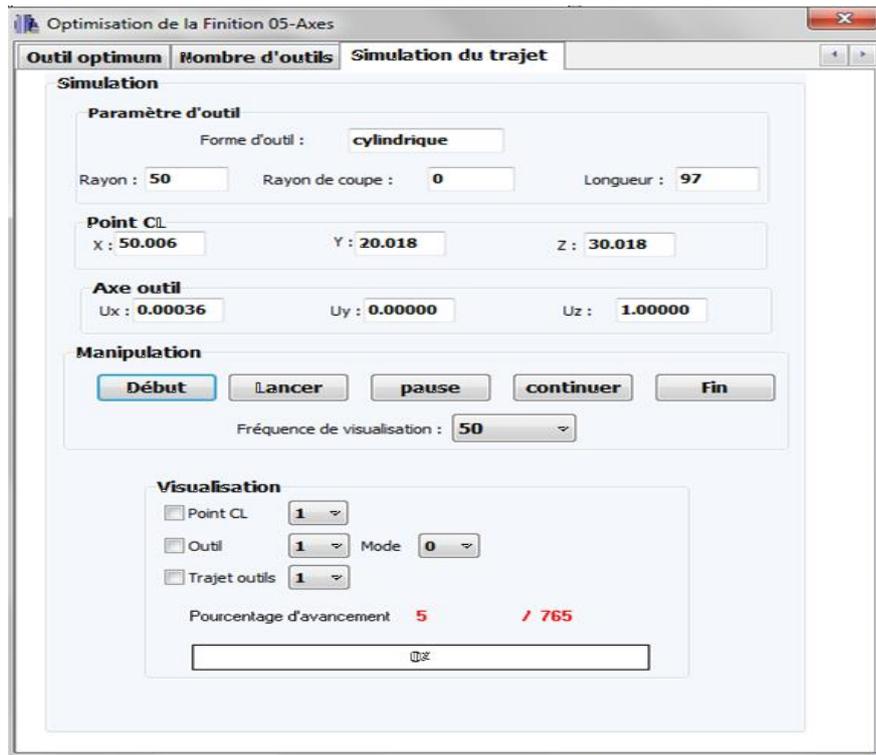


Figure 11. Onglet « Simulation du trajet ».

4. Test et validation :

Arrivé au terme de la présentation de l'application, la phase de validation du travail et les différents résultats auxquels nous avons abouti sont présentés dans la suite. Le test de validation consiste à utiliser un modèle STL d'une forme gauche issue d'une conception générée dans un logiciel de CAO spécialisé dans la conception des surfaces gauches. Tout le processus d'optimisation de la finition 05-Axes est appliqué sur l'exemple représenté par la Figure 12 pour montrer la validation de l'application.



Figure 12. Exemple de test.

- **Etape 1** : lecture du fichier texte contenant les paramètres STL de la pièce à usiner (Figure 13).

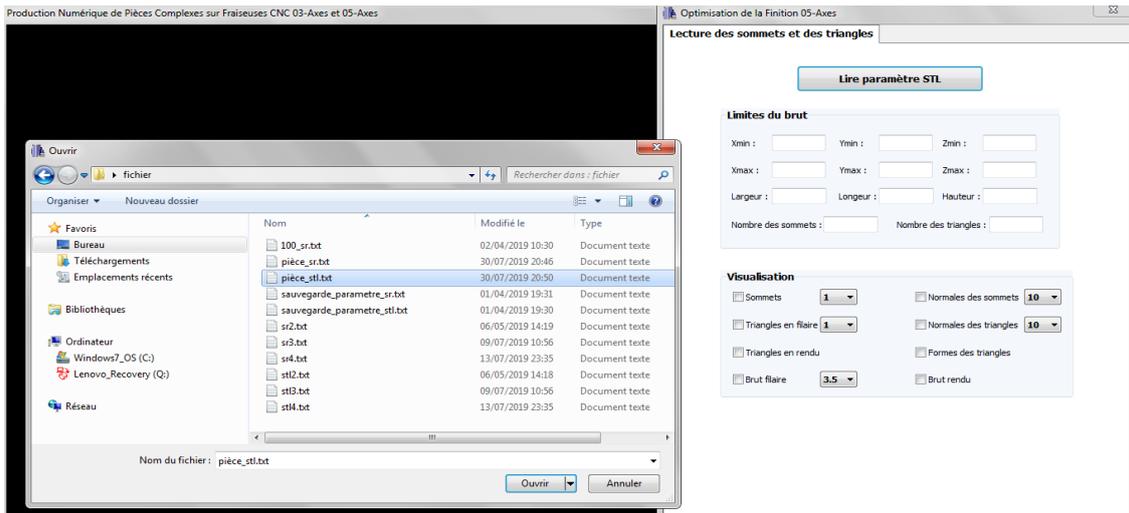


Figure 13. Lecture du fichier de la pièce.

Les résultats obtenus lors la lecture sont :

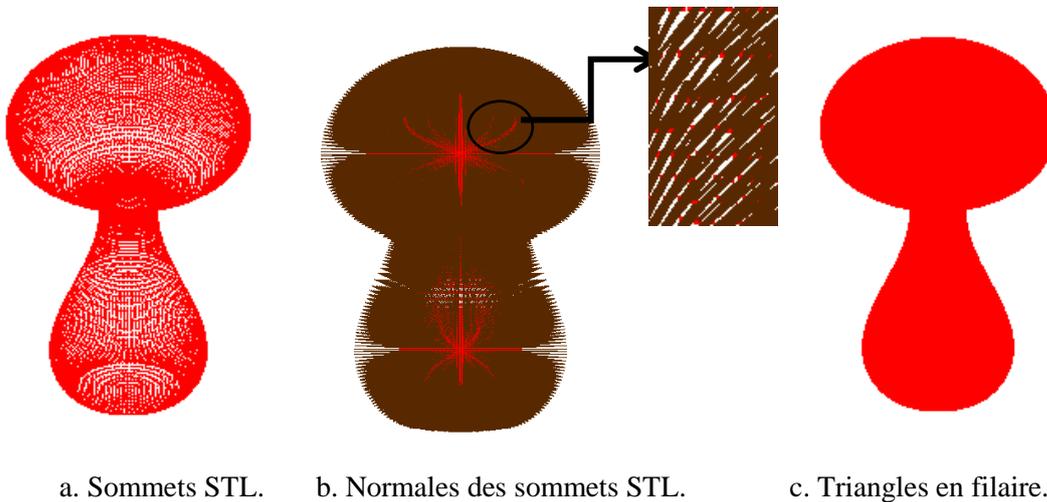
$X_{min} = -19.474 \text{ mm}$ $Y_{min} = 20.627 \text{ mm}$ $Z_{min} = -33.725 \text{ mm}$

$X_{max} = 20.398 \text{ mm}$ $Y_{max} = 60.499 \text{ mm}$ $Z_{max} = 29.390 \text{ mm}$

Longueur = 39.872mm Largeur = 39.872mm Hauteur = 63.115mm

Nombre total des sommets = 35224 Nombre total des triangles = 70444

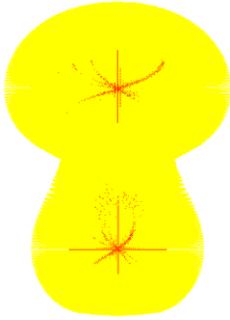
Les autres paramètres qui sont visualisés sont donnés par la Figure 14.



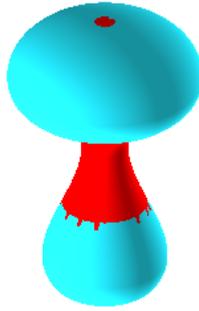
a. Sommet STL.

b. Normales des sommets STL.

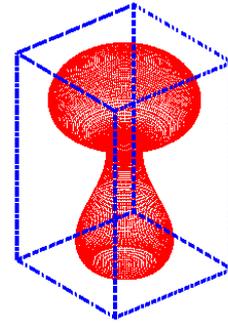
c. Triangles en filaire.



d. Normales des triangles STL.



e. Formes des triangles.



f. Brut en filaire.

Figure 14. Paramètres visibles après la lecture du fichier.

➤ **Etape 2 :** ouverture du fichier contenant les paramètres des sous régions.

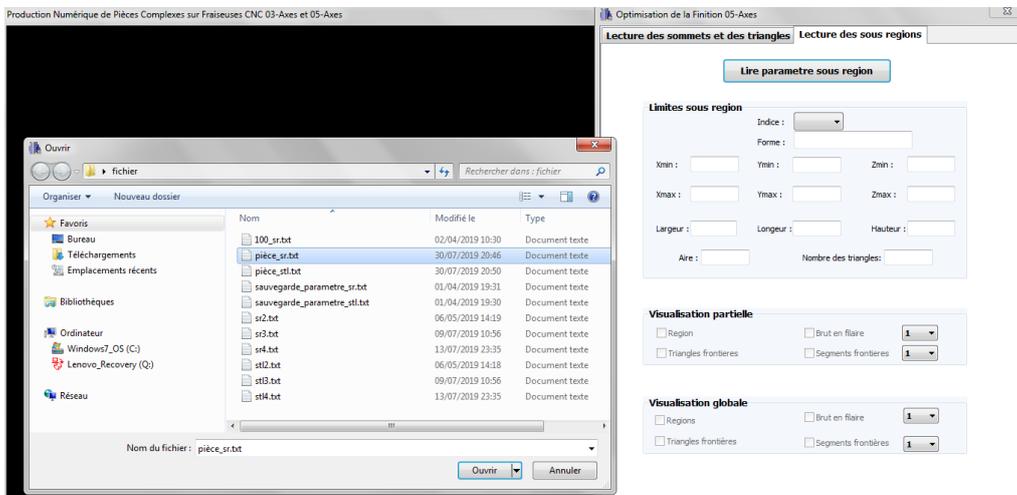
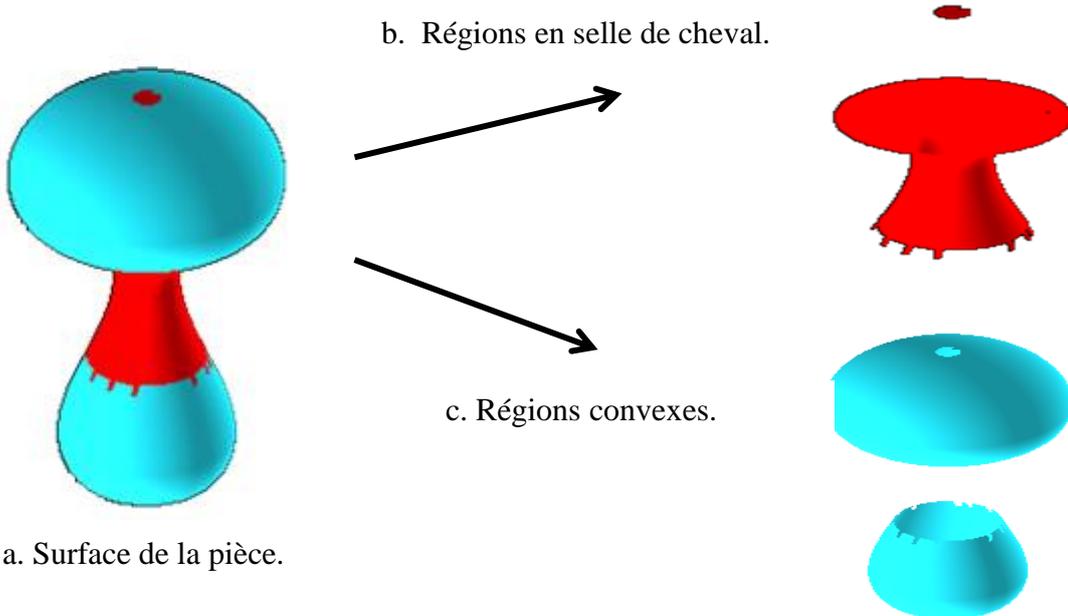


Figure 15. Lecture du fichier pièce.

La surface de la pièce est composée de 05 régions : 03 convexe et 02 en selle de cheval.



a. Surface de la pièce.

b. Régions en selle de cheval.

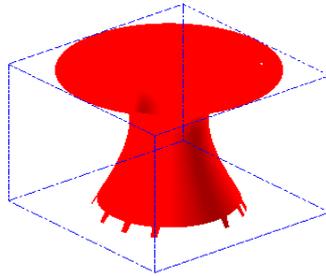
c. Régions convexes.

Figure 16. Régions de la surface.

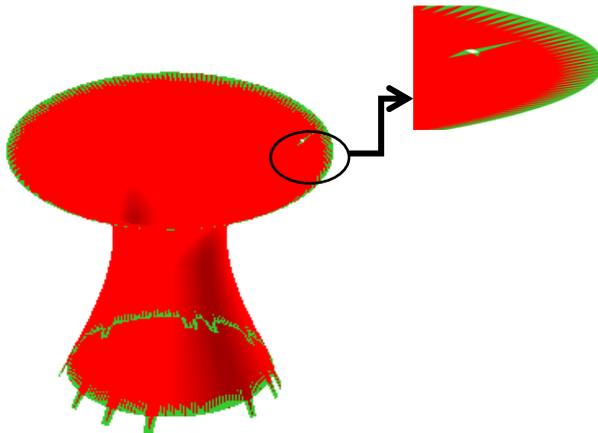
Une fois l'utilisateur sélectionne l'indice de la région, les différents paramètres seront affichés. A titre d'exemple, la région « selle de cheval » est considérée (Figure 17).

Les résultats obtenus lors de la lecture sont :

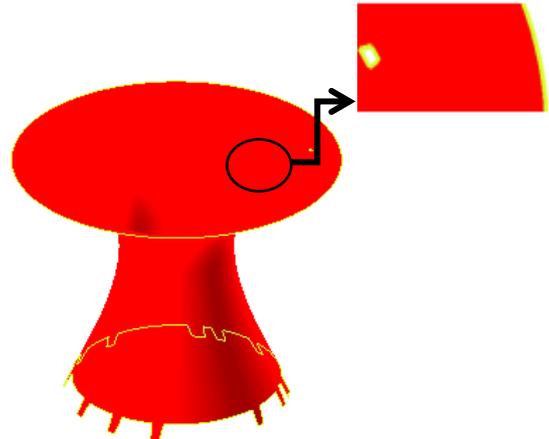
Xmin = -13.244 mm	Ymin = 26.858mm	Zmin = -14.928mm
Xmax = 14.167mm	Ymax = 54.268mm	Zmax = 8.205mm
Longueur = 27.411mm	Largeur = 27.411mm	Hauteur = 23.133mm
Aire = 1252.300	Nombre total des triangles = 18235	



a. Brut de la région.



b. Triangles frontières.



c. Segments frontières.

Figure 17. Visualisation des paramètres de la région « selle de cheval ».

- **Etape 3 :** création des passes de la pièce en spécifiant la profondeur de passe. Pour l'exemple de validation, le nombre de passes total est égal à 25 pour une profondeur de 4mm. Afin de visualiser les plans d'usinage, nous avons choisi la region « Selle De Cheval » comme exemple.

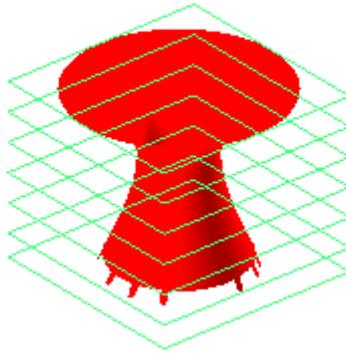
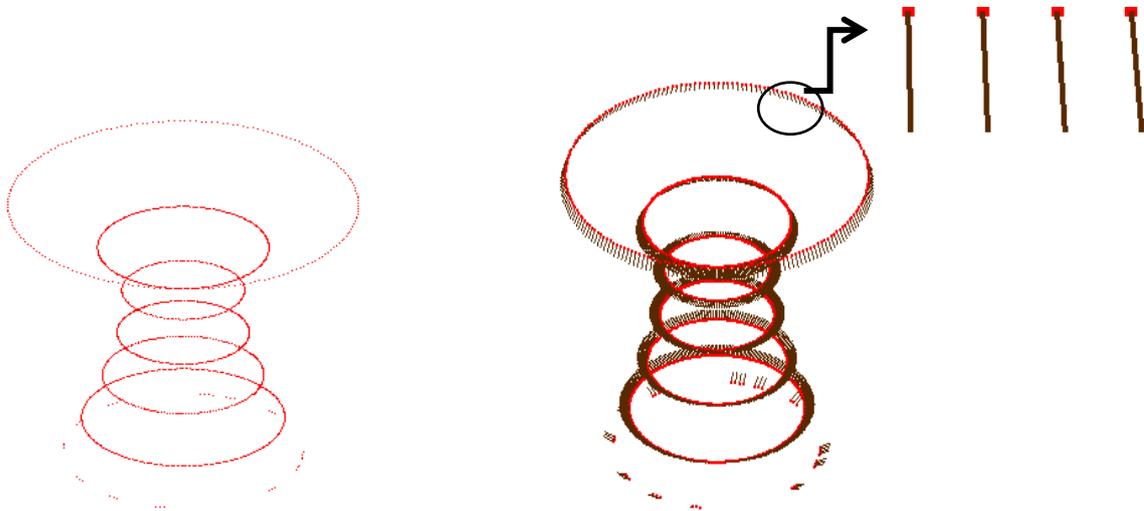


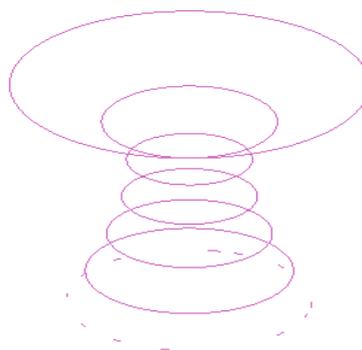
Figure 18. Passes d'une région « Selle De Cheval ».

➤ **Etape 4 :** calcul des sommets d'intersection. Les paramètres de visualisation pour la region « Selle De Cheval » sont montrés dans la Figure 20.



a. Sommets d'intersection.

b. Normales des sommets d'intersection.



c. Segments d'intersection.

Figure 19. Visualisation des paramètres d'une région.

- **Etape 5 :** création des contours. Afin de visualiser les contours résultants, une region « Selle De Cheval » est considérée.

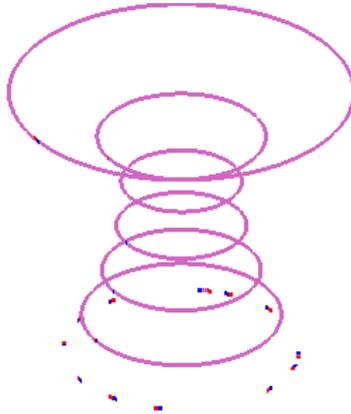
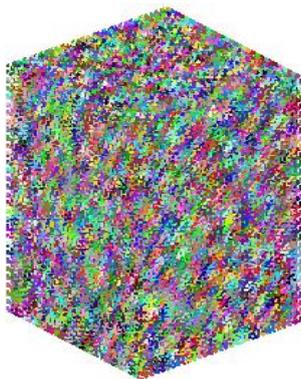
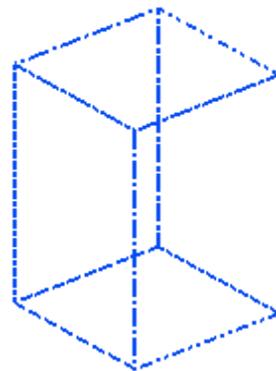


Figure 20. Contours d'une région.

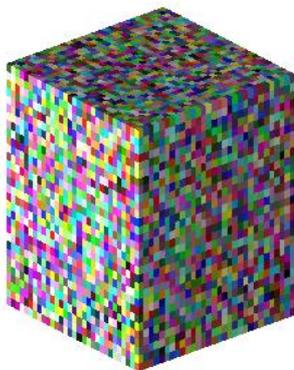
- **Etape 6 :** division du brut en cellules avec un nombre de 30 cellules suivant les trois axes X, Y et Z (Figure 21). Après l'affectation des sommets aux cellules et aux triangles, les résultats obtenus sont les suivants :



a. Cellules en filaire.



b. Une cellule en filaire.



c. Cellules en rendu.



d. Une cellules en rendu.

Figure 21. Création des cellules de sommets.

- **Etape 7** : sélection des triangles à enrichir en précisant un seuil qui varie entre l'aire minimum et l'aire maximum des triangles du modèle STL. Dans l'exemple de validation, le seuil est pris égal à 0.1mm^2 . La Figure 12 montre deux zones. La zone verte représente les triangles à enrichir et la zone rouge les triangles à ne pas enrichir.

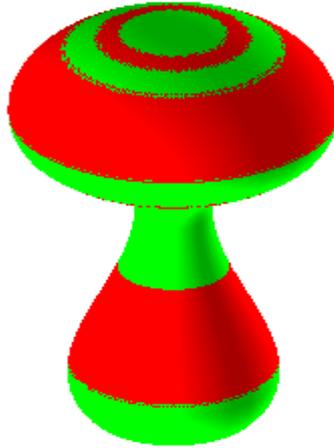


Figure 22. Triangles enrichis.

- **Etape 8** : enrichissement des triangles obtenus dans l'étape 7 en spécifiant la densité des points sur 1mm^2 . Pour l'exemple de validation le nombre de point/ mm^2 est fixé égal à 10. Le résultat obtenu est présenté dans la Figure 23.

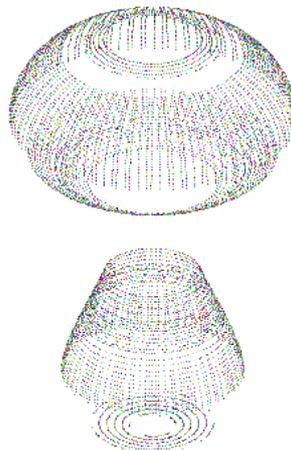


Figure 23. Enrichissement du modèle STL.

- **Etape 9** : affectation des points générés aux cellules (Figure 24).



Figure 24. Sommets des cellules.

- **Etape 10 :** en chaque point de contact d'un contour qui appartient à une passe d'une région sélectionnée, les paramètres d'outil sont calculés et les enveloppes sont déterminées. Par la suite les cellules qui se chevauchent avec les enveloppes d'outil sont identifiées ainsi que la détection des problèmes d'interférences et de collisions en fonction de la forme d'outil et des paramètres introduits (Figure 25 à Figure 28).

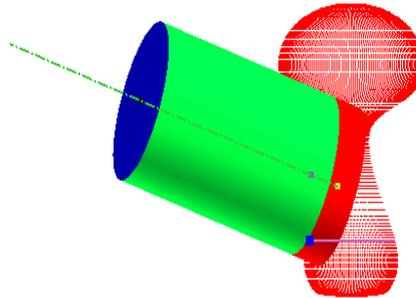


Figure 25. Visualisation du point de contact et des paramètres d'outil.

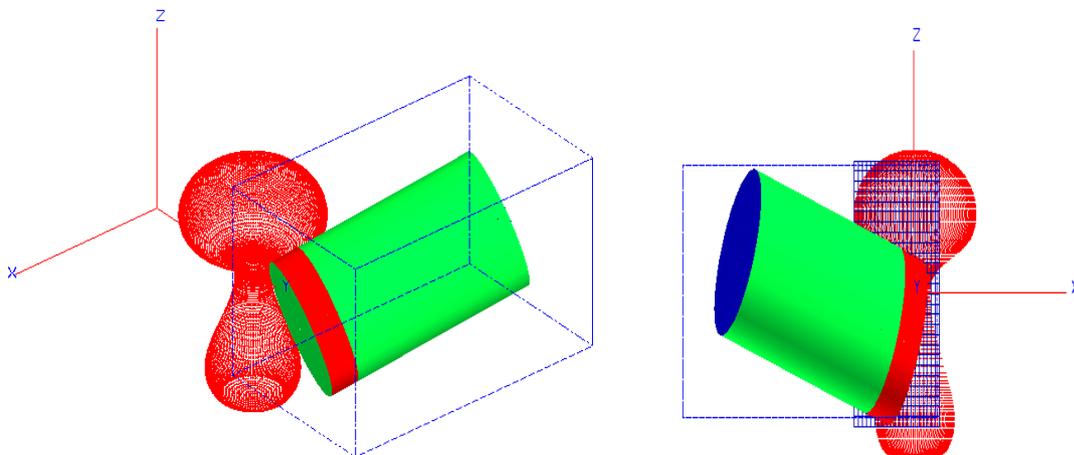


Figure 26. Enveloppe de la partie corps d'outil et cellules de chevauchement.

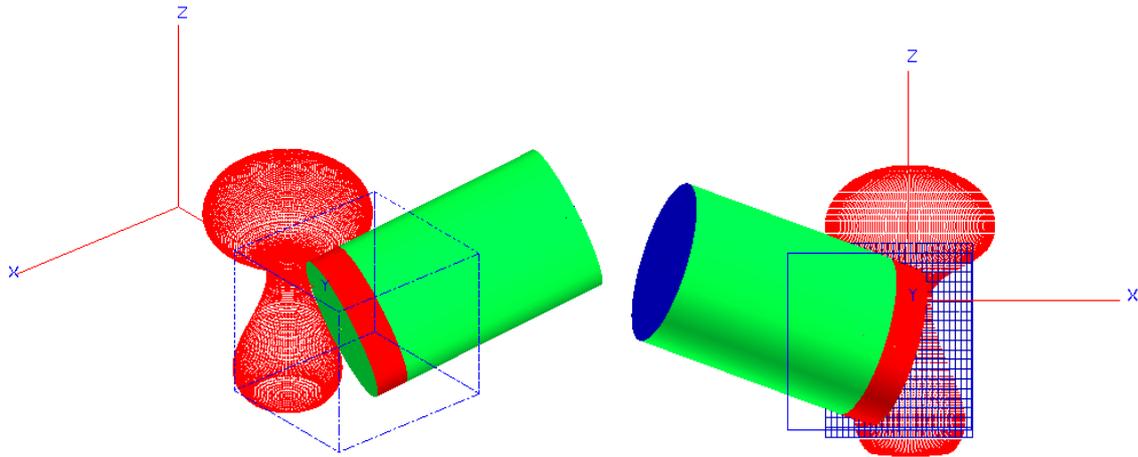
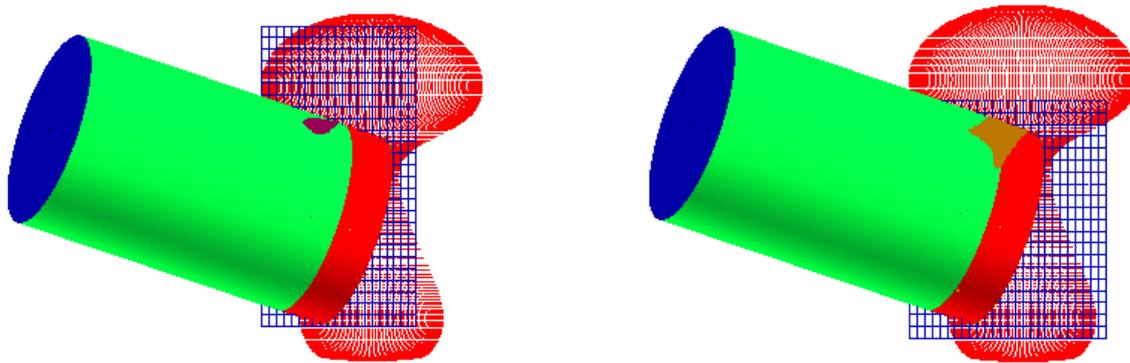


Figure 27. Enveloppe de la partie active d'outil et cellules de chevauchement.



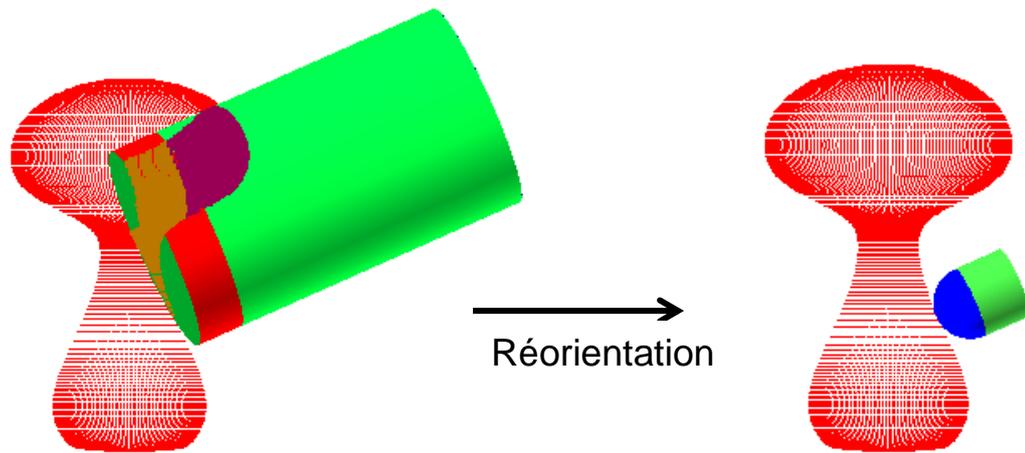
a. Sommets provoquant collisions.

b. Sommets provoquant interférences.

Figure 28. Point de contact avec interférences et collisions.

➤ **Etape 11 :**

- ✓ Affectation au point de contact sélectionné un outil optimum. L'affectation est réalisée en parcourant la base de données d'outils du plus grand au plus petit jusqu'à trouver la position de l'outil hors collisions et hors interférences. En cas de collisions ou d'interférences, une réorientation d'outil est appliquée pour éviter ces problèmes. La Figure 29 montre un exemple d'affectation telle qu'une réorientation d'outil est effectuée en appliquant une rotation autour du segment.



a. Problème d'interférence et de collision.

b. Correction du problème.

Figure 29. Affectation de l'outil optimum au point de contact.

- ✓ Affectation au contour sélectionné un outil optimum. L'affectation est réalisée en parcourant tous les sommets du contour et pour chaque sommet l'outil optimum sera déterminé et affecté au prochain sommet. L'outil optimum du dernier sommet du contour sera considéré comme outil optimum du contour et sera repositionné sur tous les sommets du contour. La Figure 30 montre un exemple d'affectation sachant qu'une réorientation d'outil est effectuée en appliquant une rotation autour du segment. A la fin du calcul, la forme et indice d'outil affecté seront affichés.

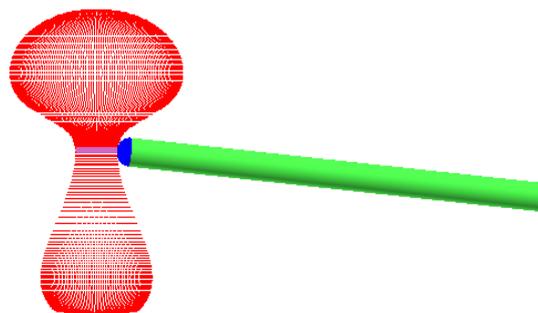


Figure 30. Affectation de l'outil optimum à un contour.

- ✓ Affectation à tous les contours de la pièce des outils optimums. L'affectation est réalisée en parcourant tous les contours et pour chaque contour un outil optimum sera déterminé. Lors du calcul, les paramètres qui sont visibles sont :

Indice sommet : 327/327

Indice outil : 17/18

Forme d'outil : Hémisphérique

➤ **Etape 12 :**

- ✓ Détermination des outils utilisés dans étape 11.3. Les résultats obtenus sont :

Nombre total des outils utilisés = 8

Tel que : Cylindrique = 1

Torique = 2

Hémisphérique = 5

- ✓ Détermination des outils à utiliser en fixant le nombre d'outil total qui sera placé dans le magasin d'outils. Pour l'exemple de validation, le nombre d'outils total optimisé est égal à 05. Donc, la partition sera :

Cylindrique = 1

Torique = 1

Hémisphérique = 3

➤ **Etape 13 :** détermination du nombre d'utilisation pour chaque outil du contour.

➤ **Etape 14 :** détermination des outils optimums. Dans notre exemple de validation, 05 outils Hémisphériques sont obtenus dans l'étape 11. Après optimisation, 02 outils seront ignorés et 03 seront gardés.

➤ **Etape 15 :** repositionnement des outils optimums.

➤ **Etape 16 :** insertion des points d'engagement et dégagement pour chaque contour usinable en introduisant la distance d'engagement et la distance sécurité. Les distances choisies dans le test de validation sont :

Distance d'engagement = 100mm.

Distance sécurité = 10mm.

➤ **Etape 17 :** génération des points de passage d'outils. Pour cela, deux modes sont possibles :

1. ***Sans optimisation*** : le séquençement d'outils se fait par rapport aux outils affectés aux contours.
2. ***Avec optimisation*** : le séquençement d'outils se fait par ordre de forme commençant par la forme cylindrique, torique puis hémisphérique et par dimensions du plus grand au plus petit outil.

➤ **Etape 18 :** simulation des mouvements de l'outil. Lors de la simulation, en chaque position de passage de l'outil, les paramètres de l'outil, les coordonnées du point

extrémité (CL), l'axe d'orientation de l'outil sont affichés. La Figure 35 montre les paramètres lors la simulation :

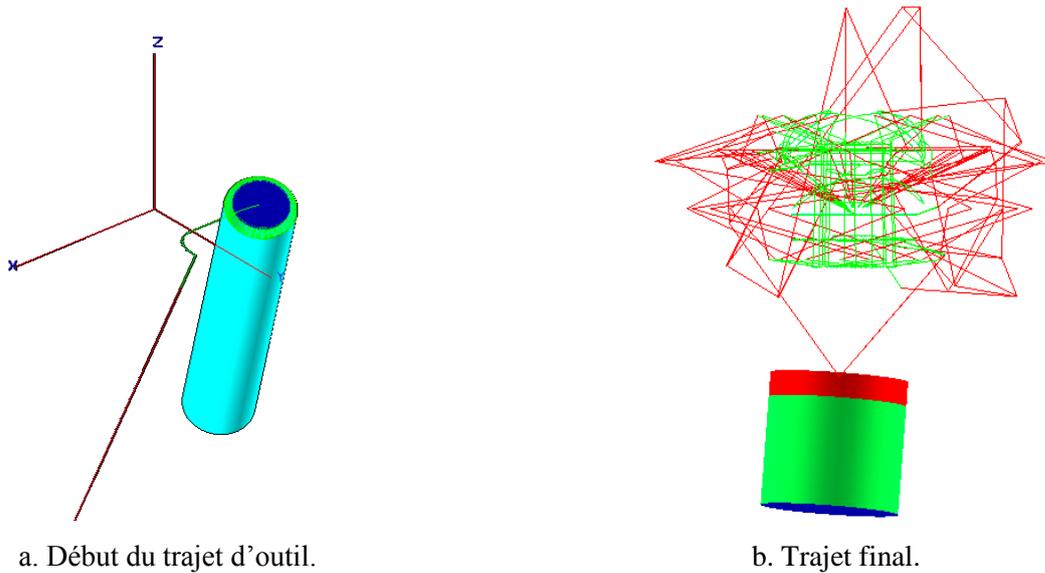


Figure 31. Simulation du trajet d'outil.

Conclusion:

Au cours de ce chapitre, nous avons présenté notre application afin de valider les résultats obtenus. Dans un premier temps, nous avons présenté l'environnement de développement de notre application ainsi que les interfaces pour décrire les différentes fonctionnalités. Dans un second temps, nous avons testé les différentes fonctionnalités de notre application à travers un exemple depuis la lecture des paramètres STL jusqu'à la simulation du trajet d'outil.

CONCLUSION GENERALE

Le travail réalisé au sein de l'équipe « Conception et Fabrication Assistées par Ordinateur » «CFAO » de la Division Productique et Robotique du « CDTA » s'articule autour du développement d'une application logicielle graphique et interactive sous Windows, ayant pour but la combinaison des trois formes d'outils « Cylindriques, Hémisphériques et Toriques » lors de la finition des surfaces de formes complexes par la stratégie « Z-Constant », définies par leurs modèles « STL », sur des fraiseuses numériques à 05-axes, en considérant les formes locales des surfaces et les temps d'usinage.

Lors de la réalisation de ce projet, une étude bibliographique relative aux méthodes de conception et d'usinage des surfaces gauches en 05-axes a été menée. Par la suite nous avons proposé une approche pour déterminer la combinaison optimale des trois formes d'outils permettant l'évitement des interférences et des collisions pour optimiser le nombre d'outils à utiliser en intégrant la contrainte du nombre maximum d'outils à utiliser et en proposant la combinaison de trois modes d'orientation de l'outil pour l'évitement des interférences et des collisions, ainsi que les algorithmes développés ont été détaillés. L'achèvement du travail réalisé est l'implémentation informatique de l'approche proposée sous Windows en utilisant le langage C++, l'environnement de développement Embarcadero et la bibliothèque graphique OpenGL.

Le résultat de l'application développée est l'intégration à l'application logicielle graphique de l'équipe « CFAO » des fonctions qui permettent de :

- ✓ Récupérer les paramètres géométriques des surfaces gauches.
- ✓ Créer les passes d'usinage.
- ✓ Générer les contours de contact outil/surface.
- ✓ Créer les cellules et enrichir le modèle STL.
- ✓ Positionner les différentes formes d'outils.
- ✓ Déterminer l'outil optimum pour chaque point de contact.
- ✓ Associer l'outil optimum à chaque contour.

- ✓ Déterminer les outils optimums théoriques pour toute la surface.
- ✓ Optimiser le nombre d'outils utilisé.
- ✓ Optimiser le trajet d'outil en minimisant le nombre de changement d'outils.
- ✓ Simuler virtuellement le trajet d'outils.

En perspective à notre travail, nous recommandons de traiter les points suivants :

- Proposition d'une nouvelle approche pour l'orientation de l'outil pour éviter les interférences et les collisions.
- Intégration de la notion du parallélisme dans le calcul des outils optimum.
- Génération du programme d'usinage « G-Code » pour n'importe quelle cinématique des fraiseuses numériques 05-axes.
- Génération du trajet d'outil en utilisant d'autres stratégies d'usinage (Plans Parallèles, Iso paramétrique, Iso-Crête, ...etc.).
- Considération de l'environnement de l'usinage (porte outil, montage d'usinage, ...etc.) lors du test des collisions.
- Simulation virtuelle de l'usinage pour vérifier la qualité de la pièce.
- Calcul des cônes de visibilité et d'accessibilité pour accélérer la correction des interférences et des collisions.
- Finition des surfaces gauches avec le profil de la fraise (fraisage en roulant).
- Programmation en utilisant des cartes GPU.
- Génération du trajet d'outil en utilisant d'autres modes d'interpolation (polynomiale, B-Spline et NURBS).
- Simulation virtuelle de la cinématique des fraiseuses numériques 05-axes.

REFERENCES BIBLIOGRAPHIQUES

[1] : <http://www.daglo.fr/techniques-usinage/>.

[2] : <http://www.usimm.ca/quelles-sont-les-etapes-de-lusinage-de-piece>.

[3] : Ghemouri L. Rezkellah S. « Optimisation de la fabrication des surfaces complexes sur des fraiseuses 05-axes par combinaison des outils hémisphériques, cylindriques et toriques », 2018, mémoire de fin d'étude, Université Saad Dahlab, Blida.

[4] : M. Bey, « Modélisation des courbes et des surfaces », Avril 2000.

[5] : En ligne : <https://wims.math.cnrs.fr/wims/wims.cgi?session=HY37A88FAA.3&+lang=Fr&+module=U2%2Fanalysis%2Fdocstokes.fr&+cmd=reply&+job=read&+doc=1&+block=tangent>

[6] : En ligne : <http://www.jaicompris.com/lycee/math/espace/orthogonalite-espace.php>

[7] : En ligne : <https://images.math.cnrs.fr/Autour-des-surfaces-de-Willmore.html?lang=fr>

[8] : M. Bey, H. Bendifallah, S. Kader et K. Boukhalfa « Optimisation de l'usinage des surfaces gauches par l'intégration des formes locales des surfaces et des stratégies d'usinage ». Actes des 6^{ème} Journées de Mécanique de l'Ecole Militaire Polytechnique 15-16 Avril 2008, Bordj El-Bahri, Alger.

[9] : K. Nguyen Tan « Surfaces polyédriques et surfaces paramétriques : une reconstruction par approximation via les surfaces de subdivision ». 2010. Thèse de Doctorat. Aix-Marseille 2, France.

[10] : H. Bendifallah, M. Bey, W. Ould Amar et N. Mahroug, N. « Optimisation de l'opération de finition des surfaces gauches par la combinaison des formes et des dimensions d'outils à partir des modèles STL ». Congrès Algérien de Mécanique, CAM2011, 14-17 novembre 2011, Guelma, Algérie.

[11] : N. Belkacemi, « Développement d'une application pour la finition des pièces des formes gauches sur des fraiseuses à 5 axes par combinaison d'outils hémisphériques », 2016, mémoire de fin d'étude, Université Saad Dahlab, Blida.

[12] : En ligne : <http://www.usimm.ca/quelles-sont-les-differences-entre-lusinage-3-axes-et-5-axes/>

[13] : L. Kheris et W. Hedroug, « Conception et développement d'une application de simulation d'enlèvement de matière lors de la finition des surfaces complexes sur fraiseuses 05-axes », 2016, mémoire de fin d'études, Université Saad Dahlab, Blida.

[14] : S. Djebali, « Optimisation globale du processus d'usinage des surfaces gauches », 2014. Thèse de Doctorat, Université de Toulouse, Toulouse III-Paul Sabatier, France.

[15] : C. Danjou, C. « Ingénierie de la chaîne numérique d'industrialisation : proposition d'un modèle d'interopérabilité pour la conception-fabrication intégrées », 2015, Thèse de Doctorat, Université de Compiègne, France.

[16] : F. Bettine, « Etude du processus d'usinage des pièces mécaniques sur des machines CNC 5 axes », 2012, Thèse de Doctorat, Université Mohamed Khider, Biskra.

[17] : S. Bensaada, « Coupe des métaux ». Cours, Université Mohamed Khider, Biskra.

[18] : P. Gilles, « Positionnement d'outil torique pour l'usinage de surfaces gauches en fraisage 5-axes avec équilibrage de l'effort de coupe transversale », 2008, Thèse de Doctorat, Université de Toulouse, Université Toulouse III-Paul Sabatier, France.