

N° Ordre...../FT/UMBB/2024



République Algérienne Démocratique et populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
UNIVERSITE M'HAMED BOUGARA BOUMERDES



Faculté de technologie
Département Génie Mécanique

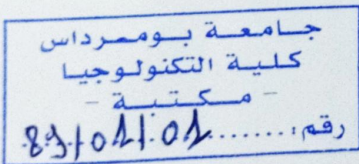
Polycopie de cours

Titre :

Conception et Fabrication Assistées par Ordinateur (CFAO)

Réalise par :

Dr. HOUARI Amin



Année université 2023-2024

Préface

Les techniques de la conception et de la fabrication assistées par ordinateur sont devenues primordiales dans le domaine de fabrication mécanique. Le perfectionnement des étudiants dans ce domaine devient une nécessité. Ce cours est imprégné d'une démarche pédagogique et pratique, il propose une méthodologie claire et simple pour la maîtrise de la CFAO.

Ce cours est destiné aux étudiants de Master II, Mécatronique, leur permettant une prise en main et une première utilisation du logiciel de CFAO. A la fin du semestre, l'étudiant devra acquérir les compétences suivantes :

- Modélisation des pièces de formes complexes (moules, matrices, ...).
- Simulation du processus d'usinage.
- Interprétation et vérification du programme d'usinage généré automatiquement.

Sommaire

Chapitre I. Généralités sur la CFAO	01
Chapitre II. Modélisation des courbes	06
Chapitre III. Modélisation des surfaces	24
Chapitre IV. Modélisation des solides	33
Chapitre V. Les machines-outils à commande numérique	41
Chapitre VI. Programmation de MOCN	52
Références Bibliographique	67

Introduction

Pour aider les métiers de la conception et de la fabrication de produits manufacturiers, de nombreux logiciels de Conception Assistée par Ordinateur (CAO) et de Fabrication Assistée par Ordinateur (FAO) ont été développés et commercialisés au cours de ces cinquante dernières années. Le premier logiciel de dessin est apparu en 1963 : le DAO (Dessin Assisté par Ordinateur). En 1970, les logiciels de conception ont été développés pour une modélisation 3D (CAO) mettant en œuvre deux différentes techniques : B-Rep (Boundary Representation) et CSG (Constructive Solid Geometry). Plus tard, la modélisation s'est appuyée sur les NURBS (Non Uniform Rational B-Spline) ce qui a permis d'harmoniser la représentation de toutes les courbes et les surfaces utilisées en CAO. Le premier logiciel de FAO commercialisé a été développé par Patrick Hanratty en 1957.

Les logiciels de la CAO, et de la FAO sont énormément évolués grâce à l'augmentation de la puissance de calcul, de mémorisation et de visualisation des ordinateurs. Cependant les systèmes de CAO et de FAO ont longtemps évolué en parallèle, posant ainsi des problèmes pour passer de l'un à l'autre.

La conception et de la fabrication assistées par ordinateur (CFAO) sont souvent présentées conjointement. En effet, ces deux champs d'application informatique en génie mécanique ont plusieurs points en commun. Chacune de ces applications porte sur les mêmes pièces mécaniques et utilise une base informatique commune : les ressources graphiques d'édition et de gestion.

En fait, la CAO et la FAO correspondent aux activités de deux processus complémentaires qui permettent de traduire une idée abstraite en un objet réel. Ces deux processus trouvent leurs définitions dans le travail du bureau d'étude et du bureau des méthodes. Partant de la détermination d'un besoin, le bureau d'étude assure la conception assistée par ordinateur de modèle en réalisant le dessin de définition des pièces. A partir de ces derniers, le bureau des méthodes élabore le processus de FAO de ces pièces. Le résultat de la FAO c'est un programme d'usinage des pièces mécaniques. Ce programme d'usinage est transmis vers la machine-outil à commande numérique (MOCN) pour l'exécution.

Ce polycopié de cours comporte six chapitres principaux, où sont exposées les notions de modélisation des courbes, des surfaces, des solides, définition de la machine et du post processeur, Génération du programme de commande numérique.

Chapitre I

Généralités sur la CFAO

Plan de cours

Objectifs généraux

- Connaissances des étudiants dans le domaine de la CFAO
- Modélisation des pièces de formes complexes
- Simulation du processus d'usinage.
- Interprétation et vérification du programme d'usinage généré automatiquement.

Objectifs de chapitre

- Définir les notations sur la CFAO et MOCN
- Connaître les logiciels de CAO/FAO
- Décrire le processus de CFAO

Déroulement

Le chapitre 1 sera abordé durant une séance :

- Généralités sur la CFAO

Pré acquis

- Outils informatique
- Notion du logiciel de conception
- Lecture d'un dessin de définition
- Machines outils conventionnelles

Moyens

- Tableau
- Vidéo projecteur
- Visite des machines CNC

I.1. Introduction

La Conception et la Fabrication Assistées par Ordinateur (**CFAO**) est une technologie qui permet d'utiliser des systèmes informatiques (*matériels et logiciels*) dans la conception et la fabrication mécanique. Cette technologie est en mouvement dans le sens d'une plus grande intégration et interaction de la conception et la fabrication. En fin de compte, cette technologie sera dirigée vers un seul objectif qui est l'usine entièrement automatisée dans l'avenir.

La conception et la fabrication assistées par ordinateur se définit comme l'ensemble des aides informatiques **au service technique**. De l'établissement d'un cahier de charges relatif à un nouveau produit jusqu'à la génération des documents et des fichiers nécessaires à la fabrication. Les logiciels de conception et de fabrication assistée par ordinateur (**CFAO**) permettent de concevoir et de fabriquer des prototypes, des produits finis et des séries de produits.

I.2. Définitions CAO/FAO/CN

CFAO est la synthèse de la CAO et de la FAO apparue dans les années 1970 avec l'introduction des machines-outils à commande numérique.

I.2.1 Conception assistée par ordinateur (CAO) :

C'est une opération qui permet de dessiner et de concevoir des pièces mécaniques à l'aide de l'ordinateur. La CAO regroupe les logiciels et les techniques de modélisation géométrique qui permettent de concevoir, de tester virtuellement et de réaliser des produits manufacturés. Parmi les logiciels CAO on mentionne : **SolidWorks**.

I.2.2 Fabrication assistée par ordinateur (FAO) :

C'est une opération qui permet de convertir un fichier **CAO** vers un programme **ISO** (ou blocs **ISO**) pour le transmettre à une **machine-outil à commande numérique**, qui fabriquera la pièce qui a été conçue en CAO. Les logiciels de FAO (comme par exemple **CAMWORKS**) définissent les trajectoires d'outils à partir d'informations géométriques, d'informations techniques, de consigne d'utilisateur et/ou informations provenant de logiciels spécifiques. Elles permettent de représenter les caractéristiques des formes, des contours ou des entités à usiner, la géométrie du brut, la géométrie d'outils.

I.2.3 Programmes de commande numérique (CN) :

La programmation de commande numérique (**CN**) permet de définir des séquences d'instructions permettant de piloter des **machines-outils à commande numérique MOCN**

qui fonctionnent avec un ordinateur connecté à une **machines-outils**. Cette programmation est actuellement fortement automatisée à partir des plans réalisés en **CAO**.

I.3. Processus de Conception et Fabrication Assistées par Ordinateur

La conception de la pièce à fabriquer est faite à l'aide d'un logiciel de Conception assistée par ordinateur (**SolidWorks**) : on nomme le fichier ainsi obtenu "**modélisation 3D**". Cette modélisation en trois dimensions de la pièce à réaliser. La modélisation 3D étant importée sur le logiciel de **FAO** puis relue par celui-ci, il est possible de passer à la programmation des parcours outils, le cœur de l'activité de **FAO**.

Le programmeur crée les parcours en respectant les choix d'outil, les vitesses de coupe et d'avance, et les stratégies d'usinage à mettre en œuvre. Certains outils de **FAO** savent relire directement les fichiers des grands fournisseurs de **CAO**.

Les systèmes **CFAO** utilisent deux applications séparées, une application de CAO et l'autre la FAO, ils nécessitent de sortir d'une application et entrer dans l'autre. Pour générer un programme **CN** d'un modèle géométrique il faut importer le modèle conçu par l'application de CAO sous le format supporté par l'application de **FAO**.

Dans d'autres cas, la CAO et FAO sont complètement intégrés et ne nécessitent pas de transfert. Pour l'opération de fabrication (**FAO**) consiste à piloter par ordinateur les déplacements d'un outil par rapport à cette pièce (ou inversement) installée sur une **machine-outil** à commande numérique (**MOCN**) pour fabriquer une forme.

En général, le logiciel de FAO définit les trajectoires des *outils* choisis sur la modélisation 3D (CAO) et enregistre celles-ci sous forme d'équations. La dernière étape consiste, depuis le programme de **FAO** ainsi élaboré et vérifié, à générer les **blocs ISO** pour la **machine-outil**. Ce programme est appelé un **Post-Processeur**. Le fichier **ISO** obtenu est transmis et exécuté par la **MOCN** (Machine-Outil à Commande Numérique) comme montré sur la figure.I.1.

I.4. Les avantages de CFAO

- Le système CFAO permet d'optimiser la compatibilité entre la CAO et FAO. Via une bonne maîtrise de la conception d'un produit.
- La possibilité de contrôler en amont toutes les étapes de la fabrication. Avec un logiciel de CFAO intégré, les concepteurs réduisent au maximum la marge d'erreur et le gaspillage et limitent les risques de baisse de qualité des produits.

- Autre avantage de cette solution, le gain de temps sur la saisie des données et la réduction des erreurs humaines pendant l'échange des données. De même, une solution CFAO permet la traçabilité des opérations, une analyse pointue des données de gestion et une réduction des dépenses dans l'acquisition de licences logicielles, car la CAO et FAO sont complètement intégrés.

<p>CAO : Fichier Entrer : Définition d'une pièce Utilisation d'un modeler géométrique surfacique et/ou volumique (SolidWorks, Catia...) Fichier Sorte : Modèle surfacique/volumique de la pièce : fichier IGES, STEP, PAR...</p>
<p>FAO : Entre : Modèle CAO de la pièce Génération des trajectoires d'usinage Sorte : Trajectoire voulue de l'outil dans l'espace pièce.</p>
<p>MOCN: Traitement et interprétation des trajectoires par la machine + Interpolations spatiale et temporelle : détermination des commandes de position, vitesse et accélération à piloter + Boucle d'asservissement des axes ⇒ Mouvement de la machine et réalisation de l'usinage.</p>

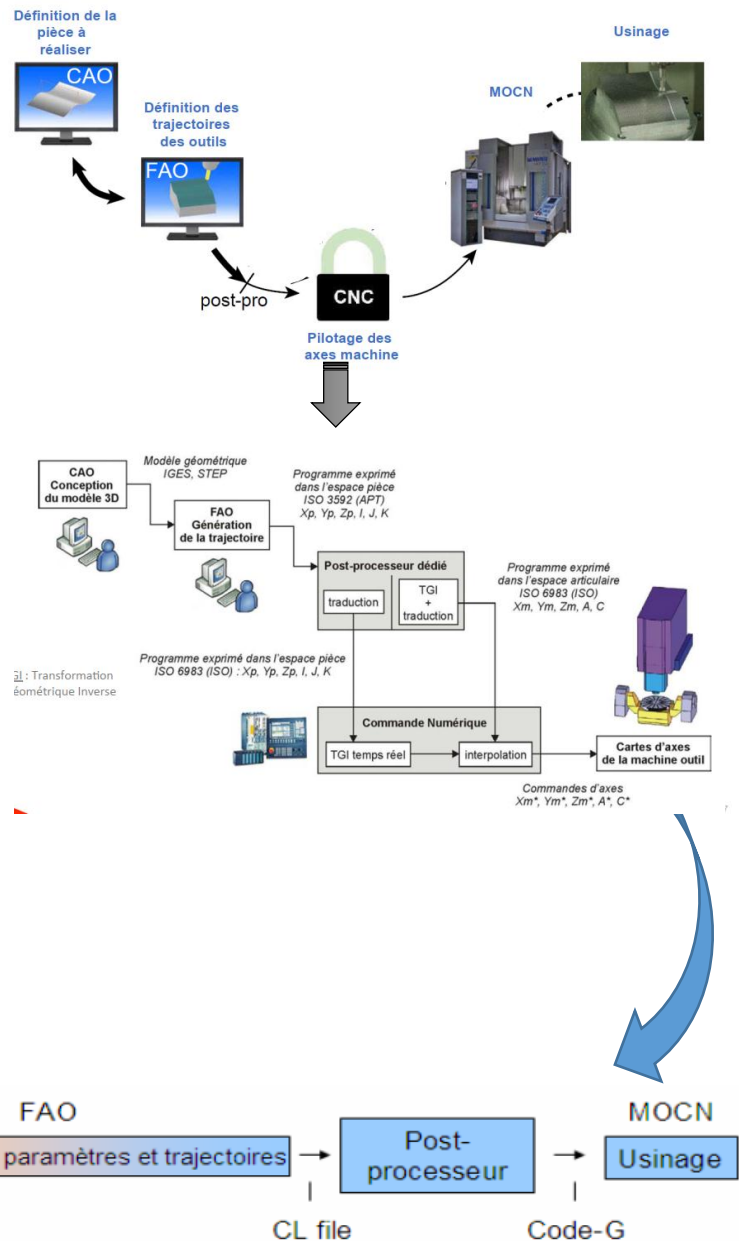


Figure.I.1 : Processus de CFAO.

Chapitre II

Modélisations des courbes

Plan de cours

Objectifs généraux

- Connaissances des étudiants dans le domaine de la CFAO
- Modélisation des pièces de formes complexes
- Simulation du processus d'usinage.
- Interprétation et vérification du programme d'usinage généré automatiquement.

Objectifs de chapitre

- Connaître les formes de représentation des courbes
- Décrire les expressions pour différents types des courbes
- Créer un graphique de type courbe de Bézier

Déroulement

Le chapitre II sera abordé durant trois séances comme suit :

- Les courbes de Bézier
- Les courbes B-Spline
- Les courbes BURBS

Pré acquis

- Notion Mathématiques
- Notion des méthodes numérique

Moyens

- Tableau
- Vidéo projecteur

II.1. Introduction

La discipline de modélisation géométrique est un ensemble de méthodes mathématiques qu'on utilise pour d'écrire la forme d'un objet ou pour exprimer un quelconque processus physique en termes de **métaphore géométrique appropriée**. Ces méthodes comprennent le dessin géométrique assisté par ordinateur, la modélisation des solides, la géométrie algébrique et la géométrie calculatoire. Dans les années 1960, que sont apparus les premières fonctionnalités et les premiers logiciels permettant de modéliser des pièces. Ils étaient conçus pour des applications particulières parmi lesquelles on trouvait l'usinage. Le dessin géométrique assisté par ordinateur applique les mathématiques des courbes et surfaces à la modélisation, essentiellement en utilisant les équations paramétriques de la géométrie différentielle.

La construction des modèles géométrique qui répondent aux problématique physique peut se faire à l'aide opération mathématique tel que :

- L'approximation des points à l'aide de primitives linéaires comme segment droit, triangle pour construire un modèle polygonal.
- La deuxième solution consiste en la construction d'un modèle continue sur la base de courbe et de surface polynomiales comme : **Bézier**, **B-spline**, **NURBS** ou implicite en utilisent les blocs ce surface à squelette.

De façon générale les critères à considérer lors du choix d'un **modèle de représentation** sont nombreux, parmi lesquels on citera la :

- **Généricité** : ensemble des formes
- **L'interactivité** : facilité de contrôle des formes
- **Visualisation** : facilité d'affichage
- **Représentation** : stockage en mémoire et conversion avec d'autres modèles.

II.2. Les formes de représentation des courbes

Si l'on souhaite représenter les courbes d'un objet à l'aide d'équation mathématique et non une série de traits, on peut utiliser des fonctions

II.2.1. Fonction explicite

Une courbe plane en 2D (Fig. II.1.a) peut être décrite par une équation de la forme : $f(x) = y$, cette représentation n'est pas nécessairement inversible, elle ne permet pas de

représenter les courbes fermés en une seule équation. Les courbes explicites dépendent des axes de référence, elle n'est pas très utilisée en CFAO.

II.2.2. Fonction implicite

Une courbe peut être également représentation en 2D (Fig. II.1.b) sous la forme : $f(x, y) = 0$, une courbe implicite est un test caractérisant les points de l'espace selon leur appartenance à une primitive. A l'aide des fonctions implicites on peut définir des courbes de niveau pour faire de l'aplatissement ou l'agrandissement, comme par exemple : $x^2 + y^2 - 1 = 0$ (cercle de centre 0,0)

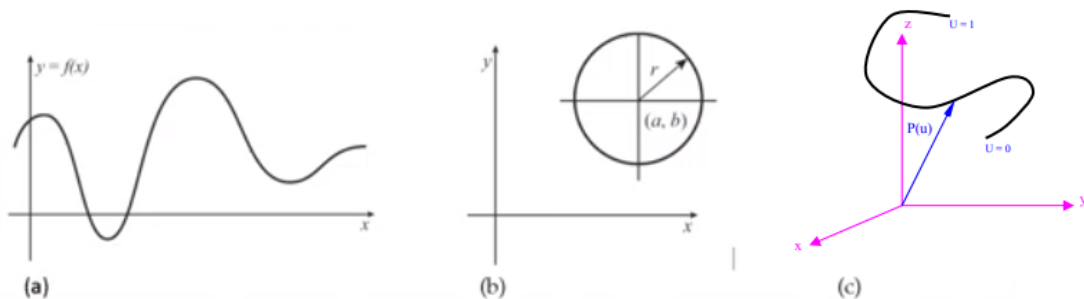
Remarque :

A l'aide de fonction implicite et explicite la représentation universellement employée maintenant c'est la fonction paramétrique polynomiales qui sera étudiée dans ce chapitre.

II.2.3. Courbes paramétriques

Les fonctions paramétriques expriment toutes les variables en fonction d'un ou plusieurs paramètres indépendants (Fig. II.1.c).

- Les points sur la courbe sont définis par un système d'équations
- La dérivée de la courbe en une donne la direction de la tangente
- Une distance paramétrique ne correspond pas nécessairement à une distance en 3D.



$$a) f(x) = y$$

$$b) f(x, y) = 0$$

$$c) P(u) = [x(u), y(u)]$$

Fig. II.1 : Représentation des courbes

On peut décrire par exemple un système de droite à partir de deux points : P_0 et P_1 sous forme paramétrique, Comme suit (Fig. II.2):

$$P(u) = (1 - u)P_0 + uP_1$$

$$P(u) = [x(u), y(u)], \quad u \in [0, 1]$$

Avec u appartenant à l'intervalle unitaire 0 et 1

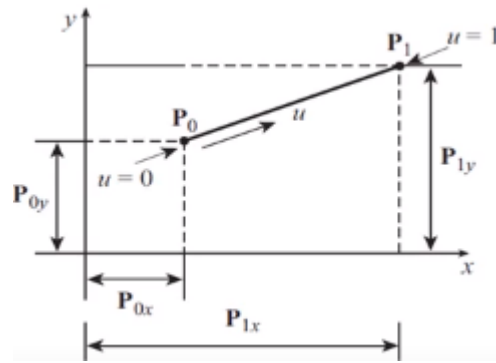


Fig. II.2 : Représentation d'une courbe paramétrique

Dans le cas condition aux limites :

Lorsque : $\begin{cases} \mathbf{u} = 0 \\ \mathbf{u} = 1 \end{cases} \quad \begin{cases} \mathbf{P}(\mathbf{u}) = \mathbf{P}(0) = \mathbf{P}_0 \\ \mathbf{P}(\mathbf{u}) = \mathbf{P}(1) = \mathbf{P}_1 \end{cases} \quad \text{Avec} \quad \mathbf{P}(\mathbf{u}) =$

$$\begin{cases} x(\mathbf{u}) = a_x + b_x u \\ y(\mathbf{u}) = a_y + b_y u \end{cases}$$

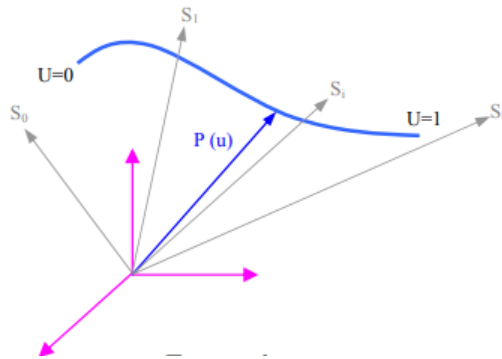


Fig. II.3 : Représentation d'une courbe 3D paramétrique

Dans le cas général pour une courbe 3D de degré 2 ((Fig. II.3) sera définie par cet ensemble des équations:

$$\mathbf{P}(\mathbf{u}) = \begin{cases} x(\mathbf{u}) = a_x + b_x u + c_x u^2 \\ y(\mathbf{u}) = a_y + b_y u + c_y u^2 \\ z(\mathbf{u}) = a_z + b_z u + c_z u^2 \end{cases}$$

Les coefficients $(a_x, b_x, c_x, a_y, b_y, c_y, a_z, b_z, c_z)$ peuvent être calculés à partir des conditions aux limites, Alors pour une surface S la fonction devient (Fig. II.4):

$$\mathbf{S}(\mathbf{u}) = \begin{cases} x(\mathbf{u}, v) \\ y(\mathbf{u}, v) \\ z(\mathbf{u}, v) \end{cases}$$

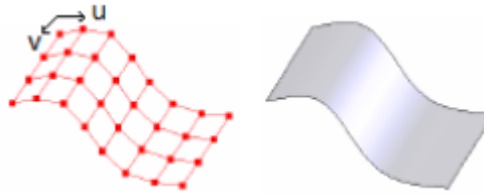


Fig. II.4 : surface paramétrique

Remarque : On utilise rarement les courbes de degré supérieur à 3 raison des oscillations qu'elles introduisant du contrôle local difficile

II.3. La définition de lissage, approximation, interpolation

On rencontre deux types de problèmes sont l'interpolation et l'approximation dans lesquelles la courbe devient plus lisse est plus ou moins astreinte à passer par l'ensemble des points M_i de coordonnées (x_i, y_i, z_i) connues donc la création de formes pour laquelle des contraintes ne sont imposées qu'en quelques points particuliers.

- **Réaliser le lissage** d'une courbe discrète à déterminer une fonction continue qui en donne une représentation satisfaisante.
- **L'approximation** consiste à approcher l'ensemble des points par une courbe qui ne passe pas forcément par tous les points.
- **L'interpolation** consiste à faire passer la courbe par tous les points considérés

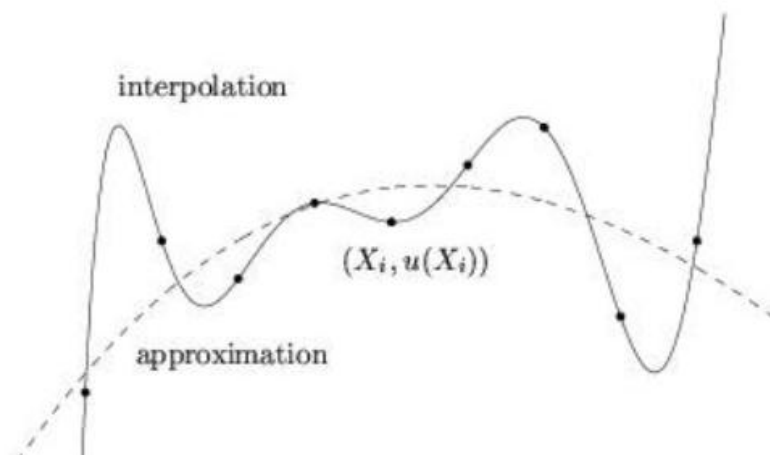


Fig. II.5 : Les méthodes de l'interpolation et l'approximation de courbe paramétrique

Quand le nombre de points augmente l'ordre du polynôme augmente et cela engendre des problèmes de stabilité. D'où la nécessité d'aborder le mode d'interpolation par morceaux ce qui est très intéressant en CFAO.

II.4 Le problème d'interpolation de base :

Il existe plusieurs approches pour trouver le polynôme $P(x)$: méthodes de Newton, méthodes de Lagrange (ils sont généralement abordés dans le cours d'une méthode numérique).

II.5. Les courbes de Bézier

II.5.1. Définition

Les courbes de Bézier sont des courbes polynomiales paramétriques développées pour concevoir des pièces de carrosserie d'automobiles. Elles ont été conçues par Pierre Bézier en 1962 pour Renault. Elles ont de nombreuses applications en CFAO et dans la synthèse d'images. Une courbe de Bézier est un segment de courbe polynomiale de degré n définie à partir d'un ensemble de $n+1$ points $\{P_i\}$ avec $i=0\dots n$, en peut dit (point de contrôle) qui interpole les deux points extrêmes (P_0 et P_n) et qui approxime les autres. La courbe Bézier est définie par un ensemble de points de contrôle avec un minimum de deux. Il peut y en avoir 2, 3, 4 ou plus. Par exemple, la courbe de deux points et trois.

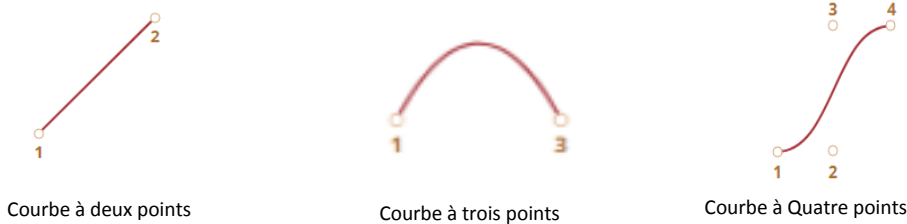


Figure.II.6 : Les courbes Bézier

- Les points ne sont pas toujours sur la courbe.
- L'ordre de la courbe est égal au nombre de points moins un.
- Une courbe est toujours à l'intérieur de l'[enveloppe convexe](#) de points de contrôle (figure.II.7).

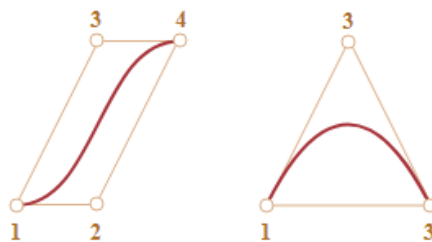


Figure.II.7. Les courbe Bézier à l'intérieur de l'[enveloppe convexe](#)

II.5.2. Théorie générale

On a plusieurs méthodes de construction des courbes Bézier comme : Algorithme de Casteljua, fonction de Bernstein et méthodes matricielle

II.5.2.1 Algorithme de Casteljua

Pour construit progressivement une courbe de Bézier quadratique (deux degré), on à trois points de contrôle P_0, P_1 et P_2 , donc on a deux segment P_0P_1 et P_1P_2 et pour définit les deux point intermédiaires $B_{0,1}(t)$ et $B_{1,2}(t)$ appartenant ces segments, on utilisons les expressions suivantes:

$$B_{0,1}(t) = (1 - t)P_0 + tP_1$$

$$B_{1,2}(t) = (1 - t)P_1 + tP_2$$

Les valeurs t et $1 - t$ représentent les distances des points d'extrémités de l'intervalle, $t = 0$ et $t = 1$. Nous appelons t et $1 - t$ les coordonnées **barycentriques** d'un point par rapport à l'intervalle $t \in [0, 1]$ et $B_{0,1}(t)$ est l'interpolation linéaire entre P_0 et P_1 (courbe Bézier linier) et $B_{1,2}(t)$ est l'interpolation linéaire entre P_1 et P_2 . Une fois change la valeur de t , on va avoir la variation de $B_{0,1}(t)$ sur le segment P_0P_1 et même pour $B_{1,2}(t)$ il va varie de P_1 vers P_2 .

Si on incrémente la valeur de t d'une petite valeur, on va avoir le déplacement de $B_{0,1}(t)$ sur le segment P_0P_1 et le déplacement de $B_{1,2}(t)$ sur le segment P_1P_2 et à partir de segment $B_{0,1}, B_{1,2}$ on peut prendre le point courant $B_{0,1,2}(t)$ correspondant à la valeur de t utilisé. Si on incrément encore une fois la valeur de t on va avoir une nouvelle position de point $B_{0,1}(t)$ sur le segment P_0P_1 et de même pour $B_{1,2}(t)$ sur le segment P_1P_2 a qui donne nouveau segment $B_{0,1}, B_{1,2}$ à partie duquel on prendre la valeur de point courant $B_{0,1,2}(t)$ appartenant à la courbe de Bézier (voir Figure.II.8), on utilisant la formule de $B_{0,1,2}(t)$:

$$B_{0,1,2}(t) = (1 - t)B_{0,1}(t) + tB_{1,2}(t)$$

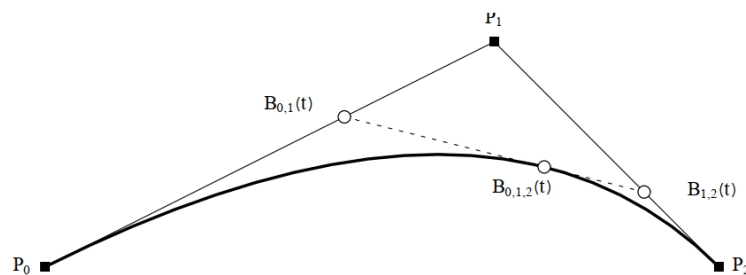


Figure.II.8 : construction d'une courbe de Bézier à 3 points (les valeurs des polynômes

$B_{0,1}(t), B_{1,2}(t)$ et $B_{0,1,2}(t)$ sont exprimés pour $t=2/3$

Les points $B_{0,1}(t)$, $B_{1,2}(t)$ et $B_{0,1,2}(t)$ y sont représentés pour $t = 2/3$, Ce procédé de construction est connu sous le nom d'algorithme de *de Casteljau*.

Exemple 1 :

Prenons les 5 points suivants : $P_1 = (0 \ -1)$, $P_2 = (2 \ -3)$, $P_3 = (5 \ 0)$, $P_4 = (4 \ 3)$ et $P_5 = (0 \ 4)$
En utilisant l'algorithme de *de Casteljau*, Construire une courbe de Bézier à 5 points avec $t=1/3$.

II.6.2.2 Polynôme de Bernstein

La construction se généralise à n points en utilisant les polynômes de **Bernstein**. Donc la courbe de Bézier pour les $(n+1)$ points de contrôle (P_0, \dots, P_n) , est l'ensemble des points définis par la représentation paramétrique :

$$P(t) = \sum_{i=0}^n B_i^n(t) P_i \quad \text{pour } t \in [0; 1]$$

Les polynômes de Bernstein $B_i^n(t)$ sont des scalaires dépendant d'un paramètre t .

Les polynômes de **Bernstein**, nommés ainsi en l'honneur du mathématicien russe **Sergueï Bernstein** (1880-1968), permettent de donner une démonstration constructive et probabiliste du théorème d'approximation de **Weierstrass**. Ils sont également utilisés dans la formulation générale des courbes de Bézier.

Description : Pour un degré $n \geq 0$, il y a $(n+1)$ polynômes de Bernstein : B_0^n, \dots, B_n^n définis, sur l'intervalle $[0 ; 1]$, s'écrit :

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

Ou : n et i sont les coefficients **binomiaux**

Ces polynômes présentent plusieurs propriétés importantes par exemple : non négativité, symétrique et la Partition de l'unité (Cauchy):

$$\sum_{i=0}^n B_i^n(t) = 1$$

.A partir de l'équation précédente, on peut développer le principe de Bernstein. Ou bien en prenant l'identité $[(1-t) + t]$ et en exécutants un développement binomiale de l'expression à la puissance n^{ieme} , on obtient :

$$[(1-t) + t]^n = \sum_{i=0}^n \binom{n}{i} t^i (1-t)^{n-i} = 1 \quad \text{avec} \quad \binom{n}{i} = \frac{n!}{i!(n-i)!}$$

Et : $i \in [0, n]$

Exemples 2: Remplissez les lignes vides

Pour n=1, le polynôme de Bernstein est

$$1^1 = (1-t + t)^1 = ((1-t) + t)$$

Puis on ajoute les n+1 points de contrôle pour chaque coefficient du polynôme pour définissent la courbe de Bézier donnée par l'équation : $P(t) = (1-t)P_0 + tP_1$

C'est donc courbe de Bézier de linéaire un degré pour $t \in [0, 1]$

Pour n=2, le polynôme de Bernstein est :

.....

Puis on ajoute les n+1 points de contrôle pour chaque coefficient du polynôme :

.....

C'est donc courbe de Bézier de deuxième degré pour $t \in [0, 1]$

Pour n=3, le polynôme de Bernstein est

$$1^3 = (1-t + t)^3 = ((1-t) + t)^3 = (1-t)^3 + 3(1-t)^2t + 3(1-t)t^2 + t^3$$

Puis on ajoute les n+1 points de contrôle pour chaque coefficient du polynôme

$$P(t) = (1-t)^3P_0 + 3(1-t)^2tP_1 + 3(1-t)t^2P_2 + t^3P_3$$

C'est donc courbe de Bézier de degré trois pour $t \in [0, 1]$

Finalement, on peut obtenir le polynôme de **Bernstein** mathématiquement de la Figure.II.7 en remplaçant les expressions de $B_{0,1}$ et de $B_{1,2}$ dans $B_{0,1,2}$:

$$B_{0,1,2}(t) = (1-t)^2P_0 + 2t(1-t)P_1 + t^2P_2$$

Cette équation peut également s'interpréter comme une somme pondérée des points P_0 , P_1 et P_2 dépendante du paramètre t . L'importance de cette pondération est représentée dans la **figure II.9** en fonction de la valeur de t .

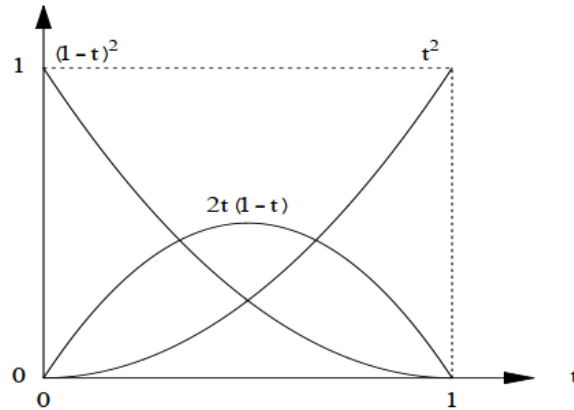


Figure.II.9 : Polynômes de Bernstein de degré 2.

Mais la plupart du temps on utilise des courbes de Bézier de degré trois afin d'assurer un ordre supérieur de continuité et on devra utiliser quatre points de contrôles avec ces quatre points de contrôles on aura une courbe de trois degré défini par les polynômes de Bernstein qui sont donnés par l'expression :

$$B_{0,3}(t) = (1-t)^3, \quad B_{1,3}(t) = 3t(1-t)^2, \quad B_{2,3}(t) = 3t^2(1-t), \quad B_{3,3}(t) = t^3$$

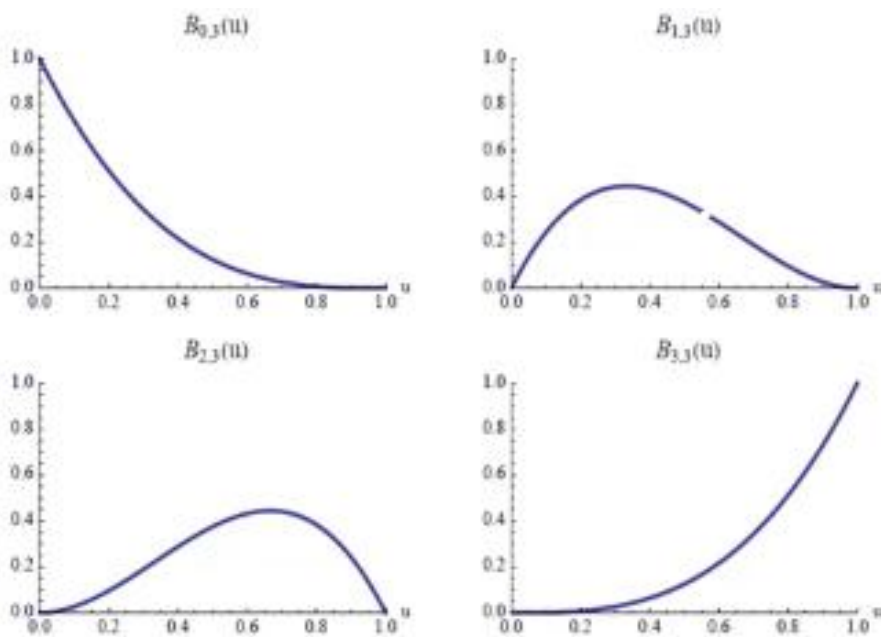


Figure.II.10 : Polynômes de Bernstein de degré 3.

Donc on peut écrire la forme générale de la courbe Bézier de degré 3 comme suit :

$$P(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3$$

II.5.2.3 Expression matricielle des courbes de Bézier

Une courbe de Bézier peut s'écrire sous la forme matricielle de la façon suivante :

$$B(t) = \sum_{i=0}^n b_i^n(t) P_i$$

$$= [P_0, P_1, \dots, P_n] \begin{bmatrix} b_0^n(t) \\ b_1^n(t) \\ \vdots \\ b_n^n(t) \end{bmatrix}$$

Or, la courbe de Bézier étant un polynôme de degré n , on souhaite obtenir une écriture utilisant explicitement les puissances de t . En développant l'expression de :

$$b_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

En utilisant la formule du binôme de Newton sur l'identité $(1-t)^{n-i}$:

$$\begin{bmatrix} b_0^n(t) \\ b_1^n(t) \\ \vdots \\ b_n^n(t) \end{bmatrix} = M_n \begin{bmatrix} 1 \\ t \\ \vdots \\ t^n \end{bmatrix}$$

Où M_n est la matrice dont les coefficients binôme (Cette matrice est standard pour chaque degré).

Cette expression est intéressante car les modèles de courbes paramétriques peuvent s'écrire sous forme matricielle, ce qui permet d'en avoir une approche générique lorsqu'il s'agit de les évaluer.

Exercice 3 : Ecrire la courbe Bézier de degré 3 sous forme matricielle.

II.5.2.4 Courbes de Bézier rationnelles

Bien que pratiques et intuitives, les courbes de Bézier souffrent de nombreux problèmes. Le plus important est qu'elles ne permettent pas de tracer des courbes aussi simples que des arcs de cercles comme le précise le théorème suivant.

Théorème :

Il n'existe pas de courbe paramétrée polynomiale qui décrive un arc de cercle. Pour pallier à ce problème, on peut généraliser les courbes de Bézier.

Définition Courbes de Bézier rationnelles :

On se donne $n + 1$ points P_i et pour chaque point un poids : $w_i \geq 1$. On définit alors :

$$P(t) = \frac{\sum_{i=0}^n B_i^n(t) P_i w_i}{\sum_{i=0}^n B_i^n(t) w_i}$$

II.6. Les courbes B-spline

Les courbes de Bézier sont des outils puissants mais dont le principal inconvénient est que le degré des courbes augmente avec le nombre de points de contrôle ; leur cout de calcul devient donc cher dès que le nombre de points de contrôle devient important (on ne dépasse pas 10 points de contrôle en pratique).

Une solution aux inconvénients des courbes de Bézier

- **les fonctions de base** ne doivent pas dépendre du nombre de points de contrôle
- la courbe doit être modifiable localement.

L'idée des courbes **Splines** est qu'une courbe complexe peut être obtenue en construisant une **fonction par morceau** à partir d'un ensemble des courbes simples. Un intérêt immédiat de ce mode de construction est que le changement de la position d'un point de contrôles n'aura plus qu'un impact limité sur la fonction obtenue.

II.6.1. Définition

Les courbes B-splines sont des familles de courbes à paramètres très utilisées dans les logiciels de conception assistée par ordinateur (CAO). Elles sont définies en raccordant des morceaux de courbes polynômiales de sorte que le raccord soit suffisamment régulier.

- En mathématiques appliquées, une spline est une fonction définie par morceaux par des polynômes. Une B-spline peut être utilisée pour les problèmes d'interpolation, d'approximation ou pour générer des formes complexes.
- Les courbes B-splines sont plusieurs courbes de Bézier se rejoignent bout à bout
- La courbe B-spline est définie de la même manière que la courbe de Bézier (Bernstein)

Les polynômes de cette Spline doivent se raccorder aux différents points de jonction :

- Donc les courbes Spline diffèrent des courbes de Bézier puisque elles sont **composées de plusieurs segments des courbes** ce qui permet d'avoir un contrôle local sur l'allure de la courbe et ceci constitue le point de contrôle.
- Le degré d'une courbe Spline ne dépend pas du nombre de points de contrôle, il est défini séparément.

- Une particularité de Spline est de permettre l'interpolation d'un ou plusieurs points de contrôles
- Contrairement aux courbes de Bézier, les Splines n'interpolent pas nécessairement les points de contrôle aux extrémités (**comme montrer sur la figure ci-dessous**)

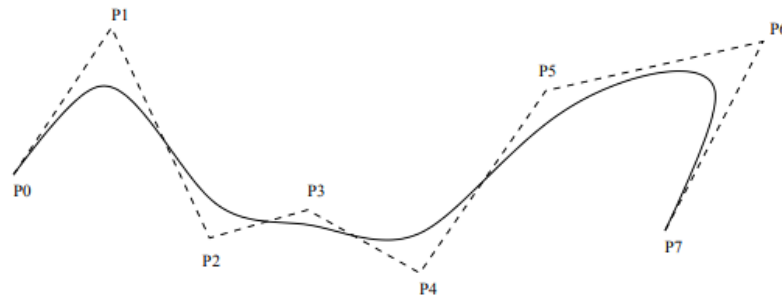


Figure.II.11. Une courbe B-spline constitué en 7 morceaux de degré 6 avec 8 points de contrôle

II.6.2. Théorie générale

a) La continuité de la courbe : La première dérivation à la fin de la première courbe de Bézier est la même que la dérivée première au début de la deuxième courbe de Bézier en fonction de paramètre t .

b) Définition de la courbe B-spline : La courbe B-spline est définie par :

- $n + 1$, le nombre de fonctions de contrôle
- k , le degré de la courbe.
- $k - 1$ C'est l'ordre de la courbe)
- $\{P_0, \dots, P_{n+1}\}$ les $n + 1$ points de contrôle
- $[t_0, \dots, t_{k+n+1}]$ un vecteur nodal de valeurs de paramètres
- n intervalles $\Rightarrow n + 1$ points de contrôles
- k est le degré de la courbe, donc $k + 1$ est le nombre de points de contrôles qui influencent un seul intervalle. $1 \leq k \leq n$
- Il y a $k + n + 2$ nœuds avec : $t_i \leq t_{i+1}$

Une spline polynomiale de degré k est la réunion d'un ensemble de courbes polynomiales de degré inférieur ou égal à k (k entier non nul fixé) se raccordant avec une continuité qui peut être différente entre les points de jonction. Nous utilisons *la base B-spline*. Ainsi une courbe spline, écrite dans *la base des Spline* s'exprime d'une manière semblable à une courbe de Bézier. Une courbe B-spline non uniforme de degré k est donné par :

$$P(u) = \sum_{i=0}^n N_{i,k}(t) P_i \quad \text{pour } t \in [0, n - k + 2]$$

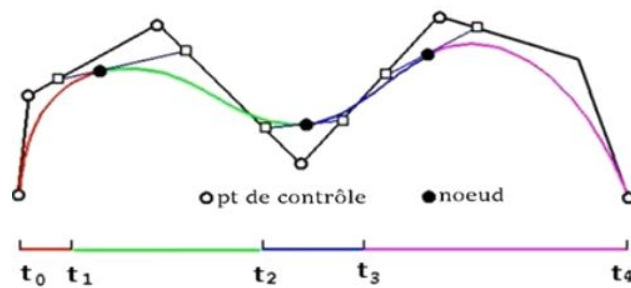


Figure.12. Les courbes B-spline

- On a $n + 1$, point de contrôle, l'ordre des polynômes est $k-1$.
- Les $N_{i,k}(t)$ sont *les fonctions de base des spline (B-spline)* définie pour *le vecteur nodal* $[t_0, t_1, \dots, t_{n+k+1}]$

Les $N_{i,k}(t)$ sont définies récursivement à l'aide de *l'algorithme de De Boor* comme suit :

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

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

N.B

1. Le dénominateur des fonctions de base peut être nul, dans ce cas on dit que la fraction vaut 0 ($0/0 = 0$)
2. La courbe B-spline est identique à une courbe Bézier quand $(n + 1) - (k + 1) = n - k + 2 = 1$, cela veut dire qu'on a un seul segments de courbe dans la B-spline.
3. Les fonctions de base $N_{i,k}$ vérifient la condition :

$$\sum_{i=0}^n N_{i,k}(t) = 1$$

II. 6.3. Exemple :

On cherche à calculer $p(u)$ pour différentes valeurs de k , avec six points de contrôle ($n=5$), Dans un premier temps, il faut calculer les valeurs des nœuds pour ensuite évaluer les fonctions de base.

Cas : $n=5$ et $k=3$

On cherche à construire une B-spline quadratique ($k=3$) sur la base de six points de contrôles ($n=5$). On a :

$$\begin{cases} n - k + 2 = 4 \text{ nombres des segments} \gg u \in [0 ; 4] \\ n + k + 1 = 9 \gg \text{Il y aura 8 nœuds } t_i \end{cases}$$

- Courbe B-spline non-uniforme non-rationnelle pour $n=5$ et $k=3$
- Fonction de base ($n=5$ et $k=3$)

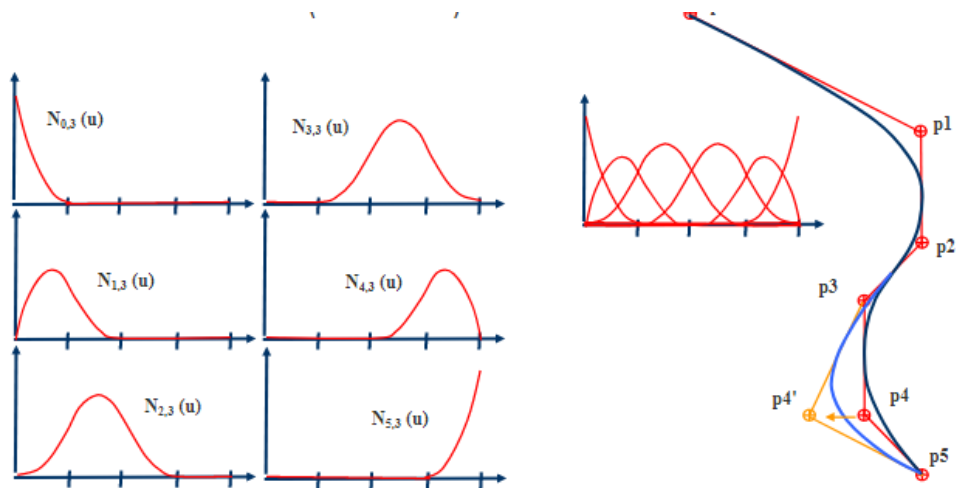


Figure II.13. Représentation Courbes B-SPLINE quadratique

II.7. Courbes NURBS

II.7.1. Définition :

Lorsque des vecteurs de nœuds non uniformes sont utilisés, ces B-Spline rationnelles sont connues sous le nom de **NURBS** (*Non-Uniform knot vector Rational B-Spline*). La formule des courbes NURBS possède de grandes correspondances avec celle des B-splines. Elle est simplement généralisée afin d'être appliquée à des coordonnées homogènes.

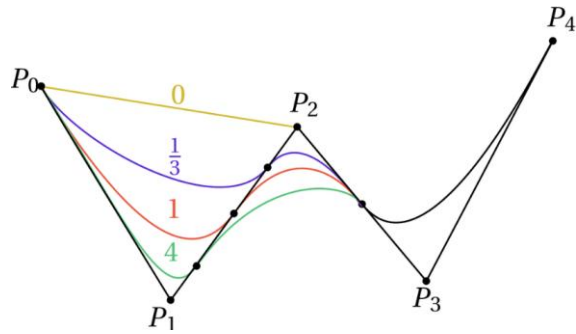


Figure.II.14 : Courbe NURBS avec variation de la coordonnée homogène

Les NURBS peuvent représenter de façon exacte toutes les formes polygonales, toutes les coniques, toutes les courbes polynomiales paramétriques par morceaux ainsi que toute courbe complexe. Ces courbes d'une grande capacité de description sont utilisées par les modeleurs des logiciels de CAO. Différents poids associés aux P_i modifient le facteur d'attraction

$$p(u) = \sum_{i=0}^n N_{i,k}(u) \mathbf{P}_i$$

$$\mathbf{P}_i = \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \xrightarrow{\text{Multiplication des coordonnées de } \mathbf{P}_i \text{ avec un poids } w} \mathbf{P}_i^w = \begin{bmatrix} w_i x_i \\ w_i y_i \\ w_i z_i \\ w_i \end{bmatrix}$$

$$p^w(u) = \sum_{i=0}^n N_{i,k}(u) \begin{bmatrix} w_i x_i \\ w_i y_i \\ w_i z_i \\ w_i \end{bmatrix}$$

Passage en coordonnées Cartésiennes en divisant $p^w(u)$ avec la 4^{ème} coordonnée, Ainsi la formule d'une courbe NURBS est illustré de la manière suivante :

$$p^w(u) = \sum_{i=0}^n \frac{N_{i,k}(u)w_i}{\sum_{j=0}^n N_{j,k}(u)w_j} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} = \frac{1}{\sum_{i=0}^n N_{i,k}(u)w_i} \sum_{i=0}^n N_{i,k}(u)w_i \mathbf{P}_i$$

$$R_{i,k}(u) = \frac{N_{i,k}(u)w_i}{\sum_{j=0}^n N_{j,k}(u)w_j} \quad \text{et} \quad p(u) = \sum_{i=0}^n R_{i,k}(u) \mathbf{P}_i$$

Où w_i sont les poids des coordonnées homogènes des points de contrôle donnés P , n le nombre de nœuds, k le degré de la NURBS, $N_{i,k}$ les coefficients calculés selon l'algorithme de Cox-de Boor, et $u \in [0,1]$ le paramètre.

Remarque : Si tous les poids sont égaux à 1, la courbe NURBS est réduite à une courbe B-spline donc Les NURBS sont rationnelles

II.7.2. Propriétés des fonctions de base NURBS

Les propriétés géométriques des Courbes NURBS :

- En choisissant correctement les points de contrôle et les poids, toute conique peut être représentée exactement par une NURBS.
- La transformée affine d'une courbe NURBS est la courbe passant par la transformée des points.
- Contrairement aux courbes B-spline, l'image d'une courbe NURBS par une projection est la courbe NURBS passant par la projection des points. Les poids doivent être recalculés en fonction de la matrice de projection.

Chapitre III

Modélisations des surfaces

Plan de cours

Objectifs généraux

- Connaissances des étudiants dans le domaine de la CFAO
- Modélisation des pièces de formes complexes
- Simulation du processus d'usinage.
- Interprétation et vérification du programme d'usinage généré automatiquement.

Objectifs de chapitre

- Décrire les expressions pour différents types des surfaces
- Créer une surface de Bézier

Déroulement

Le chapitre II sera abordé durant trois séances comme suit :

- Les surfaces Bilinéaires
- Les surfaces de Bézier
- Les surfaces B-Spline et les surfaces de révolution

Pré acquis

- Notion Mathématiques
- Notion des méthodes numérique

Moyens

- Tableau
- Vidéo projecteur

III.1. Introduction

Historiquement, la modélisation des surfaces c'est la deuxième technique apparue mais c'est en fait la première qui a permis une approche 3D. En effet, avec la modélisation surfacique, la pièce est décrite à l'aide des surfaces qui en constituent l'enveloppe. Les avantages de l'utilisation des surfaces sont :

- Une description exacte de l'enveloppe de la pièce.
- Un usinage possible de la pièce en commande numérique.
- Une meilleure visualisation de la pièce.

La modélisation des surfaces est un aspect technique très important dans de nombreux domaines (conception des formes en automobile, aéronautique, ...), lorsque les surfaces des objets sont définies à partir des réseaux de points obtenus d'après des maquettes par machine à mesurer.

III.2. Les surfaces Bilinéaires

On appelle surface Bilinéaire, une surface décrite par interpolation linéaire de 4 points (Figure III.1(a)). Ces quatre points définissent la matrice de contrôle de la surface [P], tel que :

$$[P] = \begin{bmatrix} P(0,0) \\ P(1,0) \\ P(0,1) \\ P(1,1) \end{bmatrix}$$

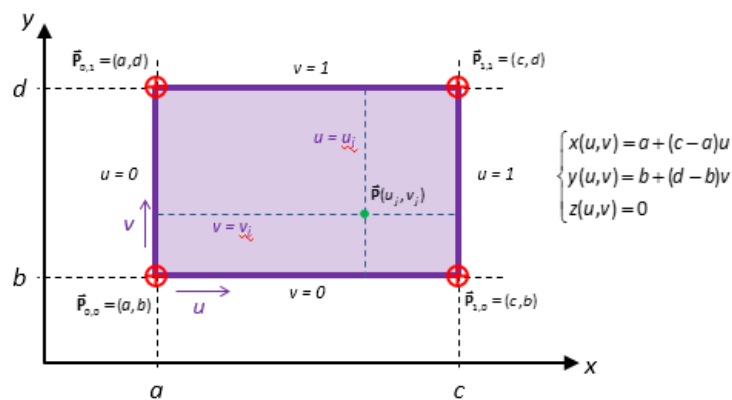


Figure.III.1 : Les surfaces Bilinéaires

L'interpolation linéaire de ces quatre points s'écrit sous la forme suivante :

$$S(u, v) = \begin{pmatrix} 1-u & u & 1 \end{pmatrix} \begin{pmatrix} -P(0,0) & -P(0,1) & P(0,v) \\ -P(1,0) & -P(1,1) & P(1,v) \\ P(u,0) & P(u,1) & P(0,0) \end{pmatrix} \begin{pmatrix} 1-v \\ v \\ 1 \end{pmatrix}$$

Sous forme matricielle, elle s'écrit sous la forme suivante :

$$[P] = \begin{bmatrix} (1-u)(1-v) & u(1-v) & (1-u)v & uv \end{bmatrix} \begin{bmatrix} P(0,0) \\ P(1,0) \\ P(0,1) \\ P(1,1) \end{bmatrix} \quad \text{Avec } 0 \leq u \leq 1 \text{ et } 0 \leq v \leq 1$$

La surface bilinéaire est une surface qui est simple à gérer. L'inconvénient de cette surface c'est l'absence de la flexibilité : par exemple pour décrire une surface de révolution, il faut plusieurs carreaux bilinéaires (Figure III.2(b))

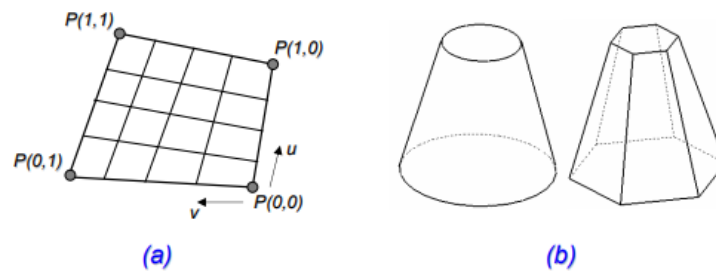


Figure III.2 : (a) Surface Biliéaire, (b) Modélisation d'un cône à l'aide de surface bilinéaires.

III.3. Surfaces de Bézier

III.3.1 Liens entre interpolation bilinéaire et Bézier

De même qu'une courbe de Bézier se construit à partir de successions d'interpolations linéaires de Bézier, une surface de Bézier se construit à partir de successions d'interpolations bilinéaires. Si on considère 4 points P_{00} , P_{10} , P_{01} et P_{11} de R^3 , alors l'interpolation bilinéaire entre ces points forme une surface :

$$B(u, v) = u \cdot (v \cdot P_{00} + (1-v) \cdot P_{01}) + (1-u) \cdot (v \cdot P_{10} + (1-v) \cdot P_{11})$$

La surface est la surface de Bézier engendrée par les 4 points P_{00} , P_{10} , P_{01} et P_{11} (voir figure III.3).

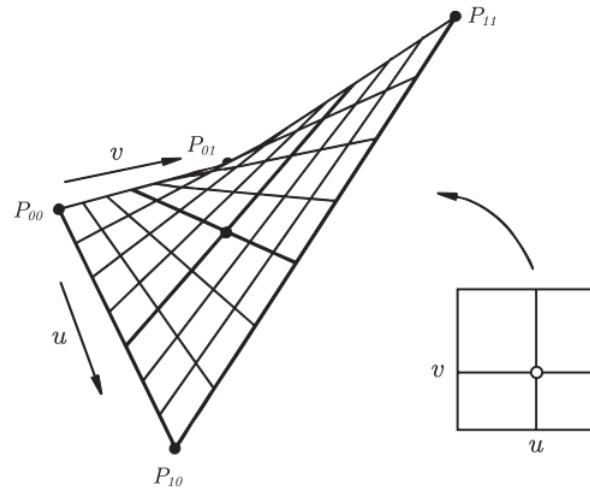


Figure.III.3 : Une surface des Bézier à quatre points coïncide avec l'interpolation bilinéaire.

III.3.2. Définition des surfaces de Bézier (Bernstein)

L'équation ci-dessous se réécrit sous forme matricielle :

$$B(u, v) = [1 - u \quad u] \cdot \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} \cdot \begin{bmatrix} 1 - v \\ v \end{bmatrix}$$

On reconnaît dans les matrices de gauche et de droite les polynômes de Bernstein vus dans la partie sur les courbes de Bézier ($u = b_1 0(u)$ et $1 - u = b_1 1(u)$). Si on considère maintenant des grilles $(n + 1) \times (p + 1)$ de points P_{ij} , alors cette écriture se généralise comme :

$$B(u, v) = [b_0^n(u) \quad \dots \quad b_n^n(u)] \cdot \begin{bmatrix} P_{00} & \dots & P_{0p} \\ P_{n0} & \dots & P_{np} \end{bmatrix} \cdot \begin{bmatrix} b_0^p(v) \\ \vdots \\ b_p^p(v) \end{bmatrix}$$

Et peut également s'écrire comme une double somme (Figure.III.4) :

$$B(u, v) = \sum_{i=0}^n \sum_{j=0}^p P_{ij} \cdot b_i^n(u) \cdot b_j^p(v)$$

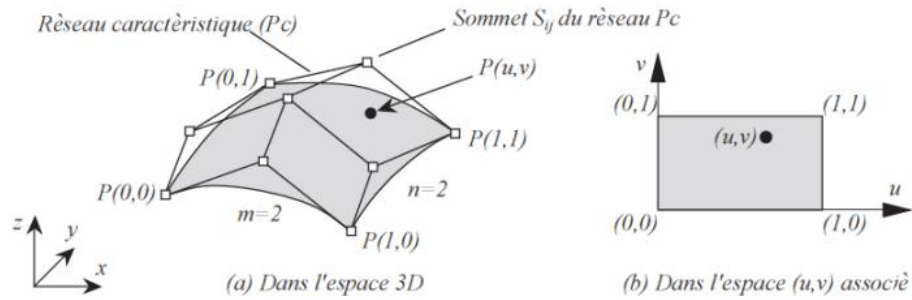


Figure.III.4 : Notations pour les surfaces de Bézier. a) un carreau de Bézier est définie dans l'espace 3D, b) et dans l'espace bi-paramétrique associé

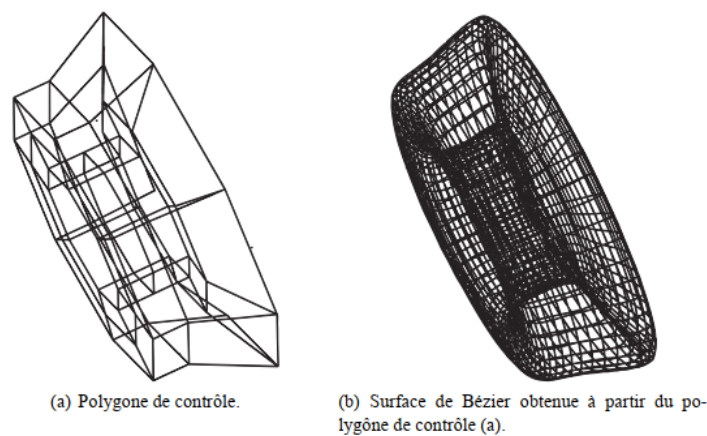


Figure.III.5 : Exemples de surfaces de Bézier représentées avec leurs polygones de contrôle

Les propriétés des surfaces de Bézier découlent directement de celles des courbes de Bézier :

- Une surface de Bézier est invariante par transformation affine
- La surface est construite (de Casteljau) par compositions successives d'interpolations bilinéaires et linéaires, comme ces opérations sont invariantes par transformation affine, il en va de même pour leur composition.
- Une surface de Bézier est contenue dans l'enveloppe convexe formée par ses points de contrôle.
- Par conséquent, l'équation 2.4 nous montre qu'une surface de Bézier est une combinaison barycentrique des points de contrôles P_{ij} avec des poids compris entre 0 et 1. Ceci implique que le barycentre obtenu est nécessairement "entre" les P_{ij} et donc que ceux-ci délimitent la zone où ce barycentre peut se situer.
- Les bords d'une surface de Bézier sont des courbes de Bézier dont les extrémités passent par les points P_{00} , P_{0p} et P_{np} .

- Le déplacement d'un point de contrôle entraîne une déformation de toute la surface.

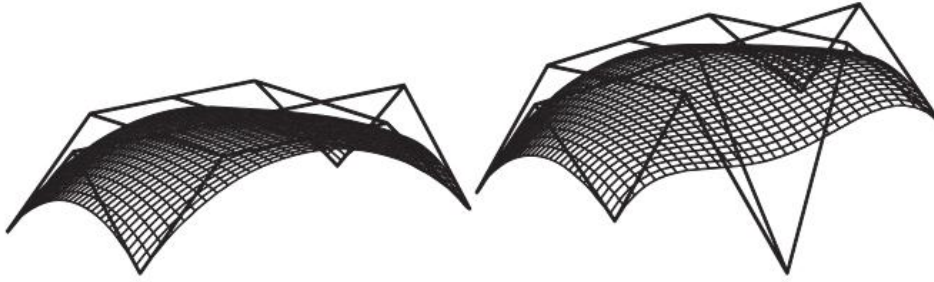


Figure.III.6 : Effet du déplacement d'un point de contrôle sur une surface de Bézier

III.4. Surfaces B-Spline

Pour les continuités d'ordre supérieur, il suffit donc de considérer le degré de continuité des courbes $B_i \cup B_{i+1}$ pour tout i . De manière similaire à ce qui avait été fait avec les courbes de Bézier pour définir les B-splines, on peut utiliser des surfaces de Bézier de degré fixe pour engendrer des surfaces complexes en conservant des degrés raisonnables pour les polynômes.

L'introduction des concepts de séquence nodale et de multiplicité de nœuds, et l'utilisation de fonctions polynomiales par morceaux $N_{im}(u)$ et $N_{jn}(v)$ à la place des polynômes de Bernstein, permettent de généraliser le modèle de Bézier. Ainsi, le modèle B-spline est défini par l'équation :

$$P(u, v) = \sum_{i=0}^r \sum_{j=0}^s S_{ij} N_{im}(u) N_{jn}(v);$$

$$u \in [u_0, u_{r+m+1}], \quad v \in [v_0, v_{s+n+1}]$$

III.5. Surfaces B-Spline rationnelle (NURBS)

Les surfaces NURBS sont basées sur des splines ou des courbes de Bézier. Par conséquent, le paramétrage (degré, points de lissage, sommets de contrôle, épaisseur et nœud) est important pour définir une surface ou une courbe NURBS. Les splines sont optimisées pour créer des surfaces NURBS, vous permettant ainsi de contrôler un grand nombre de ces options. La figure ci-dessous illustre les sommets de contrôle qui s'affichent lorsque vous sélectionnez une spline ou une surface NURBS. L'équation d'une surface B-Spline rationnelle (NURBS) est de la forme :

$$P(u, v) = \frac{\sum_{i=0}^r \sum_{j=0}^s h_{ij} S_{ij} N_{im}(u) N_{jn}(v)}{\sum_{k=0}^r \sum_{l=0}^s h_{kl} N_{km}(u) N_{ln}(v)}$$

$$u \in [u_0, u_{r+m+1}], v \in [v_0, v_{s+n+1}], h_{ij} > 0 \forall i, j.$$

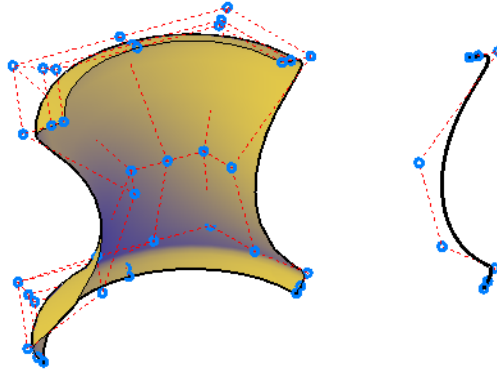


Figure.III.7 : Surfaces B-Spline rationnelle (NURBS)

III.6. Surfaces de révolution

Les surfaces de révolution sont un moyen simple d'engendrer des surfaces à partir de polygone ou de courbes planaires. On considère une courbe $P(t)$ défini par :

$$P(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$$

Alors la surface de révolution obtenue par rotation de cette courbe autour de l'axe des x a pour équation (voir aussi la Figure.III.8) :

$$S(t, \phi) = \begin{bmatrix} x(t) \\ y(t) \cdot \cos \phi \\ y(t) \cdot \sin \phi \end{bmatrix}$$

Cette expression peut se réécrire comme un produit matriciel en coordonnées homogènes :

$$S(t, \phi) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \phi & 0 & 0 \\ 0 & \sin \phi & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x(t) \\ y(t) \\ 0 \\ 1 \end{bmatrix}$$

Propriétés des surfaces de révolution

- Obtenue par révolution d'une courbe génératrice autour d'un axe de révolution.
- Génératrice égale à la courbe déplacée qui balaie la surface.
- L'intersection d'un plan perpendiculaire à l'axe de révolution qui coupe la surface fourni un cercle de rayon nommé parallèle.

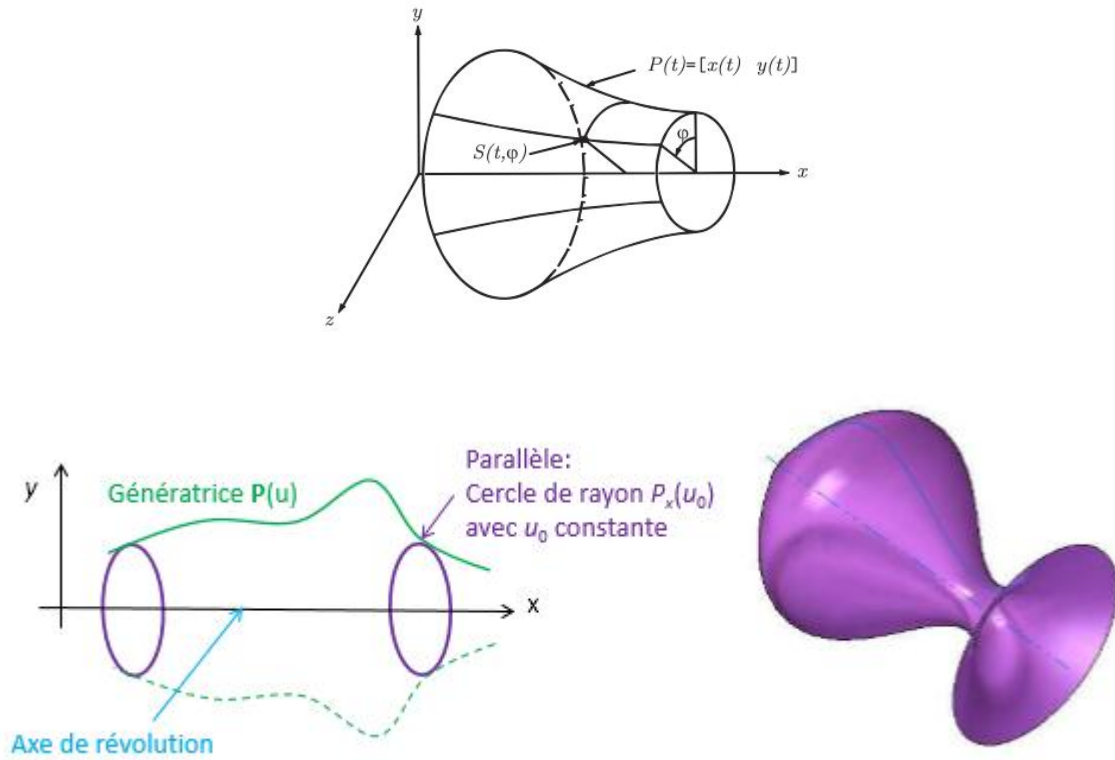


Figure.III.8 : Exemple de surface de révolution définie à partir d'une courbe de Bézier.

a) figure de base, b) surface de révolution

Chapitre IV

Modélisations des solides

Plan de cours

Objectifs généraux

- Connaissances des étudiants dans le domaine de la CFAO
- Modélisation des pièces de formes complexes
- Simulation du processus d'usinage.
- Interprétation et vérification du programme d'usinage généré automatiquement.

Objectifs de chapitre

- Connaitre la modélisation des solides
- Représentation les modèle

Déroulement

Le chapitre II sera abordé durant une séance comme suit :

- Modélisations des solides

Pré acquis

- Notion Mathématiques
- Notion des méthodes numérique

Moyens

- Tableau
- Vidéo projecteur

IV. Modélisations des solides

IV.1. Introduction

La CAO permet la modélisation géométrique tridimensionnelle et la génération d'un modèle dont la définition mathématique est compatible avec les différentes activités de CFAO et permet l'animation d'images réelle de l'objet.

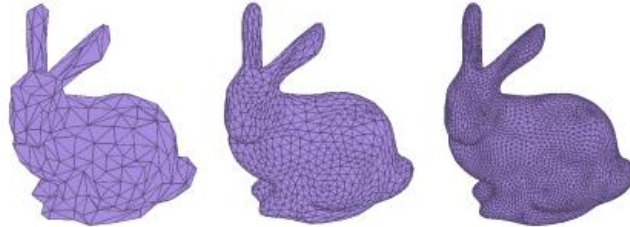


Figure.IV.1 : Modélisations des solides

La définition de la géométrie 3D d'un objet se fait selon trois types de modélisation qui sont :

- **Modèle Fil de fer** : C'est la méthode de base pour représenter la géométrie d'un objet à travers ses côtés, ceci est accompli par le stockage de ses cotés sous formes de lignes, arcs, cercles, courbes...etc. Par exemple, avec la modélisation fil de fer (*Wire Frame*), un cube est défini uniquement par ses arêtes (figure.V.2). La pièce n'est donc qu'imparfaitement définie puisqu'il manque la définition exacte des faces et de l'intérieur du cube.
- **Modèle Surfactive** : Ce mode permet la représentation du modèle 3D par des surfaces stockées et visualisées sur l'ordinateur. En effet, avec la modélisation surfactive, la pièce est décrite à l'aide des surfaces qui en constituent l'enveloppe : un cube est ainsi décrit par ses 6 faces (figure.V.2). La pièce est ainsi bien décrite, les coupes que l'on peut être amené à faire sont des coupes exactes.
- **Modèle Volumique** : La modélisation la plus complète donne les propriétés géométriques de l'objet solide (la masse, le volume, le centre de gravité les différentes vues, les sections et les coupes).

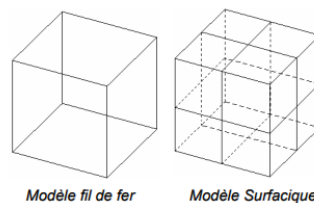


Fig. IV.2 : Différents modes de représentation (Modèle fil de fer, Modèle Surfactive) d'un cube

En modélisation volumique, la notion de topologie déviant primordial pour la modélisation de solides, Le modèle volumique ou solide permet de définir la notion d'intérieur et d'extérieur et donc d'introduire la notion de matière, en conséquence on peut :

- Effectuer un calcul de masse ou de surface mouillée
- Déterminer le centre et la matrice d'inertie nécessaires à une simulation de dynamique multi-corps
- Effectuer un calcul de structures nécessitant de passer par le maillage du volume
- Bénéficier de la fonctionnalité si importante de détection d'interpolation de pièces lors d'une simulation cinématique
- générer des plans 2D avec hachurages automatique.

IV.2. Définition modèle volumique

Modèle 3D solide généré à partir d'un ensemble d'éléments auxquels on applique des opérations booléennes et des transformations géométriques. Dans un modèle volumique, le solide est représenté par l'espace 3D occupé, il existe deux approches de base pour obtenir un modèle volumique :

1. **Modèle CSG** : La géométrie de construction de solides
2. **Modèle B-rep** boundary representation

Dans les logiciels de CAO actuels les modèles CSG et B-rep collaborent souvent à cause de leur complémentarité. Le système CAO traduit généralement une modélisation CSG en représentation géométrique BREP lors d'une mise à jour de l'arbre de construction (fonction update). C'est pourquoi il est nécessaire d'avoir effectué une mise à jour, sans erreur, avant un export dans un format neutre pour s'assurer que la représentation BREP soit conforme à l'intention de conception. Dans ce chapitre on aborde les principaux modèles volumiques (CSG, B-Rep, ...) utilisés dans les logiciels de CFAO pour établir les modèles solides 3D

IV.2.1 La géométrie de construction de solides

IV.2.1.1. Définition

La géométrie de construction de solides (CSG en anglais : "Constructive Solid Geometry") est une branche de la modélisation des solides.

Cette technique de modélisation géométrique concerne la représentation d'un objet solide comme combinaison d'objets solides simples (cylindre, sphère, cône, tore) comme montré sur la figure à l'aide d'opérations géométriques booléennes (union, intersection,

soustraction). On applique des opérations booléennes et des transformations géométriques classiques (figure.V.3)

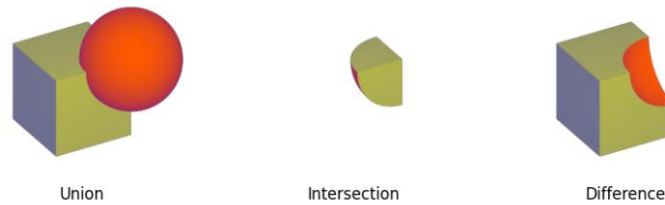


Figure IV.3 : Les opérations géométriques booléennes

Les trois opérations de base pour la géométrie solide constructive. **L'union (U)** combine les deux formes, **l'intersection (\cap)** ne renvoie que le volume commun des deux espaces et **la différence (-)** soustrait la deuxième forme de la première.

Les objets en géométrie de construction de solides peuvent être représentés par des arbres binaires, dans lesquels les feuilles représentent les primitives, et les nœuds représentent les opérations. L'utilisation d'arbre permet de faire très facilement des modifications (voir la **Figure .V.4**).

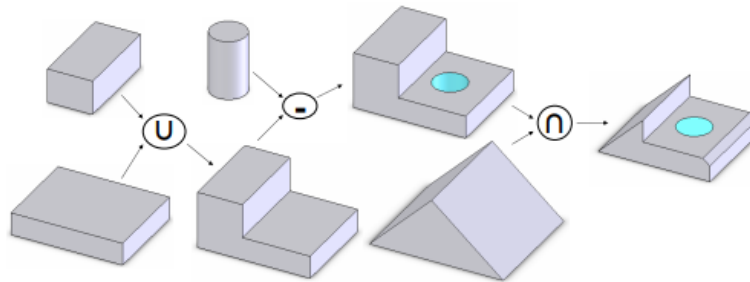


Figure.IV.4 : La composition arborescente du solide CSG

IV.2.1.2. Avantage du modèle CSG :

- facilité de création ;
- facilité de stockage ;
- Il présente quelques facilités de modification.

IV.2.1.3. Les inconvénients du modèle CSG :

- Difficultés de calcul des propriétés géométriques et physiques d'un objet complexe.
- Difficultés pour effectuer certaines conversions (arbre CSG vers le modèle de représentation par les frontières par exemple).
- Non unicité : une pièce mécanique possède plusieurs représentations CSG.

IV.2.2. La représentation par les frontières (Modèle B-Rep)

Cette représentation est très utile en CAO pour l'usinage des surfaces, pour le calcul rapide des masses, des centres de gravité... ; elle est aussi utile en infographie, où la plupart des algorithmes de visualisation supposent connues les surfaces des objets. Les modèles **B-Rep** sont composés de deux techniques :

a) La représentation facettée : Lorsqu'on utilise des surfaces planes (triangles, polygones, ...) pour modéliser un objet, une approximation polygonale est nécessaire ; on obtient un ensemble de faces représentées par leurs bords et leurs coins, la modélisation est dite **facettée**. La frontière est décrite par ensemble de faces planes, d'arêtes rectilignes et de sommets (figure.V.5).

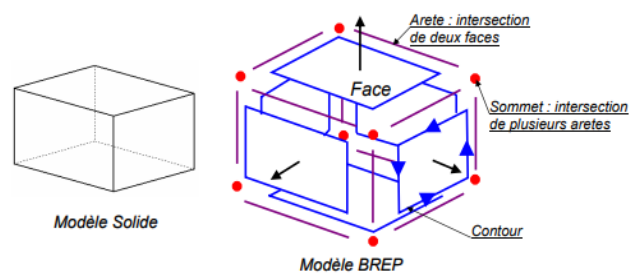


Figure.IV.5 : La représentation facettée

b) La représentation par les frontières exactes : Pour une description exacte des faces non classiques (planes, cylindriques...), les surfaces complexes sont utilisées dans de nombreux systèmes de CAO, bien que leurs manipulations soient plus compliquées. Pour garantir la validité des solides, on utilise les opérateurs d'Euler. Le principe est basé sur la loi d'Euler valable pour tout polyèdre régulier convexe simple :

$$V - E + F = 2$$

Où V est le nombre de sommets, E le nombre d'arêtes et F le nombre de faces. On peut généraliser la loi pour les solides :

$$V - E + F = 2(S - H) + R$$

S est le nombre de composants disjoints dans le solide, H est le nombre de trous dans le solide et R le nombre de cavités dans une face. **La représentation par les frontières exactes, mode voisin du précédent, mais les faces et les arêtes sont de type courbes NURBS.**

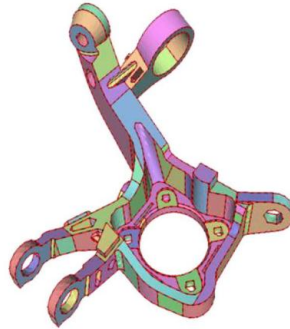


Figure.IV.6 : La représentation par les frontières (Modèle B-Rep)

Pour valider un modèle BREP, chaque face doit vérifier un ensemble de propriétés, qui assurent la reconnaissance de l'objet en tant que volume occupant une portion de l'espace ainsi que son homogénéité. En fait le modèle BREP décrit les frontières comme une juxtaposition de plusieurs faces orientées. Chaque face est composée d'une surface (entité qui décrit la géométrie de la face) et est bornée par au moins un contour fermé et orienté. Chaque contour est formé d'un ensemble d'arêtes. Chaque arête comporte une courbe porteuse et limité par deux sommets.

IV.2.2.1 Les avantages de ce modèle sont :

- ce modèle est facile en description, visualisation et transformation géométrique.
- il est aussi unique (Pour une pièce mécanique, il existe un seul modèle BREP).

V.2.2.2. Les inconvénients de ce modèle sont :

- difficultés de réalisation des opérations logiques (dues au grand nombre de faces constituant l'objet initial).
- absence d'algorithmes généraux (La plupart des algorithmes associés à ce modèle traitent des volumes dont les frontières sont composées de facettes planes, elles-mêmes délimitées par des segments de droite pour faciliter les opérations logiques).
- L'inconvénient de cette méthode est sa faible puissance de représentation.

IV.2.3. Modèle hybride

Le modèle hybride permet de cumuler les avantages de la représentation BREP et CSG en limitant au mieux leurs inconvénients respectifs. Dans ce modèle, les primitives de base sont, soit celles du modèle CSG (cône, pyramide, cube,...), soit des solides définis par leurs frontières. Lors de chaque opération sur l'arbre de construction, les éléments frontières (représentant la peau de l'objet) sont construits. C'est le modèle le plus souvent utilisé actuellement.

IV.2.3.1. Avantages du modèle :

- le modèle Hybride permet de construire aussi simplement un solide qu'avec un arbre CSG et même mieux puisque la création de primitives BREP est possible.
- le modèle Hybride permet facilement d'effectuer une visualisation rapide du solide.
- le modèle Hybride permet d'effectuer aisément des opérations complexes comme le calcul du volume d'un solide, l'appartenance d'un point au solide etc.
- l'Hybride permet de disposer de la définition exacte de la surface du solide.
- le modèle Hybride offre la possibilité de placement de contraintes géométriques et de contraintes d'assemblages.

IV.2.3.2. Inconvénients du modèle.

- il est nécessaire de stocker une très grande quantité d'informations.

il est nécessaire de maintenir en permanence la cohérence entre les deux représentations de l'information

Chapitre V

Les machines-outils à commande numérique

Plan de cours

Objectifs de chapitre

A la fin de cette chapitre l'étudiant devra être capable de :

- Présenter l'étudiant historique des machines-outils
- Définir un système de production automatisé
- Présenter des machines à commandes numériques
- Comprendre les différentiels référentiels

Déroulement

Le chapitre V sera abordé durant deux séances comme suit :

- Notions sur les machines-outils à commande numérique
- Système de référence de la MOCN

Pré acquis

- Outils informatique
- Notion de gamme de fabrication et contrat de phase
- Lecture d'un dessin de définition
- Machines conventionnelles

Moyens

- Tableau
- Vidéo projecteur
- Visite des machines CNC

V. Les machines-outils à commande numérique

V.1. Introduction

La machine-outil à commande numérique et son évolution actuelle représentent aujourd'hui le moyen de production le plus important des pièces mécaniques. Grâce au développement des techniques de modélisation, cette dernière a subi des modifications, et le couple outil machine-outil s'est adapté aux exigences de productivité modernes. Une Machine-Outil à Commande Numérique (MOCN) est une machine d'usinage à cycle automatique programmable.

V.2. Notions sur les machines-outils à commande numérique (MOCN)

Le système de programmation de la CFAO génère le programme de la commande numérique CN procédé d'automatisation qui permet les déplacements des organes de la *machine-outil* à partir de la géométrie de pièce fournit par la fonction CAO .En effet, le choix de la machine, des outils, des processus et des montages d'usinage doit se fait pour servir comme données d'entrée à la FAO.

Finalement, une machine-outil à commande numérique (MOCN), c'est une machine-outil dotée d'une commande numérique. Lorsque la commande numérique est assurée par un ordinateur, on parle parfois de machine CNC (commande numérique par ordinateur).

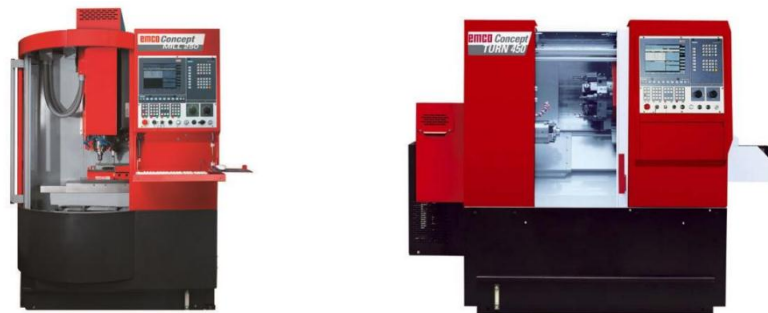


Figure.V.1 : a) Fraiseuse à commande numérique *Emco* Concept Mill 250 (à gauche),
b) Tour à commande numérique *Emco* Concept Turn 450 (à droite)

V.3. La structure d'une machine à commande numérique

Machine-outil á Commande Numérique, désigne une machine ou un centre d'usinage permettant la réalisation de travaux d'enlèvement de matière par exécution automatique d'un programme. Elle est composée généralement de deux parties principales :

- La partie opérative (une machine-outil)
- La partie commande (un directeur de commande numérique).

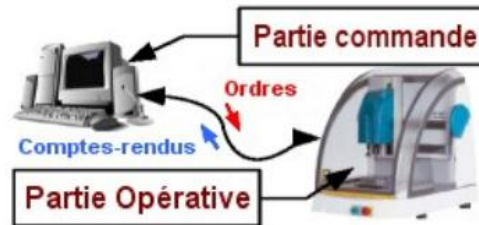


Figure.V.2 : Machine-Outil à Commande Numérique

V.3.1. Partie opérative :

Représente la machine elle-même. Elle est identique aux machines conventionnelles dans la fonction "usinage" mais elle possède des performances supérieures. Elle peut posséder plusieurs têtes d'usinage motorisées selon un ou plusieurs systèmes d'axes indépendants et elle comporte également des éléments annexes pour commander les dispositifs de serrage et de lubrification. La partie opérative d'un automatisme est le sous-ensemble qui effectue les actions physiques (déplacement, usinage, lubrification...), mesure des grandeurs physiques (accélération, vitesse...) et rend compte à la partie commande, elle est généralement composée d'actionneurs, de capteurs, d'effecteurs et d'un bâti. D'où la partie opérative reçoit les ordres de la partie commande et les exécute, voir Figure.V.3.

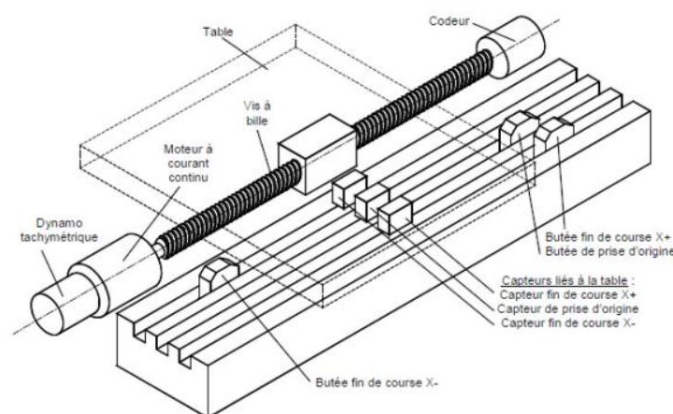


Figure.V.3 : Les éléments de la partie opérative.

V.3.2 Partie commande

La partie commande c'est le cerveau de la machine, appelé *directeur de commande numérique* (DCN). Ce dernier envoie des ordres de commande de rotation et de déplacement aux organes mobiles de la machine à partir des valeurs de consignes décrites dans le programme de la pièce à réaliser. C.-à-d. transformer les informations codées du programme en ordres aux servomécanismes de la partie opérative, afin d'obtenir les déplacements des organes mobiles, voir Figure.V.4.

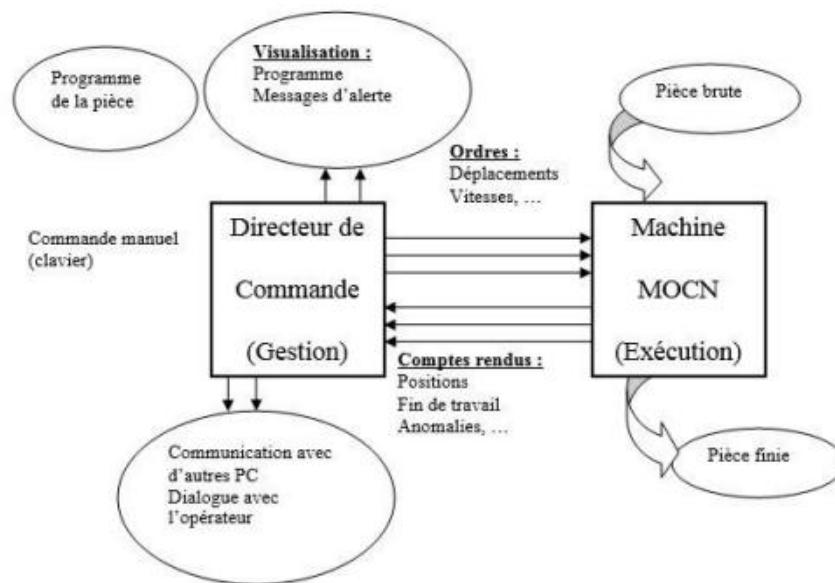


Figure.V.4 : Fonction originale d'une commande numérique.

Différente d'une machine conventionnelle, elle est constituée d'une armoire dans laquelle on trouve (Fig.V.5) :

- Le pupitre permettant de rentrer les commandes à l'aide d'un clavier.
- Le lecteur de données (ce lecteur peut être une option lors de l'achat de la machine).
- La sortie RS pour les liaisons avec les Périphériques externes.
- L'écran de visualisation de toutes les données enregistrées.
- Le calculateur.
- Les cartes électroniques (commandes d'axes, mémoire ...).

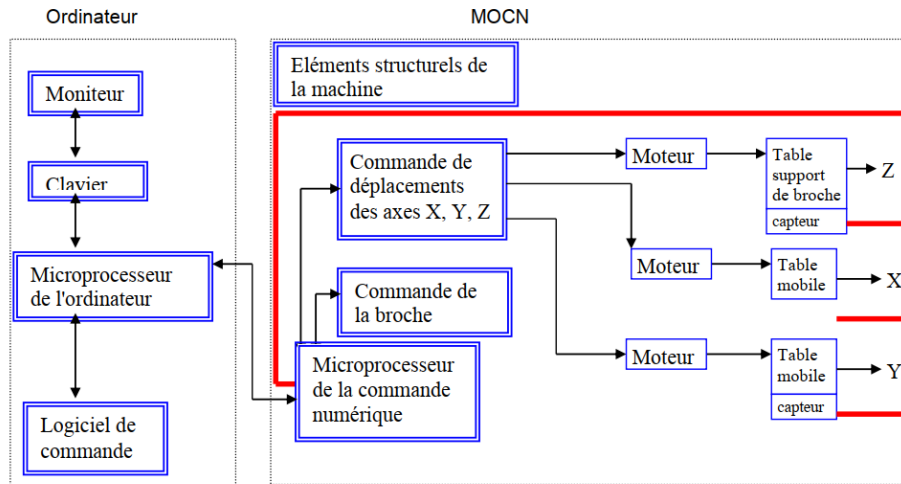


Figure.V.5 : Structure d'une machine-outil à commande numérique

V.4. Principe de fonctionnement d'une machine à commande numérique

Une MOCN est capable de commander ses propres mouvements suivant deux ou trois axes (voir plus) et de mesurer avec précision les déplacements de ses organes mobiles: **porte-pièce** et/ou **porte outils** (Figure.V.5). Ces automatismes sont gérés par un système électronique (**Directeur de Commande Numérique**) qui garde en mémoire la description des opérations à effectuer : *le programme*. Ce programme contient les instructions de mouvements de tous les éléments d'une **machine-outil** à savoir :

- L'outil (ou les outils) d'usinage équipant la machine,
- Les tables où sont fixées les pièces,
- Les magasins d'outil,
- Les dispositifs de changement de pièce,
- Les mécanismes connexes, pour le contrôle ou la sécurité, l'évacuation des copeaux...

V.5. Les avantages des machines d'usinage :

A travers le progrès technologique en termes d'électronique et d'informatique, les MO utilisées pour l'usinage des pièces mécaniques ont évolué du conventionnel au numérique pour répondre aux exigences posés par :

- la complexité des formes des pièces à usiner ;
- la précision dimensionnelle élevée et l'état de surface rugueux ;
- la cadence de fabrication ;

- la diminution du temps d'usinage pour réduire le coût de fabrication ;
- la réduction du coût de réalisation ;
- la réduction de l'intervention de l'opérateur et l'annulation de ses erreurs.

V.6. Classification des machines à commande numérique :

Le classement des machines est nécessaire car il aide au choix de machines, lors d'étude de gammes de fabrication. Traditionnellement, on classe les machines en fonction des formes de surfaces à réaliser : cylindriques/parallélépipédiques, tournage/fraisage. On classe maintenant les machines-outils par le nombre de mouvements élémentaires qu'elles peuvent mettre en œuvre lors du déplacement de l'outil par rapport à la pièce. Seuls les axes sont décomptés. La mise en œuvre simultanée de plusieurs outils entraîne l'augmentation du nombre d'axes. Les machines-outils à commande numérique (MOCN) sont classées suivant :

- Le mode de fonctionnement de la machine.
- Le nombre d'axes de la machine.
- Le mode d'usinage.
- Le mode de fonctionnement du système de mesure.

V.7. Les familles des machines à commande numérique :

On peut découper la famille des commandes numériques en quatre sous familles de machines :

- Fraisage à commande numérique (FCN).
- Tournage à commande numérique (TCN).
- Rectification à commande numérique.
- électroérosion à commande numérique.

Dans chaque famille, les méthodes de montage et de travail sont totalement différentes, mais elles se rejoignent sur le principe de programmation, la grande majorité des machines utilisant un langage ISO. On trouve plus fréquemment des MOCN :

- **Des fraiseuses à CN** (Figure 2.b): qui possèdent des vis avec noix pourvues de billes précontraintes pour éliminer le jeu. L'usage de moteurs linéaires sur certaines machines a même supprimé l'usage des vis.
- **Des tours à CN** (Figure 2.a): Sur lesquels la vitesse de rotation est recalculée dynamiquement par rapport au diamètre usiné, cela permet d'avoir des états de surface bien supérieurs à l'usinage traditionnel.

- **Des centres d'usinage à CN** : exécutant plusieurs phases d'usinage sur le même poste de travail.



Figure V.6 : MOCN : 2.a. Tour à CN, 2.b. Fraiseuse à CN

V.8. Le principe d'asservissement d'un organe mobile

Chaque axe mobile de la machine est asservi en boucle fermée en position et en vitesse (Figure 3). Ce qui permet de mesurer en permanence le déplacement de cet organe et de le comparer à sa valeur consigne, une fois l'écart devient nul le déplacement s'arrête.

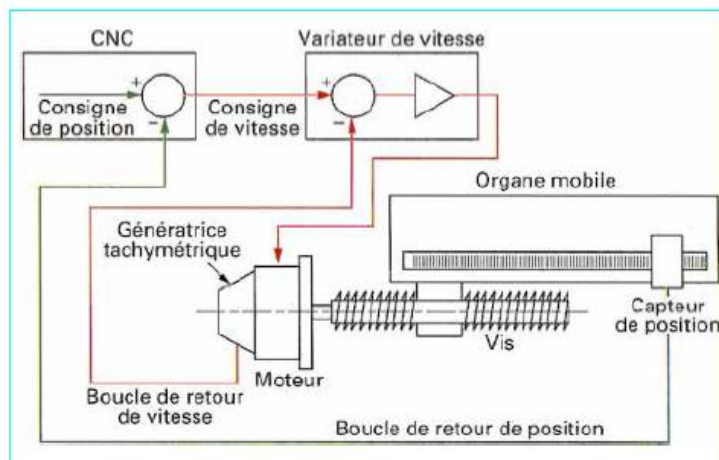


Figure V.7 : Principe d'asservissement d'un organe mobile.

V.9. Mécanisme d'entraînement de l'organe mobile :

L'organe mobile (les tables portant les pièces ou les portes outils) de la MOCN est entraîné en déplacement par le système formé de vis à billes recirculantes et de deux demi-écrous précontraints pour réduire le frottement et éliminer le jeu (Figure 4).

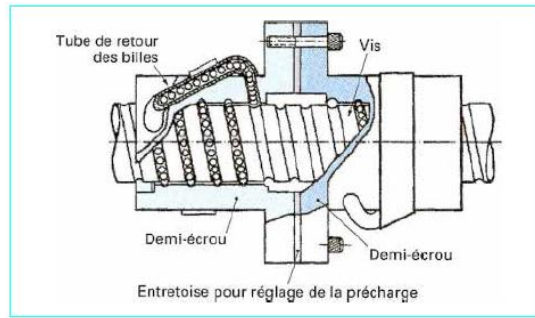


Figure V.8 : Vis et écrou à recirculation de billes.

V.10. Système de référence de la MOCN :

Les déplacements de l'outil ou du porte-pièce s'effectuent par combinaisons de translations et/ou de rotations. Chaque mouvement élémentaire (axe) est repéré par une lettre affectée du signe + ou - indiquant le sens du déplacement. Le système de référence est défini par (Figure 5) : les trois axes principaux X, Y et Z tel que :

- Z : représente l'axe de la broche : supportant la pièce en tournage ou l'outil en fraisage ;
- X : axe du long déplacement ;
- Y : axe formant un trièdre direct avec les deux axes X et Z.

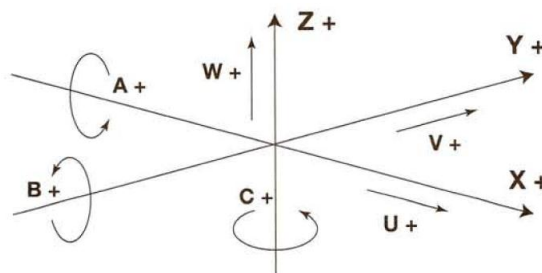


Figure V.9 : Système de référence de la MOCN

Les trois axes secondaires U, V et W translations parallèles respect. à X, Y et Z et les trois axes rotatifs A, B et C rotations autour des axes X, Y et Z est celui de la broche. En tournage (Figure 6 .a) et en fraisage (Figure 6 .b) on a les axes suivants :

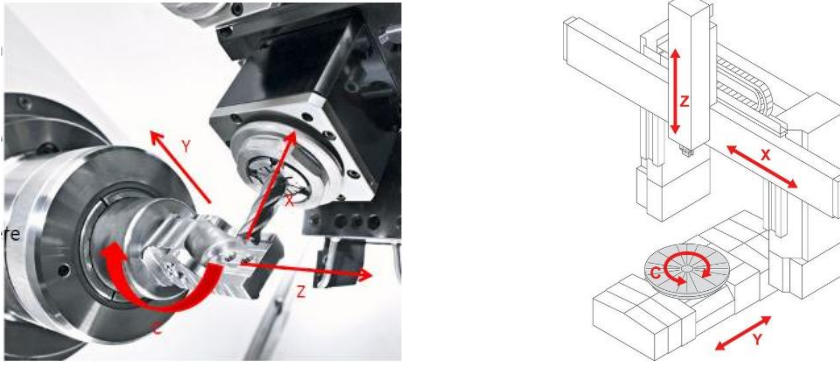


Figure.V.10 : a.Tour à CN 4 axes , b. Fraiseuse à CN 4 axes

V.10.1. Origines

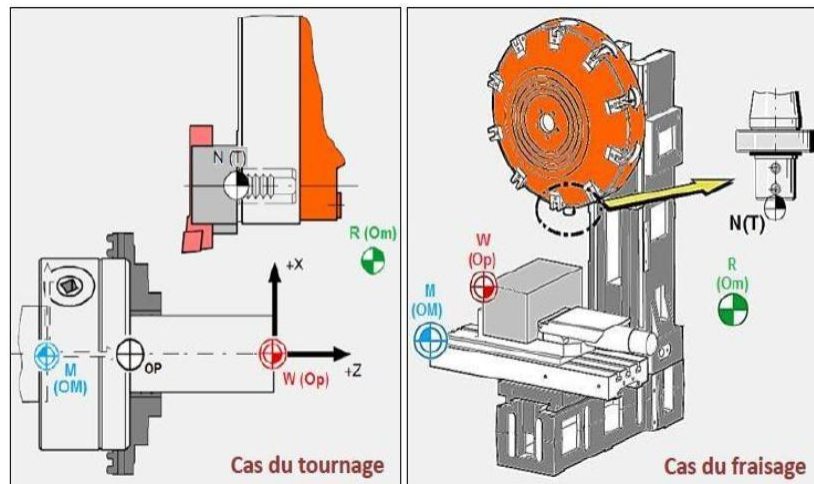
Dans l'espace de travail d'une MOCN. Sont définis différents points de de référence. Ces points sont nécessaires pour le pré réglage et la programmation de la machine. Le tableau ci-dessous résume les différentes origines des MOCN

Points utilisées	Symbole	Définition
Origine machine M (OM)		C'est la référence des déplacements de la machine. C'est un point défini (sur chaque axe) par le constructeur qui permet de définir l'origine absolue de la machine. OM et om peuvent être confondues.
Origine mesure R (Om)		C'est le point de départ de toutes les mesures dans l'espace machine
Origine Programme OP		C'est le point de départ de toutes les cotes.
Origine Pièce W (Op)		origine de la mise en position (isostatique de la pièce)

Nous présenterons les différents points ainsi que leur positionnement dans l'espace de travail d'une MOCN. Les figures suivantes, montrent les points de référence dans le volume d'usinage (la répartition des origines en tournage et en fraisage) :

Point de référence dans le volume d'usinage dans le cas fraisage.

La position de point « M », l'origine machine, étant très éloignée de la pièce à usiner, ne convient pas en tant que point de départ de la programmation. Donc, il va falloir décaler l'origine machine vers un point qui facilite la programmation. Ce point est l'origine de la pièce.



V.10.2. Types de capteurs

Les capteurs servent à mesurer la position instantanée de l'organe mobile et peuvent être :

- **Analogiques**, lorsque le déplacement demandé entraîne une variation magnétique, électrique ou autre, qui est transformée en signal de sortie (capteurs inductifs, résolveurs).
- **Numérique**, lorsque le déplacement d'une règle ou d'un disque présentant des zones alternativement sombres et claires devant un lecteur optique produit des impulsions lumineuses qui sont transformées en signal de sortie (capteurs photoélectrique, règle et disques optiques)

Chapitre VI

Programmation des MOCN

Plan de cours

Objectifs de chapitre

A la fin de cette chapitre l'étudiant devra être capable de :

- Etablir les règles de programmation ISO
- Applications les fonctions d'un programme
- Comprendre la correction du profil

Déroulement

Le chapitre VI sera abordé durant deux séances comme suit :

- Structure d'un programme
- Les fonctions d'un programme

Pré acquis

- Outils informatique
- Notion de gamme de fabrication et contrat de phase
- Lecture d'un dessin de définition

Moyens

- Tableau
- Vidéo projecteur

VI.1. Introduction

Le langage de la programmation des MOCN consiste à décrire les opérations d'usinage dans un langage codé (appelé code G) assimilable par le calculateur de la machine. Ce langage est normalisé (norme ISO 1056) où certains codes utilisés ont les mêmes fonctionnalités pour différents contrôleurs de machines-outils (NUM, FANUC, SIEMENS,...). Les autres codes peuvent avoir une interprétation différente d'un contrôleur à un autre. Le langage de programmation des MOCN possède les caractéristiques suivantes :

- La chronologie des actions,
- L'appel des outils,
- La sélection des vitesses de coupe et d'avance,
- La formulation des trajectoires,
- La définition des coordonnées de fin de trajectoire,
- Les mises en ou hors fonction d'organes de la machine.

VI.2. Structure d'un programme

La programmation s'effectue suivant le code ISO. Un programme est constitué de lignes appelées "blocs". Un bloc correspond aux instructions relatives à une séquence d'usinage. Chaque bloc est constitué d'un groupe de mots. Un mot est un ensemble de caractères composé d'une adresse suivie de chiffre constituant une information.

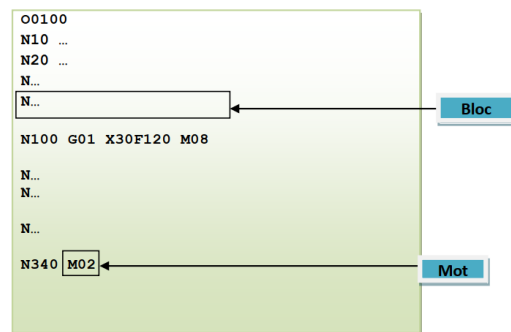


Figure.VI.1 :

Programme

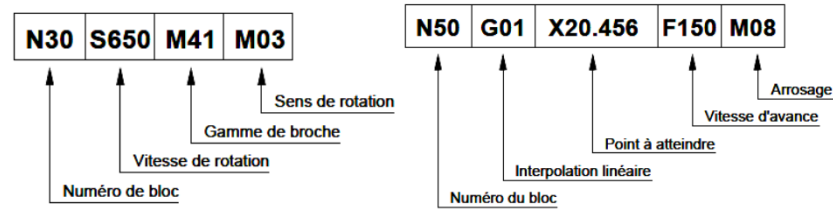
- **Bloc** : groupe de mots correspondant aux instructions relatives à une séquence d'usinage. La numérotation des blocs s'effectue de 5 en 5 ou de 10 en 10 pour permettre une insertion éventuelle de blocs.
- **Mot** : ensemble de caractères comportant une adresse suivie de chiffres constituant une information.
Exemple : X 10.850
- **Adresse** : lettre débutant un mot d'un langage machine, qui précise la fonction générale commandée.

VI.2.1. Exemples de blocs :

Ecriture d'un bloc définissant la mise en rotation de broche.

Ecriture d'un bloc

définissant une trajectoire.



VI.2.2. Principales adresses :

- N... : numéro de ligne : repérage chronologique en début de ligne.
- G... : fonctions préparatoires définissant la forme et les conditions de déplacement.
- M... : fonctions auxiliaires donnant les changements d'état de la machine.
- X..., Y..., Z... : axes principaux désignant les coordonnées des points d'arrivée.
- R... : paramètres définissant les trajectoires circulaires.
- F... : précise l'avance ou la vitesse d'avance.
- S... : précise la vitesse de coupe ou la vitesse de rotation de la broche.
- T... : symbole du numéro d'outil.

VI.3. Les différentes manières de programmer

Un programme est la transaction, dans un langage compréhensible par le directeur de commande numérique d'une machine. Des opérations d'usinage à effectuer sur une pièce. Les différentes manières de programmer sont :

- La programmation manuelle,
- La programmation assistée : soit conventionnelle par le DNC, soit avec un logiciel de FAO.

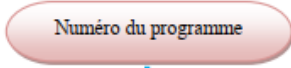
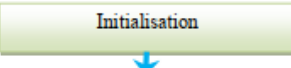
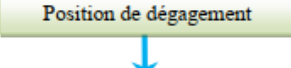
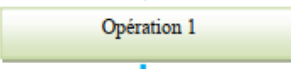
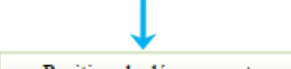
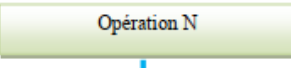

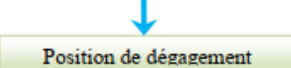
Les documents suivants sont nécessaires :

- Le dessin de définition,
- Le contrat de phase avec l'isostatisme et les paramètres de coupe,
- Le dossier de la machine.

Pour réaliser les différentes opérations nécessaires à l'usinage d'une pièce, un programme CN peut être écrit de différentes manières. Selon la nature de la pièce à usiner et sa complexité, les différentes structures de programme CN peuvent être proposées :

- Un programme principal,
- Un programme principal contenant des appels de séquences internes,
- Un programme principal et des sous programmes.

Pour l'usinage d'une pièce simple ne nécessite pas de cycles d'ébauche, un programme CN peut être structuré de la manière suivante :

Structure	Exemple
	O4723 (Tour SMI) (Ph 10)
	N28 G40 G80
	N29 G00 X100 Z100
	(Opération : centrage) N30 T0505 M06 (Foret à centrer) N40 G00 X0 Z52 N45 G95 G97 S2500 F0.05 M03 M08 N50 G01 Z40 N60 G00 Z52
	N70 G00 G00 X100 Z100 M09
 (Opération : finition profil) N240 T0303 M06 (Outil de finition) N250 G92 S4000 N260 G00 X4 Z52 F0.05 M8 N270 G01 G42 X5 Z46 N280 G96 S250 N290 G01 X15.961 Z46 N300 X24 Z31 N310 Z26 N320 X26 N330 G03 X34 Z22 I26 K22 N340 G01 Z18
	N370 G40 G00 X100 Z100 M05 M09
	N710 M02

Structure d'une opération :

Structure	Exemple
Opération précédente	(Opération : centrage)
Appel de l'outil	N30 T0505 M06 (Foret à centrer)
Approcher l'outil en rapide	N40 G00 X0 Z100
Adapter les conditions de coupe	N45 G97 G95 S2500 F0.05 M03 M08
Réaliser l'opération d'usinage	N50 G01 Z40
Dégager l'outil en rapide	N60 G00 Z100
Opération suivante	

VI.4. Les fonctions d'un programme**VI.4.1. Les fonctions préparatoires G**

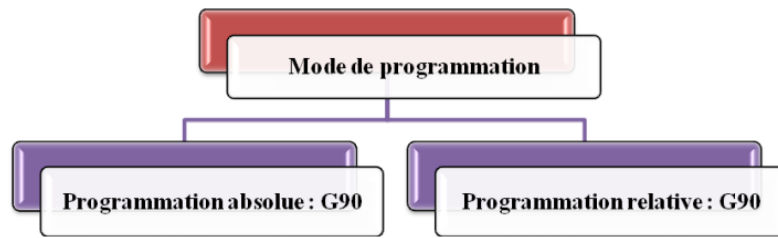
Les fonction G en générale c'est une fonction préparation a un rôle spécifique dans le programme

a) Choix du mode de programmation

Il existe deux types de commandes de déplacements de l'outil : les commandes absolues et les commandes incrémentielles (relatives).

- G90 : Programmation absolue par rapport à l'origine programme. La valeur programmée sur un axe est repérée par rapport à l'origine programme (OP).
- G91 : Programmation relative par rapport au point de départ du bloc. La valeur programmée sur un axe est repérée par rapport à la dernière position programmée.

Pour assurer l'usinage d'une pièce sur **machine-outil commandée numériquement**, le programmeur peut recevoir le dessin de produit fini coté suivant deux modes.

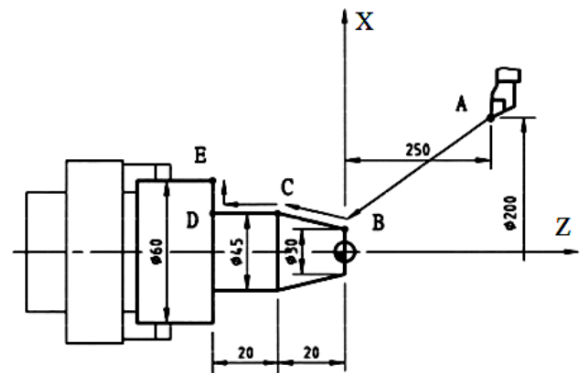


➤ **Exemples de tournage :**

Veillez indiquer l'itinéraire de mouvement de l'outil ci-dessous par des méthodes de positionnement absolu et positionnement relatif.

Programme de positionnement absolu

G54 X200	
Z250	Réglage du système de coordonnées.
G00 X30 Z0	Déplacez l'outil du point A au point B.
G01 X45 Z-20 F0.2	Coupe d'outil du point B au point C.
Z-40	Coupe d'outil du point C au point D.
X60	Coupe d'outil du point D au point E.



Programme de positionnement incrémentiel ou relatif

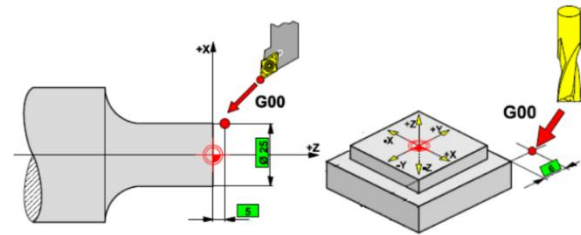
G54 X200	
Z250	Réglage du système de coordonnées.
G00 U-170	Coupe d'outil du point A au point B.
W-250	Coupe d'outil du point B au point C.
G01 U15 W-20 F0.2	Coupe d'outil du point C au point D.
W-20 U15	Coupe d'outil du point D au point E.
U15	

b) Programmation des déplacements (interpolation)

b.1) Interpolation linéaire à vitesse rapide G00

G00 : Interpolation linéaire à vitesse rapide.

Le point programmé est atteint en effectuant une trajectoire linéaire à vitesse rapide. La trajectoire est la résultante de tous les déplacements d'axes programmés dans le bloc.

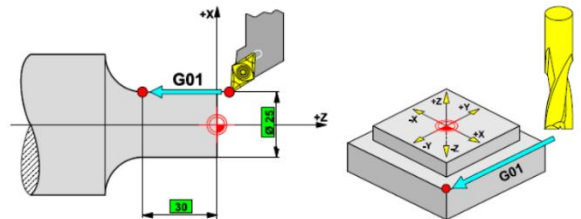


Syntaxe : N..G00 X.. Y.. Z..

X.. Y.. Z.. Point à atteindre

b.2) Interpolation linéaire à vitesse d'avance programmée G01

Le point programmé est atteint en effectuant une trajectoire linéaire à vitesse d'avance programmée. La trajectoire est la résultante de tous les déplacements des axes programmés dans le bloc.



Syntaxe : N..G01 X.. Y.. Z.. [F..]

G01 : Interpolation linéaire à vitesse d'avance programmée.

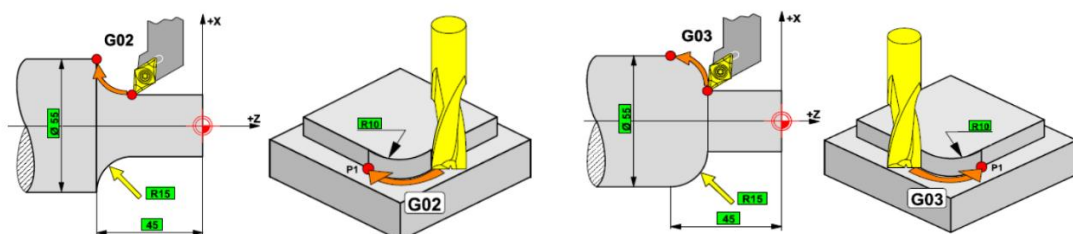
X.. Y.. Z.. Point à atteindre

F.. : Vitesse d'avance.

b.3) Interpolation circulaire

G02 : Interpolation circulaire sens anti-trigonométrique à vitesse d'avance programmée.

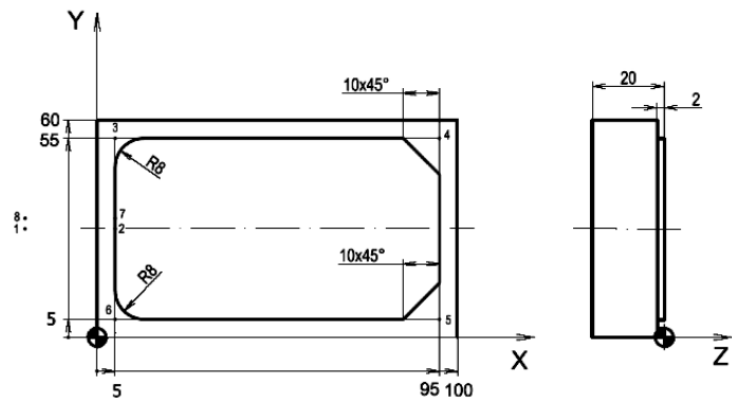
G03 : Interpolation circulaire sens trigonométrique à vitesse d'avance programmée.



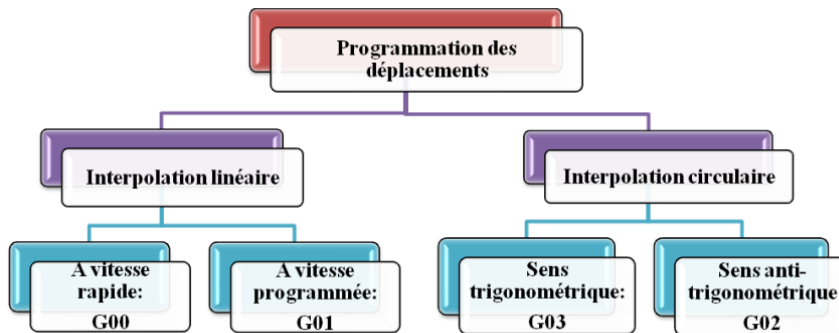
b.4) Fonction pour le chanfrein et pour l'angle de l'arc

Cette fonction est utilisée pour simplifier le codage du programme pour une section à angle droit de la pièce de travail ayant un chanfrein de 45° ou des angles d'arcs tangentiels dans le plan X-Y. La commande pour le chanfrein et pour l'angle de l'arc nécessite la commande de déplacement G01 avec une valeur de déplacement supérieure à l'angle du chanfrein ou à la valeur du rayon en amont.

	(Point
N20 G00 X-22	1)
Y30 Z1	(Point
N25 G01 Z-2	2)
N30 G41	(Point
N35 G01 X5	3)
N40 G01 Y55 R8	(Point
N45 G01 X95 C10	4)
N50 G01 Y5 C10	(Point
N55 G01 X5 R8	5)
N60 G01 Y32	(Point
N65 G01 X-22	6)
G40	(Point
N70 G00 Z50	7)
.....	(Point
	8)



➤ **Organigramme d'interpolation :**



➤ **Exemple :**

N20 T0303

G96 S200 M03

G00 X0 Z3 M08

G42 G01 Z0 F0.2

G03 X20 Z-10 R10

G01 Z-50

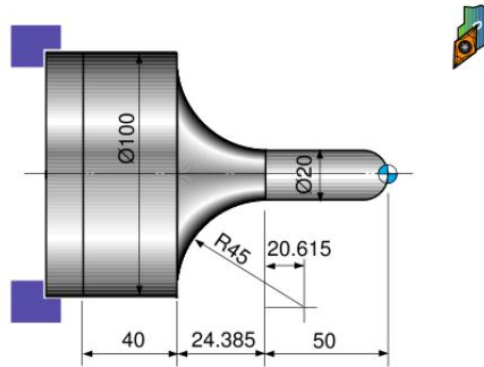
G02 X100.0 Z-74.385 R45

G01 Z-125

G40 U2 W1

G00 X200 Z200 M09

M30

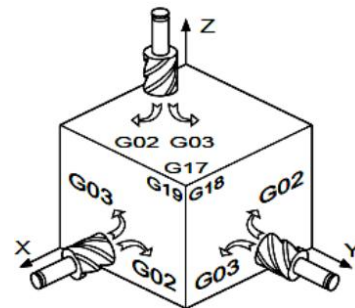


b.5) Choix du plan d'interpolation :

Deux axes linéaires pilotés dépendant du choix du plan d'interpolation (Fraisage uniquement) :

- axes X (ou U) et Y (ou V) en G17,
- axes Z (ou W) et X (ou U) en G18,
- axes Y (ou V) et Z (ou W) en G19.

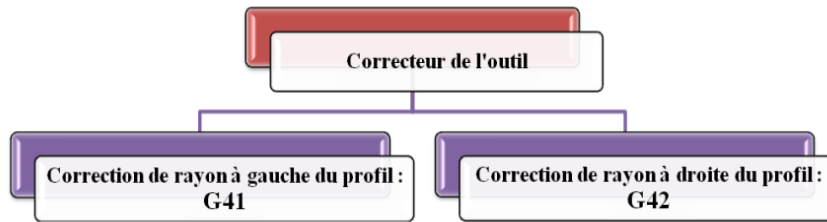
Syntaxe (Plan N... [G17] G02/G03 X...Y...I...J... /
XY) : R... [F...]
 G17 Choix du plan XY.
 X...Y... Point à atteindre :



c) Positionnement de l'outil

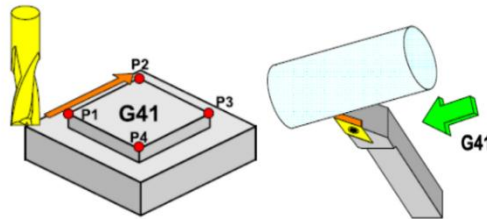
par rapport à la pièce

Comme l'outil a un rayon, le centre de la trajectoire de cet outil est décalé de la valeur de son rayon par rapport au profil de la pièce. Si les rayons des outils sont mémorisés dans la CNC, l'outil utilisé peut être décalé par rapport la pièce de la valeur de son rayon. Cette fonction est appelée fonction compensation de rayon de l'outil.



c.1) Correction de rayon à gauche du profil à usiner G41

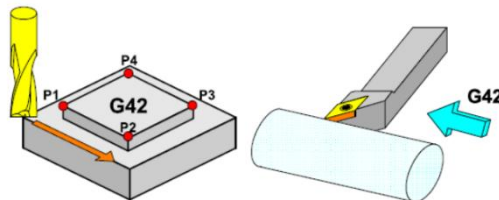
Les trajectoires outil programmées sont corrigées (décalées à gauche) d'une valeur égale au rayon d'outil (R) déclaré par le correcteur D.



c.2) Correction de rayon à droite du profil à usiner G42.

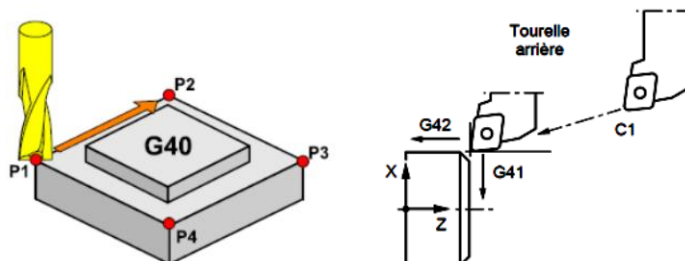
Les trajectoires outil programmées sont corrigées (décalées à droite) d'une valeur égale au rayon d'outil (R).

Syntaxe: N...G41/G42 X...Z...



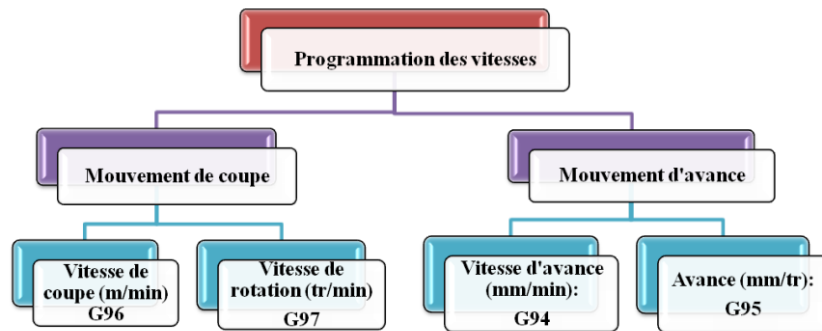
c.3) Annulation de correction de rayon G40.

Pilotage du centre de l'outil : les trajectoires programmées sont appliquées au centre de l'outil.



➤ **Exemple** : Correction de rayon suivant la position de la tourelle :

d) Programmation des vitesses :



Syntaxe : N.. G97 S... [M03/M04]

N.. G96 S... [M03/M04]

N.. G94 F200 (Vitesse d'avance en mm/min).

N.. G95 F0.3 (Vitesse d'avance en mm/tr)

G97 : Vitesse de broche exprimée en tr/mn.

G96 : Vitesse de coupe constante en m/mn.

S : Argument obligatoire lié à la fonction la vitesse programmée.

G94 : Vitesse d'avance en mm/min

G95 : Vitesse d'avance en mm/tr

e) Retour aux origines (G28)

La commande G28 retourne rapidement l'outil de sa position actuelle vers le point d'origine machine, en passant par un point intermédiaire. X, Y et Z sont les valeurs des coordonnées du point intermédiaire qui servent de mécanisme de sécurité pour empêcher les outils de percuter n'importe quelle pièce, accessoire et appareillage sur son retour à sa position d'origine.

Syntaxe : G90 G28 X... Y... Z... ; (mode de coordonnées absolues)

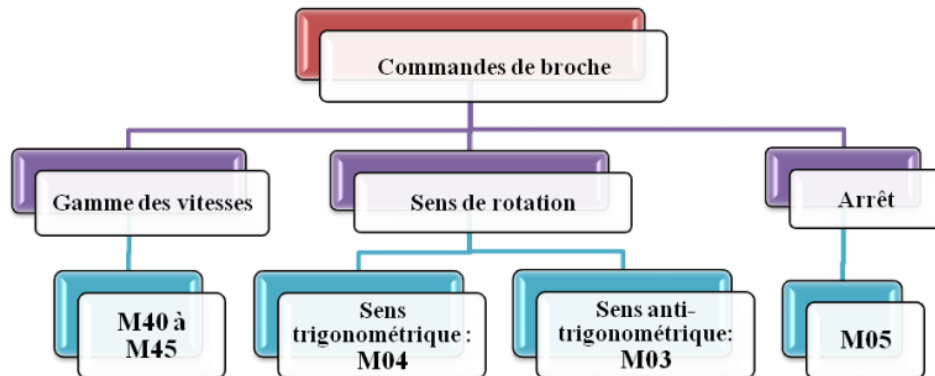
G91 G28 X... Y... Z... ; (mode de coordonnées relatives)

Tableau VI.1 : Les fonctions préparatoires, instructions G

COMMANDES	Description
G00	Avance rapide en coordonnées cartésiennes
G01	Interpolation linéaire en coordonnées cartésiennes.
G02	Interpolation circulaire dans le sens horaire en coordonnées cartésiennes.
G03	Interpolation circulaire dans le sens antihoraire (trigonométrique) en coordonnées.
G04	Temporisation.
G09	Arrêt précis.
G10	Avance rapide en coordonnées polaires.
G11	Interpolation linéaire en coordonnées polaires
G12	Interpolation circulaire dans le sens horaire en coordonnées polaires.
G13	Interpolation circulaire dans le sens antihoraire en coordonnées polaires.
G16	Sélection du plan.
G17	Sélection du plan.
G18	Sélection du plan.
G40	Suppression compensation du rayon d'outil (plaquette en tournage).
G 41	Compensation du rayon d'outil à gauche
G 42	Compensation du rayon d'outil à droite.
G 48	Retrait du contour de la manière dont s'est opérée l'approche.
G 50	Suppression modification de l'échelle.
G 51	Sélection modification de l'échelle.
G 53	Suppression du décalage d'origine.
G 54 à G 57	Décalage d'origine.
G 58 et G59	Décalage d'origines additives, programmables.
G 60	Mode d'arrêt précis.
G 62,63	Suppression du mode d'arrêt précis.
G71	Indication de cotes en millimètre.
G90	Programmation en cotes absolues.
G 91	Programmation en cotes relatives.
G 92	Limitation de vitesse pour G96.
G 94	Avance par minute, vitesse d'avance A en mm/mn.
G 95	Avance par tour (f) en mm/tr.
G 96	Vitesse de coupe (m/mn).
G 97	Vitesse constante

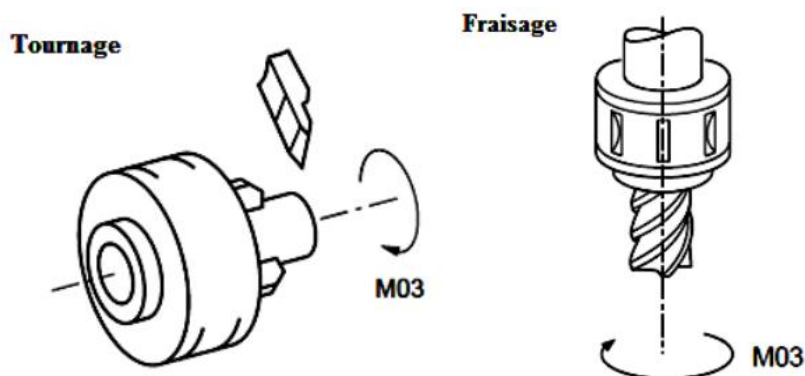
VI.4.2. Les fonctions auxiliaires M

a) Commandes de broche



b) Commande du sens de rotation M03- M04

M03 : Rotation de broche dans le sens anti-trigonométrique. La commande permet la mise en rotation de la broche à la vitesse programmée.



- **M04** : Rotation de broche dans le sens trigonométrique. La commande permet la mise en rotation de la broche à la vitesse programmée.
- **M05** : Arrêt de broche. La commande arrête la rotation de la broche.
- **M06** : Appel ou changement d'outil. La fonction permet l'appel d'un outil et le positionnement de celui-ci à son poste d'usinage.

Lorsque des perçages, des taraudages, des alésages, des fraisages et autres opérations d'usinage doivent être effectuées, il est nécessaire de sélectionner un outil adéquat. Lorsqu'un numéro est attribué à chaque outil et que le numéro est spécifié dans le programme, l'outil correspondant est sélectionné

Syntaxe : N.. T.. M06 T.

La fonction «T» affectée d'un numéro sélectionne l'outil.

Tableau des codes M usuels

Type	Signification	Fonction
M00	Arrêt programmé	Après
M01	Arrêt optionnel	Après
M02	Fin de programme (identique à M30)	Après
M03	Rotation broche sens horaire	Avant
M04	Rotation broche sens trigonométrique (ou anti-horaire)	Avant
M05	Arrêt rotation broche	Après
M06	Changement outil	Après
M07	Marche second arrosage	Avant
M08	Marche arrosage principal	Avant
M09	Arrêt arrosage	Après
M19	Orientation broche	Après
M30	Fin de programme (identique à M02)	Après

VI.5. Exemple

O 0450	Numéro de programme
N10 G90 G92 S2500	Programmation absolue, limitation de la vitesse de rotation à 2 500t/min.
N20G96S180G95 F0.15	Régler la vitesse de découpe à 180m/min et de l'avance à 0,15mm/tr
M03	mise de la broche en rotation dans le sens horaire.
N30 T0101	Sélection de l'outil 01.
N40 G00 X29 Z2	Interpolation rapide, déplacer l'outil vers le point de départ du
N50 G01 Z-14,5	tournage (point d'approche).
N60 G02 X40 Z-20 R5,5	Interpolation linéaire à vitesse d'avance programmée jusqu'au point
N70 G01 Z-32	de coordonnées (X29, Z-14,5).
N80 X44	Interpolation circulaire dans le sens horaire jusqu'au point de
N90 G03 X60 Z-40 R8	coordonnées (X40, Z-20).
N100 G01 Z-52	Interpolation linéaire à vitesse d'avance programmée jusqu'au point
N110 X67	de coordonnées (X40, Z-32).
N120 G28 U0 W0 M05	Interpolation circulaire dans le sens horaire jusqu'au point de
N130 M30	Coordonnées (X60, Z-40).
	Retour automatique au point de référence et arrêt broche.
	Fin programme.

Reference

- [1] W.Dahmen, C.Micchelli, H.-P.Seidel. Blossoming begets B- spline bases built better by B-patches. Mathematics of Computation, 59(199):97-115, 1992.
- [2] P.Fong and H.-P.Seidel. An implementation of triangular B-spline surfaces over arbitrary triangulations. CAGD, 3(10):267-275, 1993.
- [3] H.Qin, D.Terzopoulos. Triangular NURBS and their Dynamic Generalizations. CAGD, 14:325-347, 1997.
- [4] M.Franssen, R.C.Veltkamp, W.Wesselink. Efficient Evaluation of Triangular B-spline Surfaces. CAGD, 17:863-877, 2000.
- [5] A.Goshtasby. Geometric Modeling Using Rational Gaussian Curves and Surfaces. CAD, 27(5). 363-375, 1995.
- [6] C.Grimm and J.Hughes. Modeling surfaces of arbitrary topology using manifolds. Proc. SIGGRAPH '95, pp:359-368.
- [7] J.C.Navau and N.P.Garcia. Modeling surfaces from meshes of arbitrary topology. CAGD, 17(7):643-671, 2000.
- [8] J.C.Navau and N.P.Garcia. Modelling surfaces from planar irregular meshes. CAGD, 17(1):1-15, 2000.
- [9] S.Haker, S.Angenent and A. Tannenbaum, etc.. Conformal surface parametrization for texture mapping. IEEE TVCG, 6(2):1-9, 2000.
- [10] M.Desbrun, M.Meyer, P. Alliez. Intrinsic Parameterizations of Surface Meshes. Eurographics'2002, pp: 209-218.
- [11] B.Lvy, S.Petitjean and N. Ray, etc.. Least Squares Conformal Maps for Automatic Texture Atlas Generation. Proc. SIGGRAPH '2002, pp. 362-371.