ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DE LA
MAITRISE EN GÉNIE DE LA PRODUCTION AUTOMATISÉE
M.Ing

PAR
EL ASLI, Neila

APPROCHE HYBRIDE BASÉE SUR LES MACHINES À VECTEURS DE SUPPORT ET LES ALGORITHMES GÉNÉTIQUES POUR L'ESTIMATION DES COÛTS DE FABRICATION

MONTRÉAL, LE 14 MAI 2008

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

M. Victor Songmene, directeur de mémoire
Département de génie mécanique à l'École de technologie supérieure

M. Thien-My Dao, codirecteur de mémoire
Département de génie mécanique à l'École de technologie supérieure

M. Ali Gharbi, président du jury
Département de génie de la production automatisée à l'École de technologie supérieure

M. Barthélemy H. Ateme-Nguema, examinateur externe
Département des Sciences de la Gestion, Université du Québec en Abitibi-Témiscamingue

REMERCIEMENTS

Pour commencer, j'aimerais exprimer ma profonde gratitude à mon professeur *M.Thien-My DAO* pour le soutien moral et financier qu'il m'a offert durant ces années de maîtrise. Je ne saurais pas vous remercier, cher professeur, pour toute la confiance, la compréhension et la patience que vous m'avez accordées pour bien commencer et finir ce travail.

Mes vifs remerciements s'adressent aussi à mon directeur *M.Victor Songmene* pour sa disponibilité, son aide, ses conseils et ses encouragements qui m'ont aidé à mener à terme la rédaction de ce mémoire.

Je remercie également les membres du jury, messieurs *Ali Gharbi* et *Barthelemy H.Ateme N* d'avoir accepté d'évaluer ce travail.

Un grand merci à mon cher mari *Zied*, qui m'a soutenue, encouragée et n'a épargné aucun effort pour m'aider à persévérer pour bien finir ma maîtrise.

Je n'oublie pas tous ceux et celles, amis et collègues, qui ont contribué de loin ou de près à m'offrir soutien, confiance et encouragement le long de ces années, j'en avais vraiment besoin pour arriver à ce jour. Merci beaucoup *Barthelemy* pour toute l'écoute que tu m'as offerte quand tu étais parmi nous à l'ETS, j'en garderai de très bons souvenirs.
Merci à mon amie, *Felicia* pour son soutien lors des durs moments, j'en suis vraiment très reconnaissante. Merci encore *Felicia* et bonne continuation !

Pour terminer, j'aimerais dédier ce mémoire à mes parents, mes frères, ma sœur qui sont si loin géographiquement, mais si près de mon cœur. Vous attendiez tant que je finisse et voici que le moment est enfin arrivé.

Merci à tous !

APPROCHE HYBRIDE BASÉE SUR LES MACHINES À VECTEURS DE SUPPORT ET LES ALGORITHMES GÉNÉTIQUES POUR L'ESTIMATION DES COÛTS DE FABRICATION EN PHASE DE CONCEPTION

EL ASLI, Neila

RÉSUMÉ

L'estimation du coût des produits est une étape cruciale pour les entreprises manufacturières d'aujourd'hui; surtout, en phase de conception, lorsque les conditions et les moyens de fabrication ne sont pas encore complètement connus. Pour ces raisons, il est important de fournir au concepteur les outils nécessaires en vue d'une estimation de coûts efficace, précise et adaptée aux connaissances relatives aux produits à ce stade.

Dans ce travail, nous proposons une nouvelle méthode hybride d'estimation de coûts de produits basée sur les machines à vecteurs de support (communément appelées SVM) et les Algorithmes Génétiques (AG). Cet outil de l'intelligence artificielle fondé sur la théorie de l'apprentissage statistique a été choisi pour sa grande capacité d'apprentissage et de généralisation.

Dans notre approche proposée, les SVM sont utilisées pour faire une approximation de la relation entre les conditions de conception et les paramètres du produit dans le cas d'estimation de coût. Les *AG* ont servi pour sélectionner les hyper-paramètres des *SVM*. En plus, et pour identifier les paramètres ou variables les plus influençant sur le coût final, nous avons fait appel aux *« fuzzy curves »*, basées sur la théorie de la logique floue. De cette manière, nous pourrions *jouer* sur ces paramètres afin d'optimiser le coût final du produit.

En résumé, notre approche hybride est capable d'effectuer l'estimation des coûts des produits, ainsi qu'une sélection des variables les plus pertinentes influençant sur ce dernier. Pour démontrer son potentiel et sa robustesse, une application dans le domaine de fabrication mécanique est présentée.

A HYBRID APPROACH BASED ON SVM AND GENETIC ALGORITHMS (GA) FOR ESTIMATION OF EARLY COMPONENT DESIGN STAGES MANUFACTURING COST

EL ASLI, Neila

ABSTRACT

Estimating manufacturing cost is a crucial activity for today's global competitive manufacturing firms, especially at the early design stages when the product is not completely defined and, the conditions and means of manufacturing are not fully known. It's important to provide to the designer an economic and practical tool to make cost estimations at this stage.

In this work, an hybrid approach based on the Support Vector machines (SVM) and genetic algorithms (GA) is developed. The SVM are learning machines that can perform binary classification (pattern recognition) and real valued function approximation (regression estimation) tasks. This tool of the artificial intelligence founded on the theory of the statistical learning was selected for its great capacity of training and generalization. The proposed approach is founded on assumption that the conditions of design and the parameters of the product contribute to its final cost. Since it's difficult to find an explicit relation between such conditions and parameters and the final cost of the product, the SVM are used to approximate this relation, through their training from examples.

To optimize the configuration of our approach, namely the structure which gives us the most precise result in terms of error of prediction, we used the genetic algorithms for the choice of hyper-parameters of the SVM. Additionally, the system provides, in the case of product, the most important parameters or cost drivers. We used for this goal «*the fuzzy-curves*» which are based on the theory of fuzzy logic to arise the significant variables in a non-linear modeling. Thus, the user will be able to know the effect of variation of the design parameters on his cost and decide on which parameters; it will be able to investigate to optimize the final production cost.

The proposed approach for the production cost estimation is compared with others techniques as the neural networks approach and its applications show that the proposed hybrid approach provides a better performance. The proposed methodology carries out the cost estimation of the product, as well as a selection of the most relevant cost drivers influencing on the final manufacturing cost. This will be an economic and practical tool for the engineer designers to estimate the manufacturing cost of a product at its design stage. Also, the application to the design of some mechanical components has confirmed the effectiveness of the proposed technique.

TABLE DES MATIÈRES

Page

LISTE DES TABLEAUX

Page

LISTE DES FIGURES

INTRODUCTION

Il a été convenu que 70 % du coût d'un produit est déterminé au stade de conception. Et donc, c'est pendant cette phase de conception du produit que les possibilités de réduction du coût total sont les plus importantes. Par conséquent, il est avantageux de pouvoir estimer le coût du produit tôt durant son cycle de développement. Toutefois, durant l'étape de conception, le produit n'est jamais complètement défini et les conditions et moyens de fabrication ne sont pas encore définitivement connus.

Pour ces raisons, il est nécessaire de fournir au concepteur les outils nécessaires afin d'en arriver à une estimation de coûts efficace et adaptée aux connaissances relatives au produit en question. En effet, l'estimation de coût est un processus de chiffrage qui permet à son utilisateur de prévoir le coût final d'un futur projet ou produit, sans que tous les paramètres et/ou conditions soient connus lorsque cette estimation est mise en place.

Dans ce travail, la méthode utilisée pour l'estimation de coûts des produits est basée sur les machines à vecteurs de support (communément appelées SVM). Cet outil de l'intelligence artificielle, dont les performances sont bien connues pour les applications de classification, de reconnaissance, etc., a été choisi pour sa grande capacité d'apprentissage.

L'approche proposée est basée fondamentalement sur l'hypothèse que les conditions de conception et les paramètres du produit contribuent à son coût final. Plusieurs techniques d'estimation de coût ont été développées pour différents types de produits en se basant sur leurs divers paramètres. Les paramètres explicites dont le coût du matériel, peuvent facilement être obtenus, tandis que ceux qui sont implicites comme le coût de la complexité du design doivent être obtenus à travers une analyse de l'historique des coûts du produit. Par conséquent, la question clé de l'estimation de coût traitée dans ce mémoire est de savoir comment utiliser l'historique des coûts pour comprendre ces paramètres implicites.

Généralement, il est difficile de trouver une relation explicite entre les paramètres définissant le produit et son coût final, les SVM, étant un outil intelligent et fondé sur la théorie de l'apprentissage statistique, sont utilisées en vue d'approximer cette relation.

Nous commençons ce travail par présenter les différentes approches d'estimation de coûts couramment utilisées actuellement, en précisant leur domaine d'application par rapport au cycle de vie des produits. Nous allons, par la suite, introduire les SVM, notamment les SVM pour la régression, utilisées dans ce travail pour l'estimation des coûts. Nous allons également présenter la théorie de l'apprentissage artificiel et ses fondements mathématiques. Pour les SVM, tout comme les réseaux de neurones, plusieurs paramètres sont considérés dans la structure même de l'outil; (par exemple pour les réseaux de neurones (*ANN)* , il y a le nombre de couches cachées, le nombre de neurones dans chaque couche, le type de la fonction d'activation, etc.). Pour atteindre la meilleure configuration possible de notre outil, à savoir la structure qui nous donne le résultat le plus précis, nous allons utiliser les Algorithmes Génétiques (AG) pour le choix de ces hyperparamètres.

Par ailleurs, nous allons faire ressortir, dans le cas d'un produit, les paramètres les plus importants, qui influencent le plus sur le coût. Nous avons utilisé pour ce but *« les fuzzy curves»* qui se basent sur la théorie de la logique floue pour ressortir les variables significatives dans une modélisation non linéaire. Nous allons aussi faire un survol des méthodes existantes pour l'identification de variables pertinentes, et nous mettrons l'accent sur cette technique à la fois simple et robuste qui nous servira pour identifier les paramètres les plus importants influant sur le coût final d'un produit. Notre choix s'est arrêté sur cette méthode, puisque comme les SVM, l'historique des produits déjà fabriqués est requis.

Le premier chapitre de ce travail, présente les différentes méthodes d'estimation de coût développées dans le passé, ainsi qu'une classification de ces dernières tel que trouvé dans les divers travaux de littérature consultés.

Le deuxième chapitre regroupe les bases théoriques des différentes techniques utilisées le long de ce travail, à savoir, les machines à vecteurs de support (SVM), les Algorithmes Génétiques (AG) et les *fuzzy-curves*.

Le troisième chapitre synthétise l'approche proposée dans ce mémoire ainsi que la méthodologie suivie pour la réalisation de l'estimation de coût par SVR-Génétique. Ce chapitre est la partie la plus importante de ce travail et résume principalement ma contribution dans ce domaine de recherche.

Le quatrième et dernier chapitre, et dans le but de démontrer la performance ainsi que le potentiel d'application de l'approche proposée, présente deux applications de produits mécaniques, ainsi qu'une partie d'analyse et discussions par rapport aux résultats trouvés.

CHAPITRE 1

LES TECHNIQUES D'ESTIMATION DE COÛT EN FABRICATION MÉCANIQUE

De nos jours, les entreprises vivent dans un monde de compétition où elles cherchent à maximiser leur profit et à acquérir de plus en plus de marchés prometteurs. Face à cette concurrence technico-économique, la maitrise des coûts d'un projet ou d'un produit devient une obligation pour le succès de ces entreprises. La première étape pour la maitrise de ces coûts, est une estimation, précise, rapide et robuste. L'estimation de coût est devenue alors un domaine de recherche et de développement de plus en plus répandu dans diverses disciplines. Plusieurs techniques et méthodes en sont ressorties.

Dans ce chapitre, nous recensons, classifions et analysons les différentes approches existantes pour cette estimation de coûts, connues à ce jour, et de les classer selon le contexte de leur application.

1.1 Classification des différentes méthodes d'estimation de coût.

Dans la littérature, plusieurs travaux ont porté sur les méthodes d'estimation des coûts dans divers domaines, parmi lesquels, nous citons, par exemple, le génie-civil (Hegazy et Ayed, 1998), le textile (Camargo et al., 2003) et le génie-logiciel (Idri et al., 2002). Nous nous sommes intéressés dans ce travail aux méthodes utilisées en fabrication mécanique.

Une revue de ces différents travaux nous a permis de classer les différentes méthodes existantes en 4 grandes catégories : les méthodes analytiques, les méthodes paramétriques, les méthodes analogiques et les systèmes experts qui, en plus de l'estimation du coût, font la sélection des matériaux, des procédés, etc. et sont souvent des outils d'aide à la décision pour les concepteurs.

La méthode analogique se subdivise en sous-catégories. Par exemple, la technique de raisonnement à partir de cas (en anglais, *Case-based reasoning* (CBR)), élaborée par Duverlie (duverlie, 1996).

La méthode analytique regroupe les techniques génératives, l'analyse basée sur les activités [en angalis *Activity-Based Costing* (ABC)], qu'il ne faut pas confondre avec la méthode ABC ou loi de Pareto (méthode du 80-20), et la technique de modélisation se basant sur les formes des produits *Feauture-Based* estimating. Et finalement, la méthode paramétrique est employée dans les modèles de régression et dans la nouvelle approche des réseaux de neurones qui, fait partie du domaine de l'intelligence artificielle.

La figure 1.1 résume selon notre propre catégorisation, les différentes techniques d'estimation des coûts et leur classification.

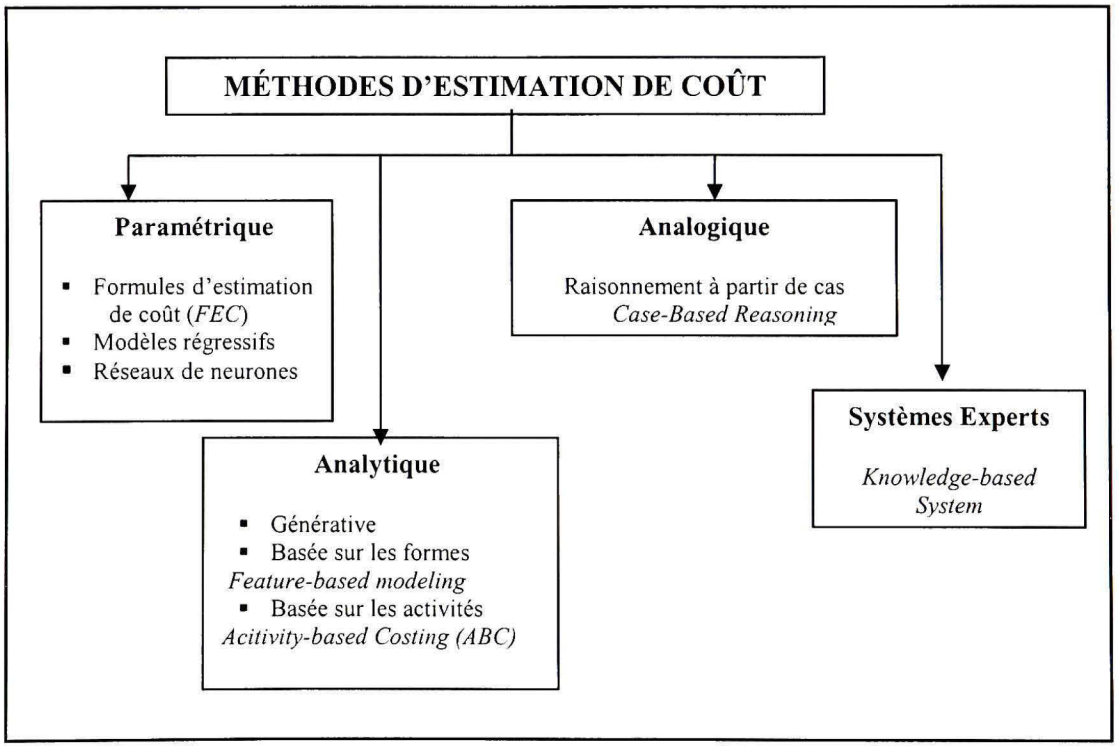

Figure 1.1 ***Classification des différentes méthodes d'estimation de coût.***

Nous trouvons des auteurs qui ont classifié autrement les méthodes existantes, Evans et al. (2006), a identifié 10 méthodes d'estimation de coût, il n'a pas reparti ses méthodes en sous groupes, comme nous l'avons fait dans ce travail. Ben Arieh et al. (1999), propose une classification en 5 méthodes et ajoute la méthode des réseaux de neurones comme nouvelle approche.

Duverlie (1999), présente 4 méthodes pour l'estimation de coût, Watson et al. (2006) identifie 2 grandes approches : celle qui est basée sur l'expérience passée, et celle des méthodes génératives qui peuvent être divisées en techniques explicites (*rule-based*), méthodes de ratio, méthodes paramétriques et méthodes analytiques. Il ajoute à ces deux grandes approches, une autre faisant appel à l'intelligence artificielle comme les réseaux de neurones et la logique floue.

Niazi (2006), (figure 1.2) apporte une toute autre catégorisation des méthodes d'estimation de coût puisqu'il fait une distinction entre les méthodes qualitatives et quantitatives. Parmi les méthodes qualitatives, il classe les méthodes intuitives et analogiques. Dans la catégorie des méthodes quantitatives, il place les méthodes paramétriques et analytiques. Parmi les méthodes intuitives basées sur les expériences passées, il classe les systèmes experts. Pour l'auteur, l'approche par réseaux de neurones, est une méthode analogique, qui se base sur l'expérience antérieure pour estimer le coût de nouveaux produits.

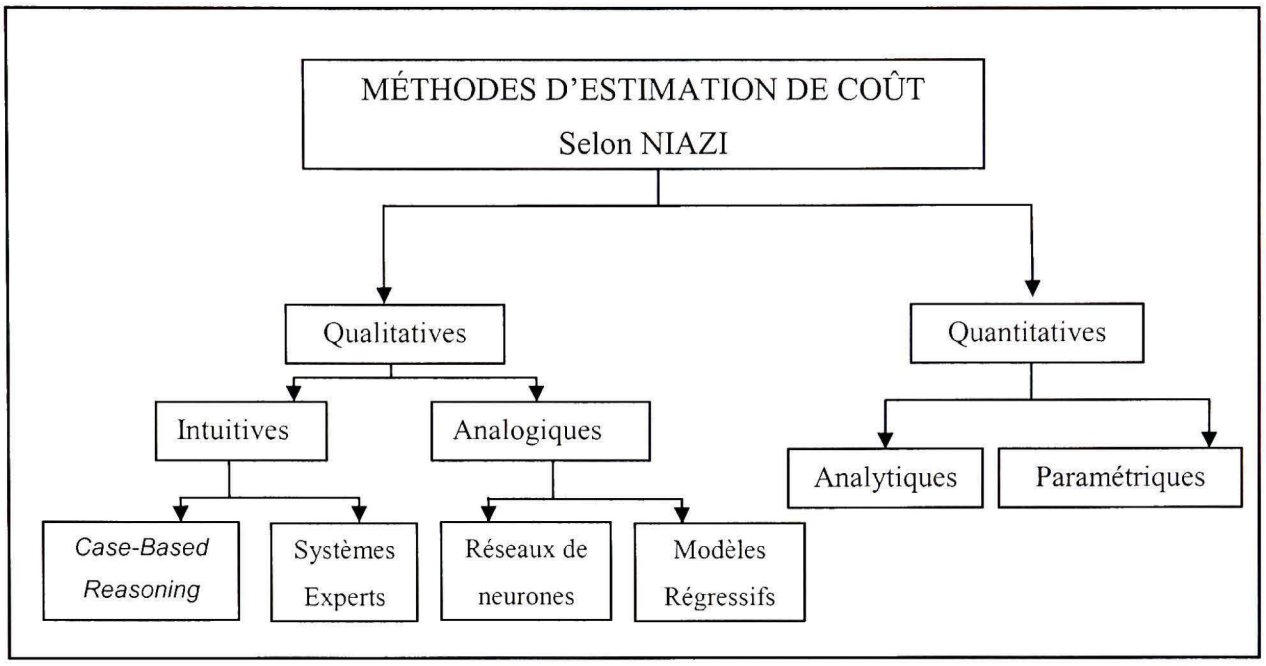

Figure 1.2 ***Classification des méthodes d'estimation de coût selon NIAZI.***

La majorité des autres travaux considèrent seulement les 3 principales modélisations pour l'estimation de coûts, à savoir le modèle paramétrique, analogique et analytique. Layer et al. (2002) distingue aussi entre les méthodes qualitatives et les méthodes quantitatives. Dans la catégorie quantitative, l'auteur classe 3 modèles : modèles statistiques, modèles analogiques et modèles analytique-génératifs.

Après avoir fait le tour des travaux de la littérature pour la classification des différentes méthodes, nous allons présenter dans ce qui suit chacune d'elles en citant les travaux qui ont été élaborés.

Pourquoi cette classification ?

Avant de rentrer dans les détails de chacune des méthodes d'estimation de coût, il sera judicieux de présenter au lecteur les raisons de cette classification. En effet, chaque estimation de coût requiert, au préalable, une étude du contexte dans lequel elle sera

réalisée. Ce contexte comprend le domaine, le type du produit, sa phase de production et surtout les données disponibles pour cette estimation.
Nous allons montrer que dépendamment de ce contexte, une (*ou certaines*) méthode(*s*) serait (*aient*) appropriée(*s*) pour chaque cas rencontré. D'un autre coté, cette classification nous a permis de découvrir le vaste domaine *d'estimation de coût* et de choisir, par conséquent, laquelle des méthodes nous allons adopter pour ce travail.

1.1.1 La méthode analytique

La méthode analytique est la plus ancienne, la plus classique et, donc, la plus répandue, des méthodes d'estimation de coûts. Le terme, *analytique* réfère au sens de «procéder par voie d'analyse». Les modèles analytiques reposent ainsi sur une bonne connaissance des caractéristiques d'un système. Ils nécessitent des informations très détaillées sur l'élément à chiffrer de manière simplifiée. La méthode analytique consiste à décomposer le processus de réalisation d'un produit en un ensemble de tâches élémentaires puis à évaluer le coût de chacune d'elles. Le coût du système est donc la somme des coûts élémentaires.

Le fait que la méthode analytique nécessite beaucoup d'informations très détaillées, tant sur le produit que sur le procédé de fabrication, laisse supposer qu'elle est souvent très longue et difficile à utiliser. On l'utilise, dans la plupart des cas, pour calculer le coût réel à posteriori et constituer ainsi une base de données utilisable par d'autres méthodes pouvant les incorporer à priori pour une estimation de coût telle que les méthodes paramétriques.

Dans la pratique, ce moyen d'estimation est utilisé principalement durant la phase de production en série car il nécessite des informations détaillées sur le produit et sur les procédés de fabrication (nomenclature du produit, gammes opératoires, etc.) qui ne sont pas toujours disponibles lors de la conception.

Asiedu et al. (1998) donne une autre appellation à cette modélisation. Ils lui donnent le nom de modèles détaillés, communément appelés, *bottom-up estimating*. En général, cette méthode d'estimation de coût s'applique aux procédés. Divers travaux ont appliqué cette modélisation pour des cas particuliers; dont l'usinage, le moulage à injection, le pressage, pour ne citer que ceux là.

L'estimation de coût selon cette méthode, est souvent une estimation des temps requis pour la fabrication. À la suite de l'obtention des temps, on les multiplie par les différents taux horaires pour l'obtention des coûts. Tout en suivant l'approche analytique, nous trouvons des auteurs qui ont appelé leur méthode d'estimation «méthodes génératives» (*Generative Costing*) ; d'autres les ont appelé «méthodes basées sur les activités» (*Activity-Based Costing*) ou d'autres ont préféré l'appellation «analyse basée sur les formes de produits» (*feature-Based cost estimation*).

Bien que les appellations ne soient pas les mêmes, l'approche globale d'estimation demeure la même, à savoir l'approche analytique. Dans la littérature, nous trouvons par exemple, Weustink et al. (2000) qui ont développé une approche générique pour l'estimation de coût de conception de produits. Leur approche est basée sur le fait que les coûts de fabrication de produits dépendent nécessairement des opérations requises pour le produire. Nous retrouvons, bien en évidence, ce même raisonnement chez tous les auteurs qui appellent leur approche : *approche analytique*. Pour illuster leur méthode, ils l''appliquent au cas d'assemblage de produits.

Ben arieh et al. (2002) présentent la méthodologie *Activity-Based Costing* (ABC) pour évaluer le coût de conception des pièces usinées. La méthode présentée est basée sur une analyse détaillée des activités des phases de conception et de développement du produit. Jung (2002) utilise la méthode qui se base sur les formes du produit et l'applique au procédé d'usinage. L'aspect analytique, dans cette méthode, provient de l'analyse des différentes activités qui doivent se suivre afin de réaliser des formes spécifiques du produit.

Ainsi, une connaissance des différentes étapes du procédé et des différents outils nécessaires est requise.Yang et Lin (1997) ont développé un outil assistant les concepteurs dans l'estimation des coûts de fabrication des pièces usinées, en s'inspirant de l'approche *Feature-Based Cost*. Leur outil se compose d'un module de dessin assisté par ordinateur (DAO), d'une base de données de pièces et procédés et d'un module d'analyse, basé essentiellement sur l'approche *Feature-based cost*.

Plusieurs autres travaux ont utilisé la méthode analytique; citons l'exemple de Schreve et al. (1999) qui l'ont appliquée au cas du soudage à pointe et Perrey (2003) qui l'autilisée dans le contexte de fonderie de sable, etc.

1.1.2 La méthode analogique

Cette méthode repose sur une comparaison entre le produit actuel et des produits similaires antérieurs dont les coûts sont connus. La similarité est recherchée surtout du point de vue fonctionnel. Dans ce cas, on parle de jugement d'experts et d'exploitation de leurs expériences acquises durant les projets précédents.

L'efficacité de cette méthode dépend en grande partie de la capacité de l'expert à identifier les différences et les similarités entre les produits actuels et antérieurs. Le principal avantage de la méthode analogique est sa rapidité et son faible coût de mise en œuvre. Par contre, elle implique que les comparaisons entre le projet courant et ceux passés soient pertinentes, ce qui suppose que le projet ne soit pas fondamentalement différent des réalisations passées, aussi bien dans sa conception technique que dans la conception des processus de production.

La méthode est principalement employée avec les technologies de groupe qui permettent une solution typique pour chaque scénario proposé de conception. Elle fut considérée comme la meilleure alternative dans le contexte des changements technologiques rapides.

Selon l'analyse analogique, une approche de raisonnement à partir de cas, (*Case-Based reasoning*) a été développée et appliquée par Duverlie (1996). Selon l'auteur :

> « *Le raisonnement à partir de cas, est une méthode utilisant les solutions d'expériences passées (les cas) pour résoudre un problème. Ce mode de raisonnement met en jeu les opérations de base suivantes : la reconnaissance du problème, la rémunération d'expériences similaires et de leur solutions, le choix et l'adaptation d'une des solutions (cas source) au nouveau problème (cas cible), l'évaluation de la nouvelle solution et l'Apprentissage du problème résolu.* » (Duverlie, 1999, p.12).

Duverlie et Castelain (1999), comparent cette méthode à la méthode paramétrique dans le cas d'une application aux pistons.

1.1.3 La méthode paramétrique

Les méthodes paramétriques sont largement utilisées durant la phase de conception lorsque le produit et le processus de sa fabrication ne sont pas complètement établis. La technique cherche à établir des relations entre des caractéristiques techniques et physiques des produits et leurs coûts en appliquant des modélisations mathématiques ou statistiques.

Les méthodes paramétriques d'estimation de coût impliquent la récupération de données pertinentes, nommées descripteurs ou paramètres, qui vont être utilisés par la suite dans la modélisation mathématique. Le choix de ces descripteurs ou paramètres est capital dans la qualité de l'estimation paramétrique. En conséquence, les experts qui effectuent ce choix auront une influence sur la précision du modèle. Pour ce faire, des observations passées sont utiles pour élaborer ces modèles, en effet une base de données de projets antérieurs similaires est nécessaire.

Ces méthodes sont rapides d'exécution et faciles d'emploi. De plus, elles sont mises en œuvre avec des informations limitées sur le produit, d'où leur utilités en début de phase de conception. En outre, puisqu'elles se basent sur des observations et reposent sur une rigueur

scientifique, les méthodes paramétriques sont d'excellents outils pour prédire des coûts. Nombreux sont les travaux qui ont été élaborés avec cette méthode et plusieurs auteurs se sont intéressés à cette technique. Asiedu et Gu (1998), qui ont fait une revue de l'état de l'art des méthodes d'estimation de coût à travers le cycle de vie des produits, référent cette technique au *top-down estimating*, par analogie au *buttom-up estimating* faisant référence à la méthode analytique.

Selon la modélisation paramétrique, nous retrouvons plusieurs types de techniques dans la littérature, dont la technique de formules d'estimation de coût (FEC) ou CEF pour *Cost Estimation Formula* en anglais et les techniques régressives. La modélisation par réseaux de neurones fait partie de la modélisation paramétrique et, depuis moins d'une dizaine d'années, elle fait preuve d'un grand succès et suscite un grand intérêt parmi les chercheurs dans le domaine de l'estimation des coûts. Nous y reviendrons un peu plus tard dans ce chapitre.

Parmi les auteurs qui se sont intéressés à la modélisation paramétrique, citons Farineau et al. (2001) qui mettent tout l'accent sur la méthodologie de création des formules d'estimation de coût. Ils décrivent rigoureusement cette technique et expliquent ses étapes. Roy et al. (2001) développent également une méthodologie d'estimation de coût de conception et, plus précisément, les *FEC* intégrant les paramètres quantitatifs et qualitatifs. Nous retrouvons aussi Pathak (1992) qui développe un module de sélection de paramètres de machines et d'estimation de coûts d'usinage. Il utilise la technique de régression et applique sa méthodologie à un exemple de fraisage.

Apgar et Daschbach (1987) présentent la méthodologie de la modélisation paramétrique pour l'estimation des coûts en phase de conception. Ils répondent aux différentes questions suivantes : qui utilise cette technique et comment ? À quel point les estimations sont précises ? Par où commencer ?

Ils présentent également une étude comparative entre l'estimation paramétrique et l'estimation industrielle utilisée à l'époque.

1.1.4 Combinaisons de méthodes

Nous trouvons des travaux qui utilisent une combinaison de plusieurs approches pour l'estimation du coût. Nous retrouvons par exemple Chougule et Ravi. (2005), qui ont élaboré un modèle de coût basé sur les méthodes analytique et paramétrique. Ils l'ont appliqué en particulier au procédé de moulage. Nous retrouvons aussi Bouaziz et al. (2006), qui combine les méthodes analytique et analogique pour l'estimation de coût de fabrication de matrice de pressage.

1.1.5 Comparaison entre méthodes analytiques, analogiques et paramétriques

Comme nous l'avons déjà mentionné, la majorité des auteurs retiennent les trois méthodes analytiques, analogiques et paramétriques comme principales approches d'estimation de coûts. Toutes les autres nouvelles approches seront alors dérivées de ces méthodes.

Dans le travail de Duverlie (1996), une étude comparative analyse ces 3 méthodes selon 3 critères :

- Rapidité d'implantation dans l'entreprise;
- Rapidité d'utilisation;
- Précision des résultats obtenus.

Les résultats de cette étude sont donnés dans le tableau 1.1 de la page suivante.

Tableau 1.1

Comparaison entre les 3 méthodes : analytiques, paramétrique et analogique

(Tiré de Duverlie, 1999)

Méthode	Implantation	Mise en œuvre	Précision
Analogique	-	+	+
Paramétrique	-	+	+
Analytique	+	–	+

Légende : - : Mauvais
+ : Bon

Source : Ce tableau a été tiré et traduit à partir du rapport de Duverlie et al., *Estimation des coûts en production mécanique,* p.3.

Selon le même auteur (Duverlie et al., 1996), la figure 1.3 résume le domaine d'application des trois principales méthodes d'estimation de coût pendant le cycle de vie des produits. En effet, certaines techniques sont préférables à d'autres selon le contexte. D'après l'auteur, la méthodologie la plus appropriée, en phase de conception de produits nouveaux, est l'approche paramétrique. Plusieurs autres auteurs partagent son avis.

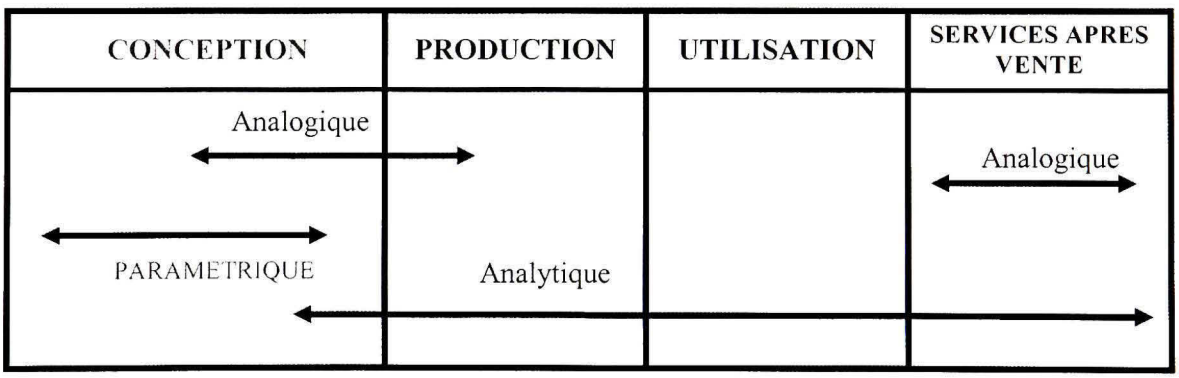

Figure 1.3 ***Phases d'application des méthodes analytique, paramétrique et analogiques.***

(Tiré de Duverlie et al., 1999 et traduit de l'anglais)

Source : Ce tableau a été tiré de l'article de Duverlie et al., *Cost Estimation During Design Step : Parametric method versus Case Based Reasoning Method*, p.896.

1.1.6 Systèmes experts

Le domaine d'estimation de coût ne cesse de gagner de plus en plus d'intérêt chez les chercheurs qui développent des outils devenant indispensables pour les concepteurs et les industriels. Voulant incessamment améliorer la qualité de ces outils, leur flexibilité et leur champs d'application, plusieurs chercheurs ont développé des systèmes experts qui servent d'outils d'aide à la décision aux concepteurs afin estimer les coûts, sélectionner les procédés et optimiser la production.

Dans la majorité des cas, ces systèmes experts se basent sur une combinaison des différentes méthodes d'estimation de coûts existantes. En général, ces systèmes experts sont des systèmes développés selon l'approche DFM (Design For Manufacture) de *Boothroyd*, qui permet une évaluation économique des choix de conception (allant du matériau à utiliser jusqu'au procédé de fabrication en passant par les caractéristiques géométriques du produit).

Plusieurs travaux ont été développés dans ce contexte. Citons par exemple celui de Chan et Lewis (2000), qui ont mis en place un système *DFM-C* se basant sur une base de connaissances «*knowledge base*» et sur des règles de production «*production rules*» appropriées. Rehman et Guenov. (1998), présentent une nouvelle méthodologie pour modéliser les coûts de fabrication dès la phase de conception. Leur approche intègre l'utilisation d'une base de connaissances et d'un raisonnement par analogie. Chin et Wong (1995) proposent un système expert pour l'estimation des coûts pour le moulage à injection (ESIMCOST : *Expert System for Injection Mold Cost Estimation*). Leur prototype a été validé en comparant les résultats obtenus à ceux des experts du domaine. Les résultats de la comparaison étaient satisfaisants.

Mcllhenny (1993) a développé un système expert d'aide à la décision, utile pour les concepteurs pour le choix des matériaux et l'estimation des coûts de production et de

moulage dans le cas des pièces moulées par injection. Tomovic (2002) présente une des méthodes les plus précises développées par *Air Force* pour l'estimation de coût et des délais de fabrication pour la *fonte des métaux* (*metal casting)*. Watson et al. (2006) ont développé un système expert hybride se basant sur les méthodes paramétriques, analogiques et celle *des ratios* pour l'estimation de coût des pièces usinées en aéronautique. Patwardhan et Ramani (2004) développent un outil d'aide à la décision (ECAS E*ngineering Cost Advisoring System*) utilisé dans le domaine du *Sand-casting*. Shehab et Abdallah (2002) ont développé un système «intelligent» pour la modélisation des coûts de deux procédés, à savoir le moulage à injection et l'usinage. Leur méthodologie se base sur l'intégration des relations mutuelles entre les facteurs de coût, les activités de production et la géométrie du produit.

1.1.7 Les réseaux de neurones

Depuis une dizaine d'années, le domaine d'estimation de coût a connu une nouvelle approche se basant sur l'utilisation des réseaux de neurones ou *ANN* (*artificial neural network*). Plusieurs auteurs se sont orientés vers l'exploration de cette nouvelle méthode d'estimation et ont témoigné de son efficacité dans ce domaine. Non seulement elle est facile à implanter, rapide à exécuter, mais aussi plus précise dans ses résultats. Bien des auteurs l'ont classée parmi les méthodes paramétriques, hypothèse que nous allons retenir dans ce travail.

Nombreux sont les travaux qui ont été élaborés avec cette nouvelle approche; citons par exemple, Cavalieri et al. (2003) qui l'a appliquée au cas d'un disque de frein et l'a comparée au modèle régressif. La nouvelle approche a donné de meilleures estimations et des résultats plus précis.

Restant dans la même philosophie en la comparant aux modèles régressifs, Shtub et Versano (1998) l'ont appliquée au pliage des tubes «*steel-pipe bending*», et encore une fois, les

réseaux de neurones ont été plus performants, plus rigoureux et plus précis comparativement aux autres méthodes paramétriques telle que la régression. Les auteurs Zhang et Fuh (1997) ont très bien expliqué le principe des réseaux et leur application pour l'estimation de coûts. Ils ont pris le cas particulier des produits d'emballage comme application à leur méthodologie. Voici, illustrées par la figure 1.4, selon les auteurs, les deux étapes de l'approche d'estimation de coût par ANN.

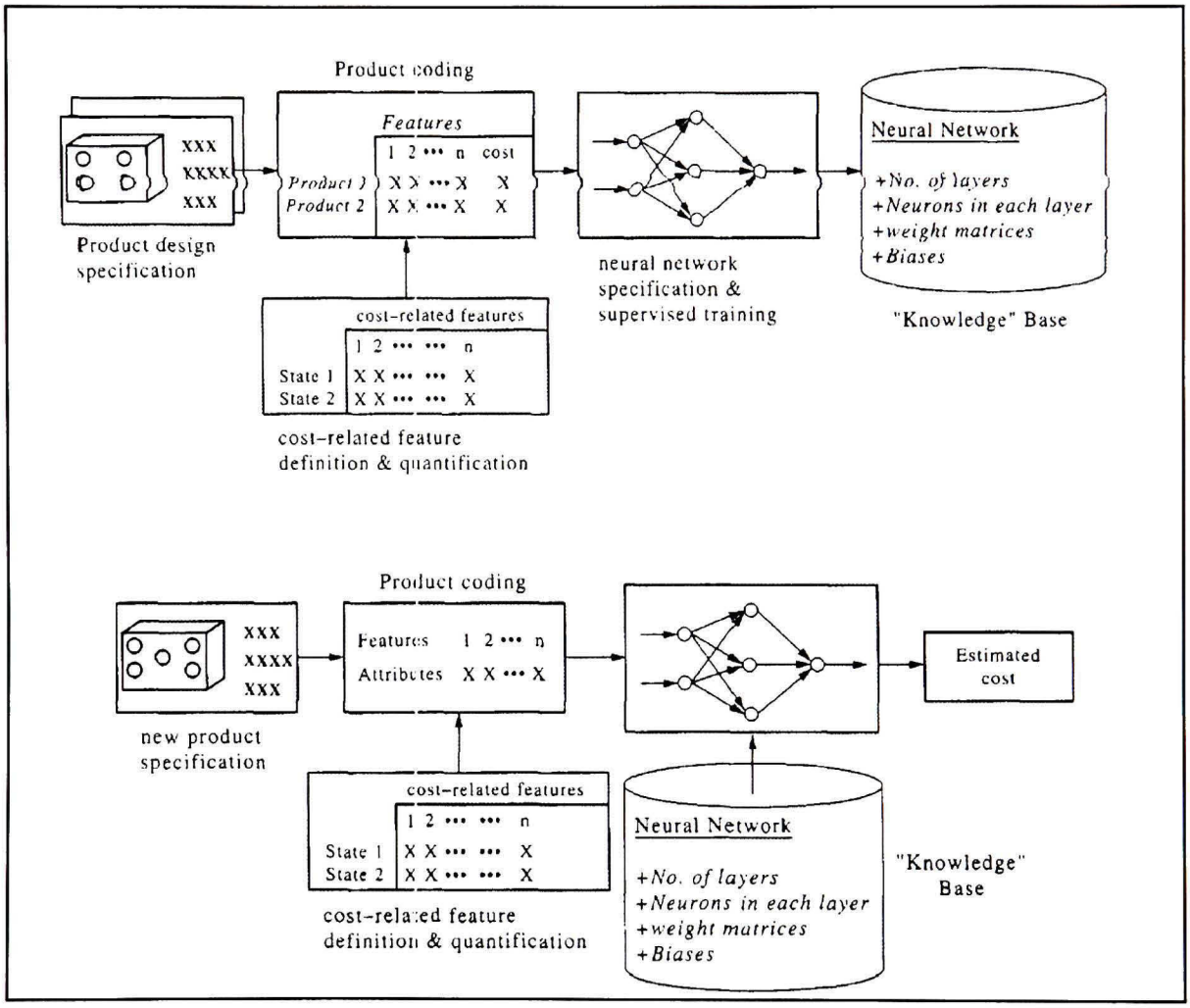

Figure 1.4 *Étapes de la méthode d'estimation de coût par réseaux de neurones.* (Tiré de Zhang et Fuh, 1997)

Source : Cette figure a été tirée de l'article de Journal de Zhang et Fuh, *A neural Network approach for Early Cost Estimation of Packaging Products*, p.437.

Plusieurs autres travaux ont été développés dans cette perspective, citons les exemples de Mckim (1993), Chen et Chen (2002), Bode (1998), Stockton et Wang (2004), De la Graza et Rouhana (1995), etc.

La difficulté de la technique résidait dans la configuration du réseau. En effet, il faudrait trouver les meilleures combinaisons de nombre de couches cachées, de nombre de neurones dans chaque couche, du choix des fonctions d'activation, etc. La solution, dans certains cas, fut la technique par essais-erreurs (Wang et Stockton, 2001). D'autres auteurs ont appliqué les algorithmes génétiques pour l'optimisation du réseau (Kim et Han, 2002; Kim.G.H et al., 2004 ; Liu et al., 2005 et Seo, 2006). Stockton et Wang (2004) ont utilisé la méthodologie *Taghuchi* afin de choisir la meilleure configuration possible du réseau.

1.2 Récapitulatifs et conclusions

Le tableau 1.2, présente un récapitulatif de notre revue de la littérature en ce qui a trait aux méthodes d'estimation de coût. Nous avons classé les travaux suivant les 4 grandes catégories de méthodes que nous avons déjà établies précédemment.

Tableau 1.2

Tableau récapitulatif de la revue de littérature concernant les différentes méthodes d'estimation de coûts

Méthode		Auteurs et Applications
Analytique	*Acitivity-Based Costing*	Ben arieh et al. (2002) : pieces usinées
	Generative	Weustink et al. (2000) : exemple en métal en feuilles (*sheet metal*).
	Based-Features	Jung et al. (2002) Yang et Lin (1997)
Paramétrique	Modèles Régressifs & *CER*	Smith et Mason (1996) Cavalieri et al. (2004) : disques de freins. Farinaeau et al. (2001) : application aux pistons. Roy et al. (2001) Camargo et al. (2003) : textile. Dean (1995).
	Réseaux de neurones	Liu et al.() fabrication de matériaux Chen et Chen (2002) : strip-steel coiler Stockton et Wang (2004) : Validation par les équations du procédé de tournage Zhang et Fuh(1997) : emballage Cavalieri et al. (2003) : industrie automobile, disques de freins Shtub et Zimerman (1993) : Assemblage Seo et Ahn (2006) : Coût de maintenance de produits Shtub et Versano (1999) : steel pipe bending
Analogique *Case-Based Reasoning*		Duverlie (1996).

Systèmes Experts	Shehab et Abdallah(2002) : Moulage par injection et usinage. Chin et Wong (1995) : Mcllhenny (1993) : Moulage par injection Tomovic (2002): *Metal casting.* Chan et Lewis (2000) : Rehman et Guenov. (1998) : Watson et al. (2006) : Patwardhan et Ramani (2004)

La figure 1.5, élaborée par Bode (2000), résume les contextes d'application des 3 plus grandes méthodes d'estimation de coûts que nous avons vues plus haut, utilisant un digramme à trois dimensions. Selon lui, 3 paramètres sont nécessaires pour définir le contexte de l'application des méthodes d'estimation ; la taille de la base de données, le nombre d'attributs ou «*cost-drivers*» et le niveau de certitude par rapport à l'influence de ces derniers sur le coût.

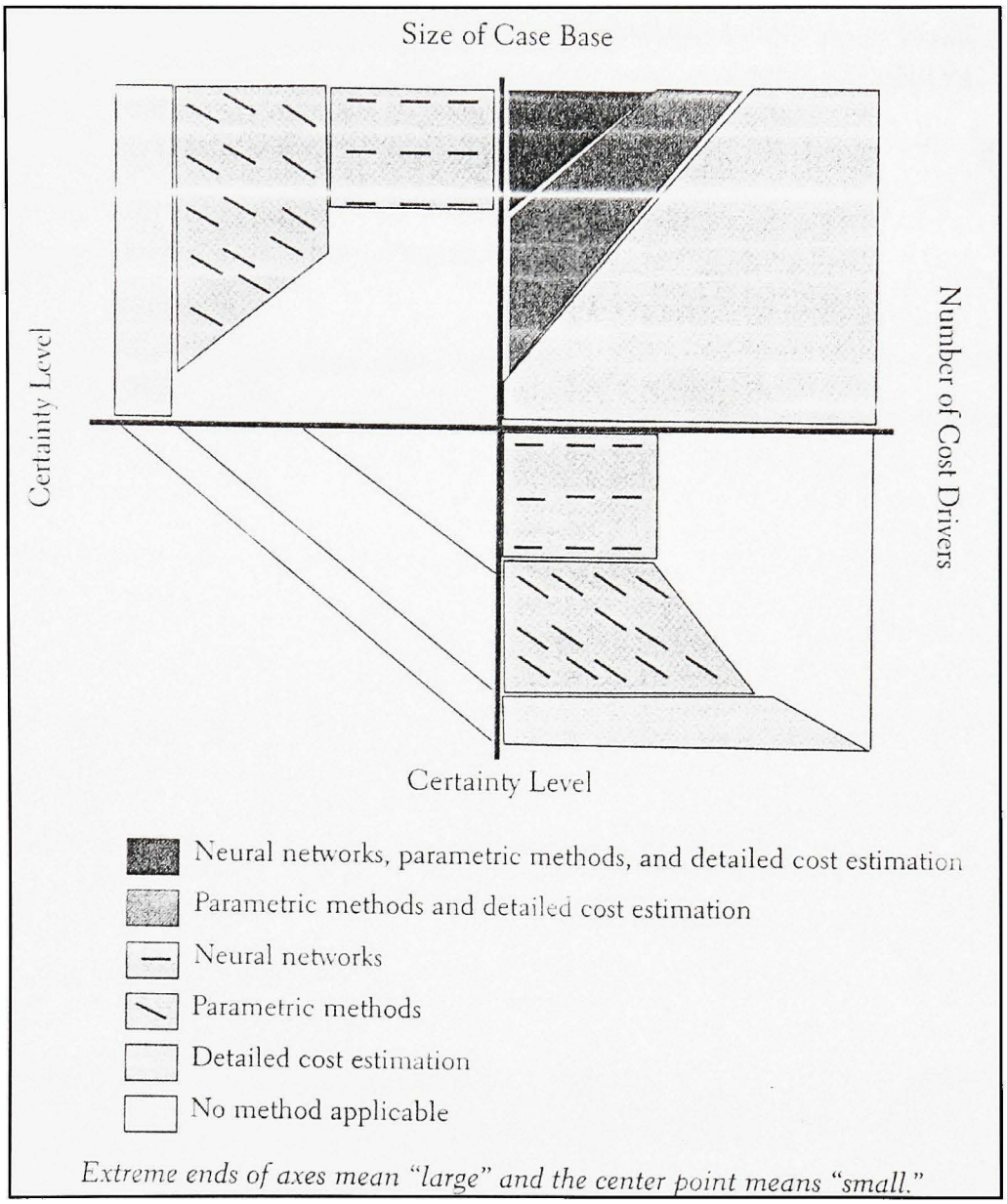

Figure 1.5 ***Contextes d'application pour les différentes méthodes d'estimation de coût selon Bode.***

(Tiré de Bode, 1998)

Source : Cette figure a été tirée de l'article de Journal de Bode, *Neural Networks Early Cost Estimation*, p.29.

Se basant sur l'hypothèse que la méthodologie paramétrique est la plus adéquate en estimation de coût en phase de conception de produits (voir figure 1.3 tirée de Duverlie, page 14 de ce mémoire), et puisque les réseaux de neurones sont classés parmi les méthodes paramétriques, nous avons pensé aux «*Support Vectors machines*» *(SVM)*, en français, «*Machines à Vecteurs de Support*», dont le principe se rapproche énormément de celui des *ANN*, pour développer notre outil d'estimation de coût des produits.

Dans le chapitre qui suit, nous allons présenter cette nouvelle approche, ses fondements mathématiques et nous allons montrer comment elles peuvent être utilisées en estimation de coût.

CHAPITRE 2

BASES THÉORIQUES DES TECHNIQUES UTILISÉES DANS NOTRE APPROCHE

Dans ce chapitre, nous allons présenter la théorie et les principes des techniques que nous avons utilisées dans notre approche pour l'estimation des coûts. Nous allons expliquer nos choix et nous allons discuter les différentes utilisations de ces techniques citées dans la littérature.

D'abord, nous présenterons les SVM, comme nouvel outil d'apprentissage artificiel et, surtout, leur utilisation pour la régression. Puis, nous passerons aux algorithmes génétiques et nous décrirons leurs principes de fonctionnement. Finalement, nous présenterons la méthode de sélection des variables par les *fuzzy-curves*, et ses fondements mathématiques.

2.1 Les machines à vecteurs de support (SVM)

2.1.1 Qu'est ce que l'apprentissage supervisé ?

L'apprentissage supervisé est une technique d'apprentissage automatique (l'un des champs de l'intelligence artificielle) où l'on cherche à produire automatiquement des règles à partir d'une base de données d'apprentissage contenant des exemples de cas déjà traités. Plus précisément la base de données d'apprentissage est un ensemble de couples entrée-sortie (x_n, y_n) $1 \le n \le N$ *avec* $x_n \in X$ *et* $y_n \in Y$, que l'on considère être triées selon une loi XxY inconnue, par exemple x_n suit une loi uniforme et $y_n = f(x_n) + w_n$ où w_n est un bruit.

Le but de la méthode d'apprentissage supervisé est d'utiliser cette base d'apprentissage afin de déterminer une représentation compacte de f notée h et appelée *fonction de prédiction*, qui, à une nouvelle entrée x associe une sortie $h(x)$.

Le but d'un algorithme d'apprentissage supervisé est, de ce fait, de généraliser pour des entrées inconnues ce qu'il a pu « apprendre » grâce aux données déjà traitées par des experts de façon « raisonnable ». On distingue généralement deux types de problèmes que l'on cherche à résoudre au moyen de la méthode d'apprentissage supervisé :

- $Y \subset \Re$: Lorsque la sortie qu'on cherche à associer à une entrée est une valeur dans un ensemble des réels, on parle d'un problème de régression.
- $Y = \{1,....I\}$: Lorsque l'ensemble des valeurs de sortie est de cardinal fini, on parle de problème de classification car le but est en fait d'attribuer une étiquette à une entrée donnée.

Parmi les méthodes d'apprentissage supervisé connues à ce jour, citons :

- la méthode des moindres carrés;
- la méthode de K plus proches voisins;
- l'arbre de décision;
- les réseaux de neurones;
- les machines à vecteurs de support sur lesquelles nous allons nous attarder dans cette partie du chapitre afin d'expliquer le fonctionnement et les fondements mathématiques.

2.1.2 Les SVM comme méthode de régression

Les ***machines à vecteurs de support*** ou **séparateurs *à vaste marge***, aussi appelées ***machines à support vectoriel*** (en anglais *Support Vector Machines* ou SVM) constituent des estimateurs universels de fonctions, dont les fondements reposent sur la théorie de l'apprentissage statistique ou artificiel cité ci-dessus. Les SVM sont généralement plus

utilisées pour des problèmes de classification, cela n'empêche pas le fait qu'on peut en étendre le champ au problème de la régression, c'est-à-dire la recherche d'une fonction $h(x) = y$ dans $\Re$ tel que pour tous les points d'apprentissage $\{(x_i, y_i)\}_{1 \le i \le N}$ $h(x_i)$, soit le plus «proche» possible de y_i.

2.1.3 Principes et fondements mathématiques

La technique de régression par SVM, appelée aussi SVR pour (*Support Vector Regression*), est basée sur l'idée de déduire une estimation $\hat{g}[x]$ de la vraie relation $y = g[x]$ existante entre le vecteur des observations x et le vecteur sortie y.

où $x \in \chi = \Re^d$ *et* $y \in \Re$

Cette déduction se fait à partir d'un ensemble d'apprentissage composé de N échantillons tel que :

1- ε est la valeur maximale de déviations qui pourraient exister entre $\hat{g}[x]$ et les sorties désirées y_i *(i =1,...N).* En d'autre terme, nous définissons un tube de largeur ε autour des sorties désirées y_i dans lequel toutes les valeurs prédites devraient y être (figures 2.1, 2.2 et 2.3).

2- La fonction $\hat{g}[x]$ doit être la plus lisse et plane possible (*smooth and flat*) (Smola et al., 2002).

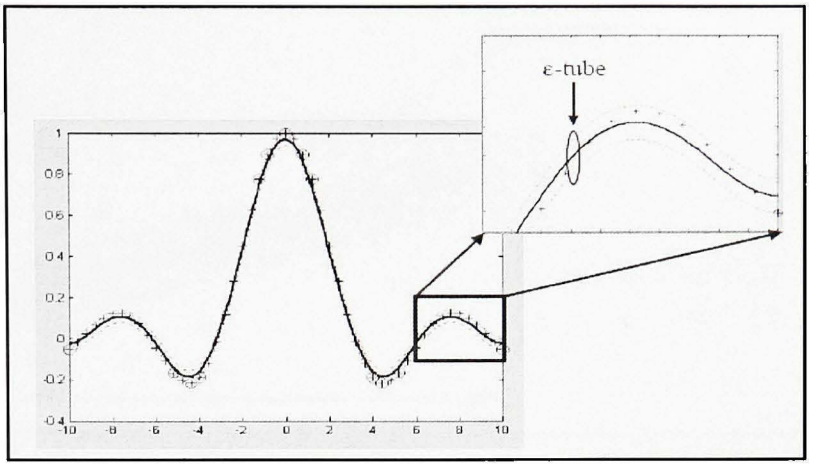

Figure 2.1 ***Illustration du ε-tube pour un cas d'approximation de fonction.***

Ces deux conditions peuvent être obtenues en transformant l'espace des entrées χ de dimension *d* à un espace de *re-description* de plus grande dimension tel que :

$\Phi(x) \in \Re^{d'}$ *(d'> d).*

Cette transformation a pour but de *lisser* au maximum la fonction $\hat{g}[x]$ et, par conséquent, de l'approximer à une façon linéaire comme suit :

$$\hat{g}[x] = \omega^* . \Phi(x) + b^* \qquad \text{ou } \omega^* \in \chi \qquad et \quad b^* \in \Re$$

D'après Vapnik (1999), la fonction linéaire optimale dans l'espace de *re-description* est celle qui minimise la fonction de perte suivante :

$$\psi(\omega, \xi) = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \qquad (1)$$

$$\text{Sujette aux contraintes} \quad \begin{cases} \left|\left((\omega.\Phi(x_i)) + b\right) - y_{ii}\right| & \leq \quad \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \qquad \forall i = 1,...,N$$

où ξ_i *et* ξ_i^* sont des variables d'écart introduites pour les échantillons qui ne se trouvent pas dans le ε- tube. Les figures 2.2. et 2.3 illustrent bien ce principe.

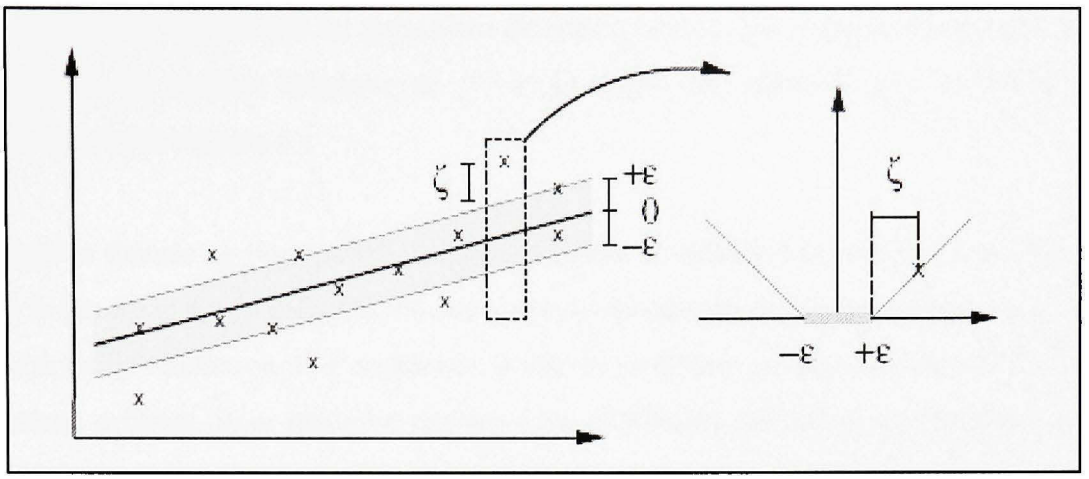

Figure 2.2 ***Illustration du principe des variables d'écart*** ξ_i ***et*** ξ_i^* ***dans le cas d'une régression linéaire***

(Tiré de Smola et al., 2002)

Source : Cette figure a été tirée de l'article de Journal de Smola et al., *A Tutorial on support vector regression*.p.200.

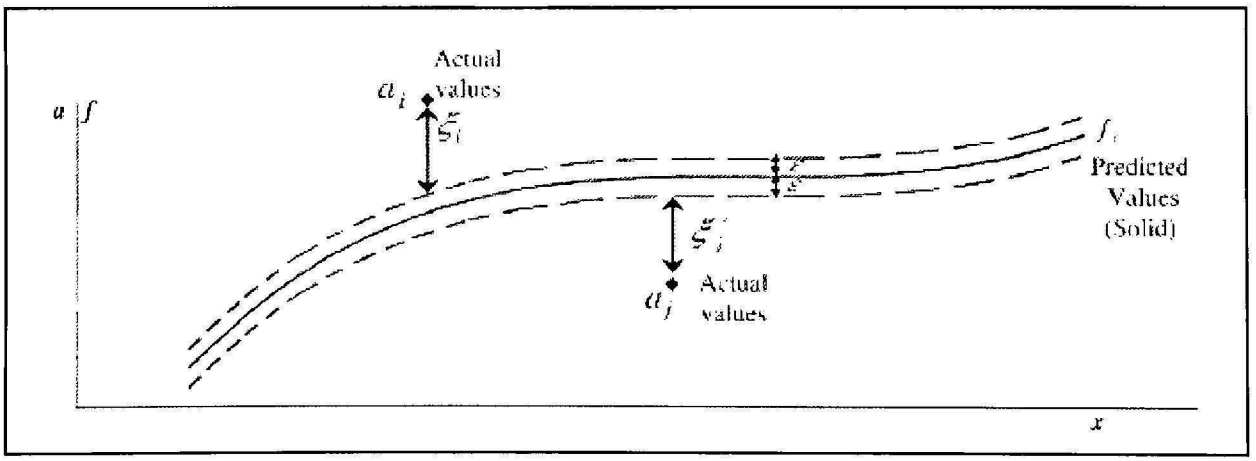

Figure 2.3 ***Illustration du principe des variables d'écart*** ξ_i ***et*** ξ_i^* ***dans le cas d'une régression non-linéaire.***

(Tirée de Vojislav, 2001)

Source : Cette figure a été tirée du livre de Vojislav.K. 2001. *Learning and soft computing- Support vector machines, Neural Networks and fuzzy Logic models.*

La constante C représente un paramètre de régularisation qui règle le compromis entre la complexité du modèle (planéité de *f)* et le degré de tolérance pour la déviation des échantillons par rapport à ε.

D'après la théorie de l'optimisation, tout problème d'optimisation possède une forme duale dans le cas où la *fonction objectif* et les *contraintes* seraient strictement convexes. Dans ces conditions, la résolution de l'expression duale du problème est équivalente à la solution du problème original. Pour résoudre ces types de problèmes, on utilise une fonction que l'on appelle *Lagrangien* qui incorpore les informations sur la fonction objectif et les fonctions contraintes et dont le caractère stationnaire peut être utilisé pour détecter les solutions. Plus précisément, le Lagrangien est défini comme étant la somme de la fonction objectif et d'une combinaison linéaire des contraintes dont les coefficients $\alpha_i \geq 0$ sont appelées des multiplicateurs de Lagrange ou encore variables duales.

Dans notre cas, on passe du problème primal au problème dual (1) en introduisant des *Multiplicateurs de Lagrange* (α_i) pour chaque contrainte. Ici, on a une contrainte par exemple d'apprentissage. Sans entrer dans les détails mathématiques (*nous référons le lecteur qui désire approfondir ces notions au travail de Smola et al. (2003)),* notre problème d'optimisation (1) se ramène à une forme duale dont la solution est l'estimation de la régression qui prend la forme suivante :

$$\hat{g}[x] = \sum_{i \in W} \alpha_i^* . \mathrm{K}(x_i . x) + b^*$$

où :

- *K(.,.)* est appelée ***fonction noyau***. Elle représente un produit scalaire dans l'espace de *re-description*. Le type de cette fonction influence sur la façon de modéliser la fonction de régression. (Le tableau 2.1 énumère quelques types de noyaux et la figure 2.3 montre les modélisations avec des noyaux différents.).

- α_i^* sont les multiplicateurs de Lagrange.
- W est sous ensemble des échantillons correspondant aux multiplicateurs de Lagrange non-nuls.

Ces échantillons de la base d'apprentissage dont les poids (Multiplicateurs de Lagrange (α_i)) sont non-nuls, représentent les vecteurs de support.

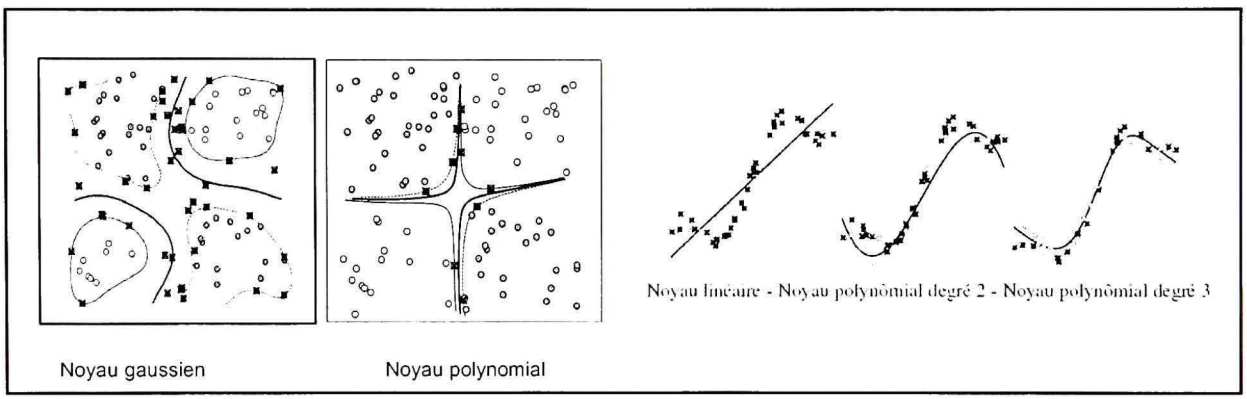

Figure 2.4 ***Différentes modélisations par différents types de noyaux pour un cas de classification et un cas de régression.***

(Tiré de Jeremy, 2006)

Source : Ces deux figures ont été tirées de la présentation de Jeremy, *Méthodes d'apprentissage avancées.*

Tableau 2.1

Différents exemples de noyaux

· Linéaire : $K(x.x') = x.x'$
· Polynomial : $K(x.x') = (x.x')^d$ *ou* $(c + x.x')^d$
· Gaussien : $K(x.x') = e^{-\frac{\|x-x'\|^2}{2\sigma^2}}$
· Laplacien : $K(x.x') = e^{-(\gamma\|x-x'\|)}$

2.1.4 Recherche des paramètres optimaux pour une bonne régression

Il est bien évident que la précision de l'estimation et la performance de ces SVR dépendent de la bonne initialisation des méta-paramètres ou hyper-paramètres des SVR, à savoir, C, ε, et le type de la fonction noyau K et ses paramètres. Ces hyper-paramètres sont :

Le noyau : la fonction noyau est utilisée pour la construction de la surface de décision non linéaire (hyper-plan) dans l'espace d'entrée.

Les paramètres du noyau : tels que la largeur du noyau gaussien ou le RBF ou le degré du noyau polynomial.

La constante de régularisation C : elle détermine le compromis entre la minimisation de l'erreur d'entrainement et la minimisation de la complexité du modèle.

La largeur du tube ε : c'est l'équivalent de l'erreur de prédiction de la base d'entraînement.

Ainsi, pour construire un modèle SVM efficace, plusieurs auteurs sont tout à fait favorables à l'hypothèse suivante : *bien choisir ces paramètres cruciaux* et plusieurs travaux ont été élaborés en ce sens. Citons, par exemple, le travail de Keerthi et al. (2002) qui utilise la technique de minimisation de marge, ou celui de Cawley (2006) qui propose la sélection de ces paramètres par la technique de validation croisées, ou (cross-validation), qui nécessite de larges bases de données ainsi qu'un long temps de calcul. Cherkasky et al.(2003) proposent une approche analytique pour la sélection des valeurs des paramètres C et ε, en utilisant un noyau gaussien. Ciwei et al.(2007) se sont basés sur les travaux des auteurs précédents pour développer une méthode appelée *flexible-C SVR* en vue de la prédiction du prix du marché d'électricité. Lin et al.(2005) développent une méthode de sélection systématique des paramètres ε et la largeur du noyau gaussien σ.

D'autres auteurs, comme Ping et al (2005), ont eu recours aux méthodes heuristiques pour la sélection de ces paramètres et ont utilisé le recuit simulé pour trouver les 2 paramètres C et ε. Il a été clairement démontré qu'on ne dispose pas de règles (*ou recettes*) pour la sélection de tels paramètres. En effet, chaque cas est unique puisque sa base de données est unique.

Trouver les paramètres optimaux des *SVR* pour aboutir à une bonne régression, ayant une précision élevée, s'avère alors une tâche cruciale surtout lorsque l'utilisateur de la régression par SVM recherche une robustesse et une finesse de résultats. Cela est le cas de l'estimation de coûts.

Pour cette raison, nous avons décidé, dans ce travail, de mieux faire ressortir les performances des *SVM*, en choisissant les bons paramètres. Au lieu d'aller par essais-erreurs à chaque estimation, perdant temps et efforts, nous avons eu l'idée d'utiliser les algorithmes génétiques, comme algorithmes d'optimisation pour sélectionner les hyper-paramètres des SVM. Dans notre cas, les travaux combinant les algorithmes génétiques aux réseaux de neurones pour le choix de la configuration de ces derniers (Kim G.H et al, 2004; Seo.KK,

2006; Kim K.J, 2002; Kim G.H et al.,2005 et Arifovic J et al., 2001) nous ont inspirés pour appliquer ces algorithmes aux SVM .

Avant de passer à la combinaison *Génétique-SVM*, nous présenterons dans la partie suivante, la théorie des algorithmes génétiques et leur principe de fonctionnement.

2.2 Les algorithmes génétiques (AG)

2.2.1 Introduction

Les algorithmes génétiques appartiennent à la famille des algorithmes évolutionnaires (un sous-ensemble des métaheuristiques). Leur but est d'obtenir une solution approchée, en un temps acceptable, à un problème d'optimisation lorsqu'il n'existe pas (ou qu'on ne connaît pas) de méthodes exactes pour le résoudre en un temps raisonnable.

Ce sont également des algorithmes stochastiques itératifs qui opèrent sur des ensembles de points codés, à partir d'une population initiale, et qui sont bâtis à l'aide de trois opérations issues de la sélection naturelle et dérivées de la génétique et des mécanismes d'évolution de la nature. Ces trois opérations sont ***le croisement, la mutation et la sélection***. Les deux premières sont des opérateurs d'exploration de l'espace, tandis que la dernière fait évoluer la population vers les *optima* d'un problème. De ce fait, on se rapproche par "*bonds*" successifs d'une solution dite *optimale*.

2.2.2 Principes des algorithmes génétiques

Le principe de base consiste à simuler le processus d'évolution naturelle dans un environnement hostile. Ces algorithmes utilisent un vocabulaire similaire à celui de la génétique. On parlera ainsi d'individus dans une population. L'individu est composé d'un ou de plusieurs chromosomes. Les chromosomes sont eux-mêmes constitués de gènes qui

contiennent les caractères héréditaires de l'individu. Les principes de sélection, mutation et croisement introduits dans ce cadre artificiel s'appuient sur les processus naturels du même nom.

Le premier pas dans l'implantation d'un algorithme génétique est de créer une population d'individus initiaux. En effet, les algorithmes génétiques agissent sur une population et non pas sur un seul individu. Par analogie avec la biologie, chaque individu de la population est codé par un chromosome ou génotype. Chaque chromosome est un point de l'espace de recherche. On lui associe la valeur du critère à optimiser. L'algorithme génère ensuite, de façon itérative, des populations d'individus sur lesquelles on applique les 3 opérations citées plus haut. La sélection a pour but de favoriser les meilleurs éléments de la population, tandis que mutation et croisement assurent une exploration de l'espace de recherche.

L'efficacité de l'algorithme génétique dépend fortement du choix du codage initial des chromosomes. Plusieurs techniques de codage existent et sont appliquées de nos jours. Le codage binaire, le codage réel, le codage à caractère multiple ou encore le codage sous forme d'arbres en sont quelques exemples.

Pour utiliser un algorithme génétique dans la résolution d'un problème spécifique, on doit disposer des cinq éléments suivants :

- Un principe de codage des éléments de l'espace admissible du problème, en éléments sur lesquels peuvent s'appliquer les trois opérateurs présentés ci-dessus. Ce codage intervient après une phase indispensable de modélisation mathématique du problème. Le choix du codage des données dépend du problème traité et conditionne l'efficacité (vitesse de convergence, précision, etc.) de l'algorithme génétique;

- Un mécanisme de génération de la population initiale. Cette population initiale qui sert de base aux générations futures, doit être *la plus hétérogène possible*;
- Une fonction d'évaluation «*f*» permettant de calculer l'adaptation de chaque élément au problème. Ce critère retourne une valeur de R^+ appelée *fitness*;
- Des opérateurs permettant de diversifier et d'améliorer la population d'une génération sur l'autre ainsi que d'explorer le plus largement possible l'espace admissible;
- Des paramètres dimensionnels : taille de la population, critère d'arrêt, probabilités de croisement (P_c) et de mutation (P_m).

2.2.2.1 La fonction d'évaluation ou fitness

Pour calculer le coût d'un point de l'espace de recherche (individu), on utilise une fonction d'évaluation ou fonction fitness *f*. L'évaluation d'un individu ne dépendant pas de celle des autres, le résultat fourni par la fonction d'évaluation va permettre de sélectionner ou de refuser un individu pour ne garder que ceux qui présentent le meilleur coût en fonction de la population courante : c'est le rôle de la fonction fitness. Cette méthode permet de s'assurer que les individus performants seront conservés, alors que les individus peu adaptés seront progressivement éliminés de la population.

2.2.2.2 Le croisement

Le croisement ou l'hybridation sélectionne les gènes parmi deux individus appelés parents. À partir de ces gènes seront générés les enfants qui héritent de certaines caractéristiques de leurs parents. La probabilité de croisement représente la fréquence à laquelle les hybridations (ou croisements) sont appliqués.

- S'il n'y a pas croisement, les enfants sont l'exacte copie des parents;
- S'il y a croisement, les enfants sont composées d'une partie de chacun de leurs parents;
- Si la probabilité de croisement est de 0%, la nouvelle génération est la copie de la précédente et si la probabilité est a 100%, tous les antécédents sont générés par hybridation.

Il y a plusieurs techniques de croisement qu'on retrouve dans la littérature parmi lesquels nous retrouvons :

- croisement à un point de coupure (simple) (figure 2.5);
- croisement multiple (multipoint) (figure 2.5);
- croisement uniforme;
- croisement arithmétique, etc.

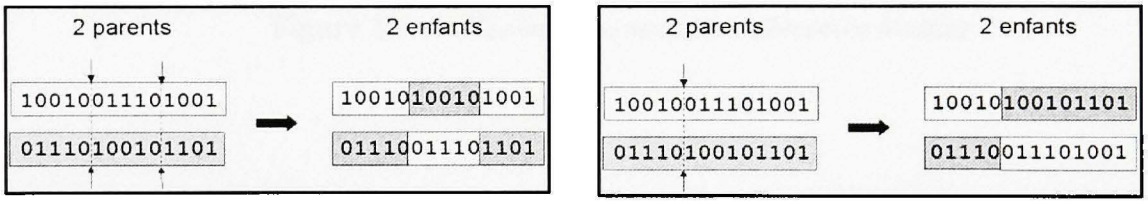

Figure 2.5 ***Exemples de croisement simple et de croisement en 2 points.***

2.2.2.3 La mutation

La mutation génère des erreurs de recopie afin de créer un nouvel individu qui n'existait pas auparavant. Le but est d'éviter à l'AG de converger vers des *extrema* locaux de fonction et de permettre la création d'éléments originaux. Si elle génère un individu plus faible, celui-ci est éliminé. La probabilité de mutation représente la fréquence avec laquelle les gènes sont mutés.

- S'il n'ya pas mutation, le fils est inséré dans la nouvelle génération sans changement;
- Si la mutation est appliquée, une partie du chromosome est changée.

La mutation est prévue pour éviter à l'AG de s'enliser dans des *optima* locaux, mais si elle est trop fréquente, l'algorithme est orienté vers une recherche aléatoire de la bonne solution. Tout comme le croisement, plusieurs techniques existent pour la mutation; nous citons, par exemple, la mutation aléatoire (figure 2.6) et la mutation non uniforme.

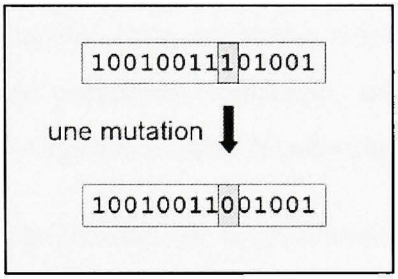

Figure 2.6 ***Exemple de mutation aléatoire binaire.***

2.2.2.4 La sélection

Cet opérateur est chargé de définir quels seront les individus de la génération ***P*** qui vont être dupliqués dans la nouvelle population ***P'*** et qui vont servir de parents (application de l'opérateur de croisement). Soit ***N*** le nombre d'individus de ***P***, on doit en sélectionner ***N/2*** (l'opérateur de croisement nous permet de repasser à ***N*** individus).

Cet opérateur est peut-être le plus important puisqu'il permet aux individus d'une population de survivre, de se reproduire ou de mourir. En règle générale, la probabilité de survie d'un individu sera directement reliée à son efficacité relative au sein de la population.

On trouve essentiellement quatre types de méthodes de sélection différentes :

- La méthode de la "loterie biaisée" (roulette wheel) de GoldBerg;
- La méthode "élitiste";
- La sélection par tournois;
- La sélection universelle stochastique.

2.2.2.5 Comment faire évoluer une population ?

Le mécanisme consiste à faire évoluer, à partir d'un tirage initial, un ensemble de points de l'espace vers le ou les optima d'un problème d'optimisation. L'ensemble du processus s'effectue à une taille de population constante, que nous notons N, de sorte que les générations successives comportent toutes N individus.

Afin de faire évoluer ces populations de la génération k à la génération k+1, les 3 opérations précédemment citées sont effectuées pour **tous** les individus de la génération k :

- Une **sélection** d'individus de la génération k est effectuée en fonction du critère à optimiser ou plus généralement du critère d'adaptation au problème (fitness), on cherche ainsi à privilégier la reproduction des *bons* éléments au détriment des *mauvais*.

Des opérateurs d'exploration de l'espace sont ensuite utilisés pour *élargir* la population et introduire de la nouveauté d'une génération sur l'autre.

- L'opérateur de **croisement** est appliqué avec une probabilité $\boldsymbol{P}_{cross}$ à deux éléments de la génération k (parents) qui sont alors transformés en deux nouveaux éléments (les enfants) destinés à les remplacer dans la génération k+1.

- Certaines composantes (les gènes) de ces individus peuvent ensuite être modifiées avec une probabilité $\boldsymbol{P}_{mut}$ par l'opérateur de **mutation**. Cette procédure vise à introduire de la nouveauté au sein de la population.

Cette procédure en 3 étapes est ensuite renouvelée à une taille de population constante. Les critères d'arrêts sont alors de deux natures :

1. Arrêt, après un nombre de générations fixé *a priori*. C'est la solution retenue lorsqu'un impératif de temps de calcul est imposé;

2. Arrêt, lorsque la population cesse d'évoluer ou n'évolue plus suffisamment rapidement, on est alors en présence d'une population homogène dont on peut penser qu'elle se situe à proximité du ou des optimums.

À ce stade, il est à noter qu' aucune certitude en ce qui a trait à la bonne convergence de l'algorithme n'est assurée. Comme dans toute procédure d'optimisation l'arrêt est arbitraire, et la solution *en temps fini* ne constitue qu'une approximation de l'optimum. Habituellement, une taille de 100 individus, 90% de probabilité de croisement et 1% de probabilité de mutation sont les valeurs les plus utilisées pour ces algorithmes.

2.2.3 Étapes d'un AG et organigramme

Pour appliquer un algorithme génétique à un problème donné, une série d'étapes doit être suivie :

- Générer une population initiale aléatoire;
- Évaluer chaque individu dans la population grâce à la fonction fitness;
- Choisir les chromosomes pour la reproduction en fonction de leurs résultats;
- Appliquer croisement et mutation pour créer de nouveaux individus;
- Remplacer l'ancienne population par la nouvelle;
- Répéter jusqu'a ce qu'une solution satisfaisante soit trouvée.

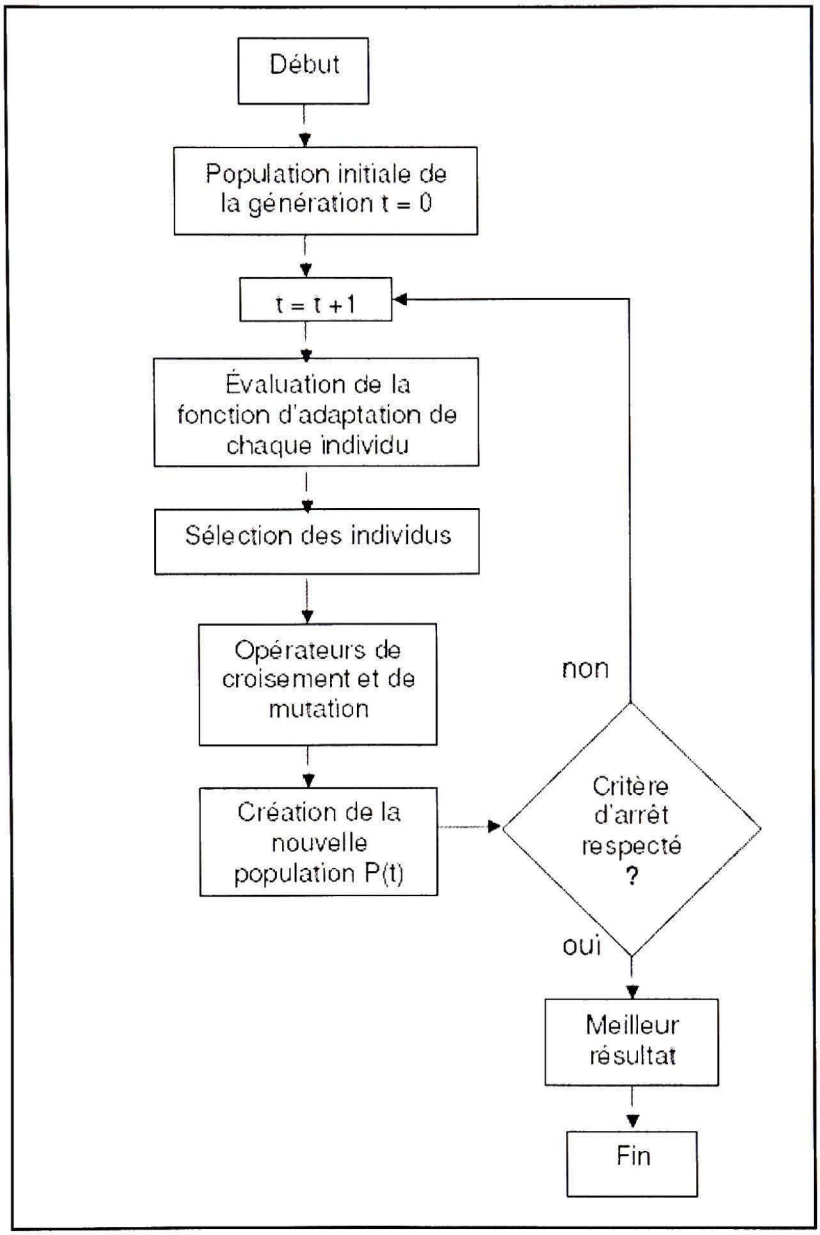

Figure 2.7 : ***Organigramme type d'un algorithme génétique.***

2.2.4 Aperçu sur le modèle SVM-Génétique proposé

Nous avons présenté, jusqu'ici, la théorie et le principe des algorithmes génétiques, nous allons, par la suite l'utiliser pour l'optimisation des hyper-paramètres des SVM afin de garantir une meilleure précision de notre estimation de coût.

Le chapitre, qui va suivre, présentera dans plus de détails la méthodologie mise en place. Cependant, nous allons présenter un bref aperçu de la technique utilisée pour appliquer notre algorithme génétique aux SVM.

Un individu (chromosome) dans notre cas sera une configuration possible de SVM, et sera composé de 4 gènes; à savoir un type de noyau particulier avec ses paramètres caractéristiques, une constante de régularisation C et une valeur de ε qui va nous fixer la largeur du ε-*tube*. La fonction *fitness* en ce qui nous concerne sera basée sur la performance d'estimation des SVM, en d'autres termes, l'évaluation sera calculée par l'erreur d'estimation.

2.3 Identification des variables les plus pertinentes

2.3.1 Mise en contexte

Dans ce travail, nous avons pensé à identifier les variables les plus influentes sur la sortie des SVM, à savoir les variables ou caractéristiques du produit qui sont les plus significatives sur son coût final, et nous avons choisi de le faire par la méthode des *fuzzy-curves*.

De ce fait, à part une estimation du coût, notre outil nous fournira un classement des caractéristiques ayant la pondération la plus importante sur le coût final du produit. À l'aide de cette information, l'utilisateur pourra savoir, sur quelle(s) caractéristiques(s), il pourra *jouer* pour rencontrer ses objectifs. Sachant que, dans certains cas, minimiser le coût total

pourra ne pas être le seul objectif à atteindre. Le concepteur pourra avoir une marge de manœuvre par rapport à certaines caractéristiques du produit.

En termes plus mathématiques, nous nous plaçons dans le contexte où l'on veut explorer les associations de plusieurs variables indépendantes (X_i) avec une variable dépendante (variable-réponse Y) bien identifiée. Parmi ces variables (X_i), on veut, par exemple, identifier celles qui décrivent le mieux la variable-issue Y.

Dans la littérature, plusieurs techniques ont été utilisées pour la sélection des variables d'un modèle. Ces méthodes sont souvent de nature statistique; comme par exemple, l'analyse de la variance plus connue sous le terme *ANOVA*, les modèles de régression, parmi lesquels nous retrouvons les procédures de sélection ascendante, descendante et sélection pas à pas et les plans d'expérience, pour ne citer que celle là.

Durant nos recherches sur ce point particulier, notre intérêt a été porté particulièrement sur une approche a la fois simple et robuste pour l'identification des variables. Cette méthode basée sur les *fuzzy-curves* ou *courbes floues,* introduite par Lin et Cunningham (1998) identifie rapidement les variables indépendantes X_i les plus significatives pour un modèle non linéaire.

2.3.2 Technique des *fuzzy-curves ou Courbes Floues*

La méthode d'identification des variables significatives par les *courbes-floues* ou *fuzzy-curves* (comme cela a été appelé par les fondateurs) a été trouvée parmi nos recherches dans les travaux de littérature. Les premiers auteurs qui l'ont fondée comme technique d'identification de variables pertinentes sont Lin et Cunningham en 1994. Ils ont mené plusieurs recherches pour la mettre au point et l'améliorer.(Lin et Cunningham, 1995, Lin et al.,1995,1996 et 1998). Nous explicitons dans ce qui suit les fondements de cette approches et le principe de son fonctionnement.

Supposons un problème ayant N variables d'entrée possibles ($x_1, x_2, ...x_N$) où N est de l'ordre des centaines, et une seule variable de sortie y. Supposons également que nous avons collecté l'information en M points relativement aux entrées x_i et à la sortie y .

Nous pouvons concevoir le tableau 2.2 suivant où les données ont été arrangées de façon plus lisible.

Tableau 2.2

Tableau des données pour la méthode des *fuzzy-curves*

(Tiré de Lin et al., 1998)

x^1	x^2	x^3	...	x^N	y
x_1^1	x_1^2	...	...	x_1^N	y_1
x_2^1	x_2^2	...	...	x_2^N	y_2
...	...	...	...	...	...
x_M^1	x_M^2	...	...	x_M^N	y_M

Source : Ce tableau est tiré de l'article de journal de Lin et al., *Nonlinear System Input Structure Identification : Two Stages Fuzzy Curves and Surfaces*, 1998.

Théoriquement, la fonction non-linéaire F modélise la relation entre les variables indépendantes x_i (entrées) et la variable dépendante y (sortie) bien que réellement cette relation soit non connue d'avance.

$$y = \Im \left(x^1, x^2, ..., x^N\right)$$

Le but de cette approche est d'identifier automatiquement et rapidement l'ensemble des variables les plus significatives par rapport à la sortie. Les auteurs de cette approche créent à partir des données de ce tableau des *fuzzy-curves* en suivant 3 étapes :

1. Pour chaque variable d'entrée x_i , nous traçons les M points $(x_{i,k}\ ;\ y_k)_{\ k=1,...M}$ dans chacun des espaces x_i-y ; $_{i=1,...,N}$. Cela est illustré par la figure 2.7 pour un système de 3 variables d'entrée x_1, x_2 *et* x_3 et pour 20 points de l'espace;
2. Pour chaque point $(x_{i,k}\ ;\ y_k)$, dans chacun des espaces x_i-y , nous créons les fonctions d'appartenance μ_k^i (*fuzzy membership functions*) telle que :

$$\mu_k^i\left(x^i\right) = \exp\left(-\left(\frac{x_k^i - x^i}{b^i}\right)^2\right), \quad k = 1,2,3....M$$

b^i étant en général variable entre 10% à 20% de la largeur de l'intervalle des entrées x^i.

Lorsque cette fonction d'appartenance μ_k^i est définie de cette manière, les relations entre les ensembles flous d'entrée $\boldsymbol{x^i}$ et de sortie y peuvent être définies par les règles floues suivantes :

$$\text{SI} \quad \boldsymbol{x^i}\ \textbf{\textit{est}}\ \mu_k^i\left(x^i\right) \text{ ALORS} \quad \text{y est } \text{y}_\text{k}$$

Pour la *deffuzification* de ces variables, nous produisons les courbes floues C_i pour chaque variable x_i comme illustré dans la figure 2.7, ou $C_i(x_i)$ est donné par :

$$\tilde{y}_c^i\left(x^i\right) = \frac{\sum_{k=1}^{M} y_k \mu_k^i\left(x^i\right)}{\sum_{k=1}^{M} \mu_k^i\left(x^i\right)}$$

Après ces trois étapes simples, nous sommes désormais capables de classer les variables x_i par ordre d'importance par rapport à leur influence sur la sortie y.

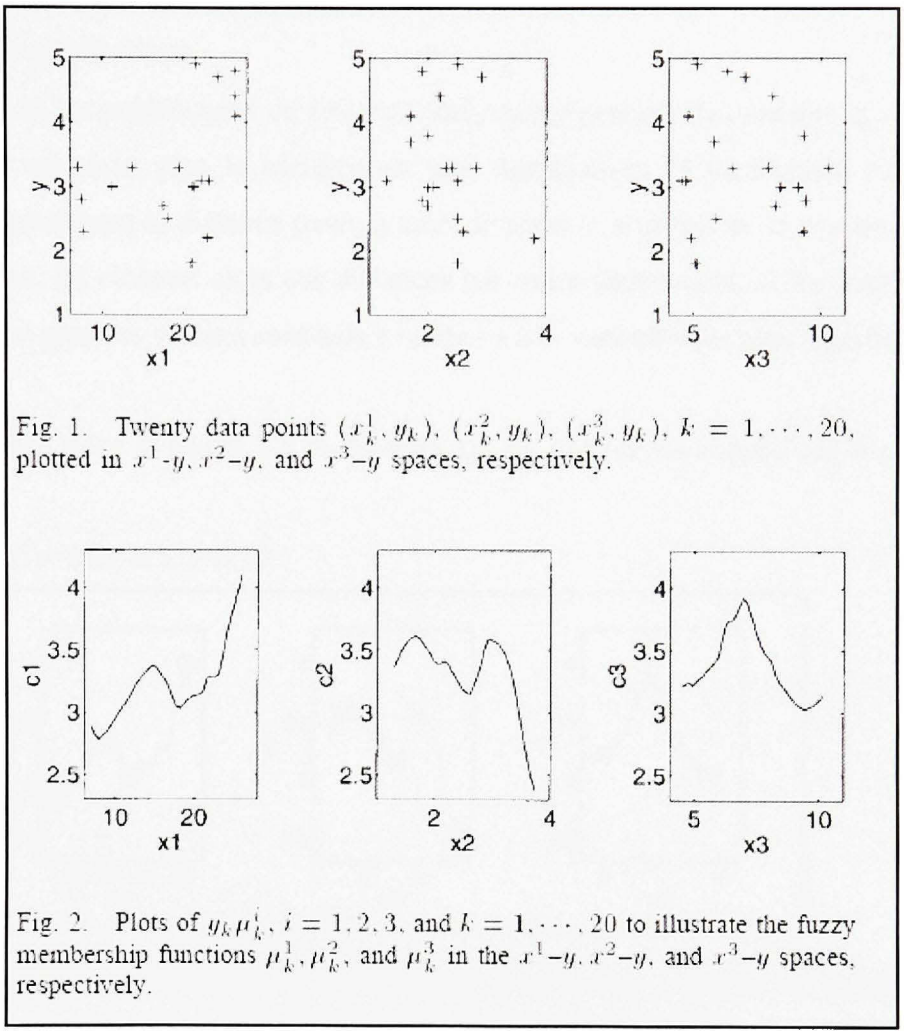

Fig. 1. Twenty data points (x_k^1, y_k), (x_k^2, y_k), (x_k^3, y_k), $k = 1, \cdots, 20$, plotted in x^1-y, x^2-y, and x^3-y spaces, respectively.

Fig. 2. Plots of $y_k \mu_k^i$, $i = 1, 2, 3$, and $k = 1, \cdots, 20$ to illustrate the fuzzy membership functions μ_k^1, μ_k^2, and μ_k^3 in the x^1-y, x^2-y, and x^3-y spaces, respectively.

Figure 2.7 ***Exemple de l'approche fuzzy-curves.***

(Tiré de Lin et al., 1998)

Source : Cette figure est tirée de l'article de journal de Lin et al., *Nonlinear System Input Structure Identification : Two Stages Fuzzy Curves and Surfaces*, 1998.

La question est comment peut-on le faire ?

Les auteurs *Lin et Cunningham* ont commencé leur publications par rapport à cette approche en 1994 et n'ont pas cessé d'améliorer la performance de leur réponse jusqu'en 1998. Nous présentons, dans l'ordre chronologique, les réponses données pour répondre à cette question selon leurs publications :

- Dans les publications de 1994 et 1995, ils ont proposé de conclure que plus la courbe C_i est plate, plus la variable est non significative, et vice versa. Par le calcul, ils déterminent la distance (*range*) entre le point le plus bas de la courbe et celui le plus haut. Ils classent alors ces distances par ordre décroissant, et les courbes qui ont les plus grandes valeurs sont celles relatives aux variables les plus significatives.

Par exemple, pour la figure 2.8 ci-dessous, ils ont calculé *les ranges* des 3 courbes comme suit :

C_1=1.32, C_2=1.24 et C_3=0.89.

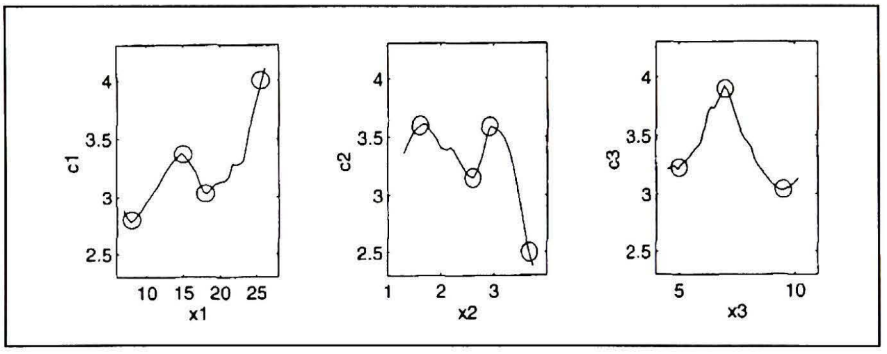

Figure 2.8 ***Illustration de la technique de calcul du «range» des courbes C_i pour décider de la pertinence des variables.***

(Tiré de Lin et al., 1995)

Source : Ce tableau est tiré de l'article de journal de Lin et al., *A new Approach to Fuzzy-Neural System Modeling,*, 1995.

Ils ont donc conclu que la variable C_1 est la plus significative par rapport aux trois.

- En 1996, ils ont utilisé l'erreur quadratique moyenne *MSE,* entre chaque *fuzzy-curve* et les données originales pour faire ressortir les variables les plus significatives. Cette erreur est calculée par :

$$\mathrm{MSE}_{c_i} = \frac{1}{M}\sum_{k=1}^{M}(c_i(x_{i,k}) - y_k)^2$$

Plus MSE est élevée, moins la variable est significative et vice versa.
En classant les MSE par ordre croissant, nous retrouvons les variables les plus pertinentes en premières positions.

- Finalement, en 1998, ils ont défini ce qu'ils ont appelé par *l'index de performance* P_y pour chaque variable d'entrée x_i.

$$\mathrm{P}\tilde{y}_c^i = \frac{1}{Mv_y}\sum_{k=1}^{M}\left(\tilde{y}_c^i\left(x_k^i\right) - y_k\right)^2$$

où $v_y = \sum_{k=1}^{M}\frac{(y_k - \bar{y})^2}{M}$ est la variance des y_k.

Ils concluent que plus cet index de performance est petit, plus la variable est significative. De ce fait, classer les index par ordre croissant, revient à classer les variables par ordre d'importance.

À la lumière de ces différentes améliorations, nous avons décidé de retenir cette dernière méthode dans notre travail, à savoir l'index de performance pour classer les variables.

Notons que nous avons vérifié cette approche à partir de différents exemples numériques que nous avons créés, allant d'exemples linéaires les plus simples, aux fonctions les plus complexes dont nous connaissions d'avance les positions des variables. Dans ce travail, l'approche nous a fourni des résultats très satisfaisants, ce qui nous a encouragés à l'adopter comme notre méthode d'identification des variables les plus pertinentes et significatives.

CHAPITRE 3

APPROCHE HYBRIDE PROPOSÉE POUR L'ESTIMATION DU COÛT FINAL D'UN NOUVEAU PRODUIT

L'approche hybride proposée, dans ce travail, est l'utilisation des SVM combinées aux algorithmes génétiques pour l'estimation de coût de produits ou de processus. Mis à part l'estimation du coût final d'un produit, notre travail permet également de trouver les spécifications de ce dernier qui sont les plus influentes sur son coût. Cela est donné à priori par la méthode des *courbes floues (fuzzy-curves)* et appuyé à posteriori par les algorithmes génétiques.

L'outil proposé se base sur l'historique des produits ayant des similarités du point de vue des spécifications techniques ou géométriques avec le nouveau produit. Les attributs de coût, dans ce cas, sont les spécifications techniques du produit même. Par exemple dimensions, masse, type de matériau, volume, etc. En général, ces attributs sont des valeurs quantitatives, mais, dans certains cas, nous nous retrouvons avec un attribut qualitatif. En effet, le type de matériau est généralement donné par le nom du matériau. Pour le quantifier, nous pouvons remplacer le nom du matériau par sa densité par exemple, ce qui nous donne une valeur quantitative qu'on pourra ajouter aux autres spécifications.

3.1 Étapes de la méthode

Tel que montré à la figure 3.1, les étapes de la méthode proposée sont les suivantes :

1 : Définition de la famille de produits (existants)

Au cours de cette étape, l'utilisateur jugera de quels produits, il alimentera sa base d'apprentissage pour permettre à l'outil d'estimation proposé, de modéliser le coût. Il définit

également les paramètres qui vont servir comme paramètres d'entrée à l'outil. On appellera ces paramètres les attributs (*ou cost-drivers*).

2 : Collecte de données

Une fois les attributs définis, l'utilisateur devrait chercher la base de données regroupant les différents produits, leurs attributs et leurs coûts respectifs. Il agencera les données dans une matrice, tel qu'indiqué dans le tableau 3.1.

Tableau 3.1
Matrice de données

Échantillon/Cost-drivers	***Cost-driver1***	***Cost-driver 2***	***Cost-driver 3***	.....	***Cost-driver p***	***COÛT***
Produit 1	Cd_1P_1				Cd_pP_1	$CoûtP_1$
Produit 2	Cd_1P_2					
Produit 3	Cd_1P_3					
Produit n	Cd_1P_n				Cd_pP_n	$CoûtP_n$

C'est très important de garder le même ordre pour les vecteurs des attributs. Un nouveau produit se définit donc par le même ordre d'attributs que celui des produits de la base d'apprentissage

3 : Identification des paramètres importants

La technique des *fuzzy-curves* s'applique à la base de données pour identifier et classer par ordre d'importance les attributs (*ou cost-drivers*) qui ont une plus grande influence sur le coût. C'est une technique d'identification des variables à priori (avant de les injecter au système de prédiction). Les étapes de la méthode ont été clairement annoncées dans la partie 2.3 *Identification des variables pertinentes* (p.37) de mémoire.

4 : Phase d'apprentissage par les SVR-GA

Une fois les données regroupées, nous les présentons en entrée des SVR-génétiques, le système effectue son apprentissage pour qu'il puisse être prêt à la prédiction du coût de nouveaux produits.

5 : Phase de test

Nous effectuons un test pour valider la performance du système en lui présentant en entrée les nouvelles données (spécifications) d'un produit ne faisant pas partie de la base d'apprentissage. Une fois le test effectué avec succès (erreur de prédiction acceptable), le système est prêt à estimer des coûts de nouveaux produits.

6 : Estimation de coût de nouveaux produits

Le système qui a effectué l'estimation sur la base de test avec succès, est capable d'estimer dorénavant le coût de nouveaux produits qui ne lui ont jamais été présentés auparavant.

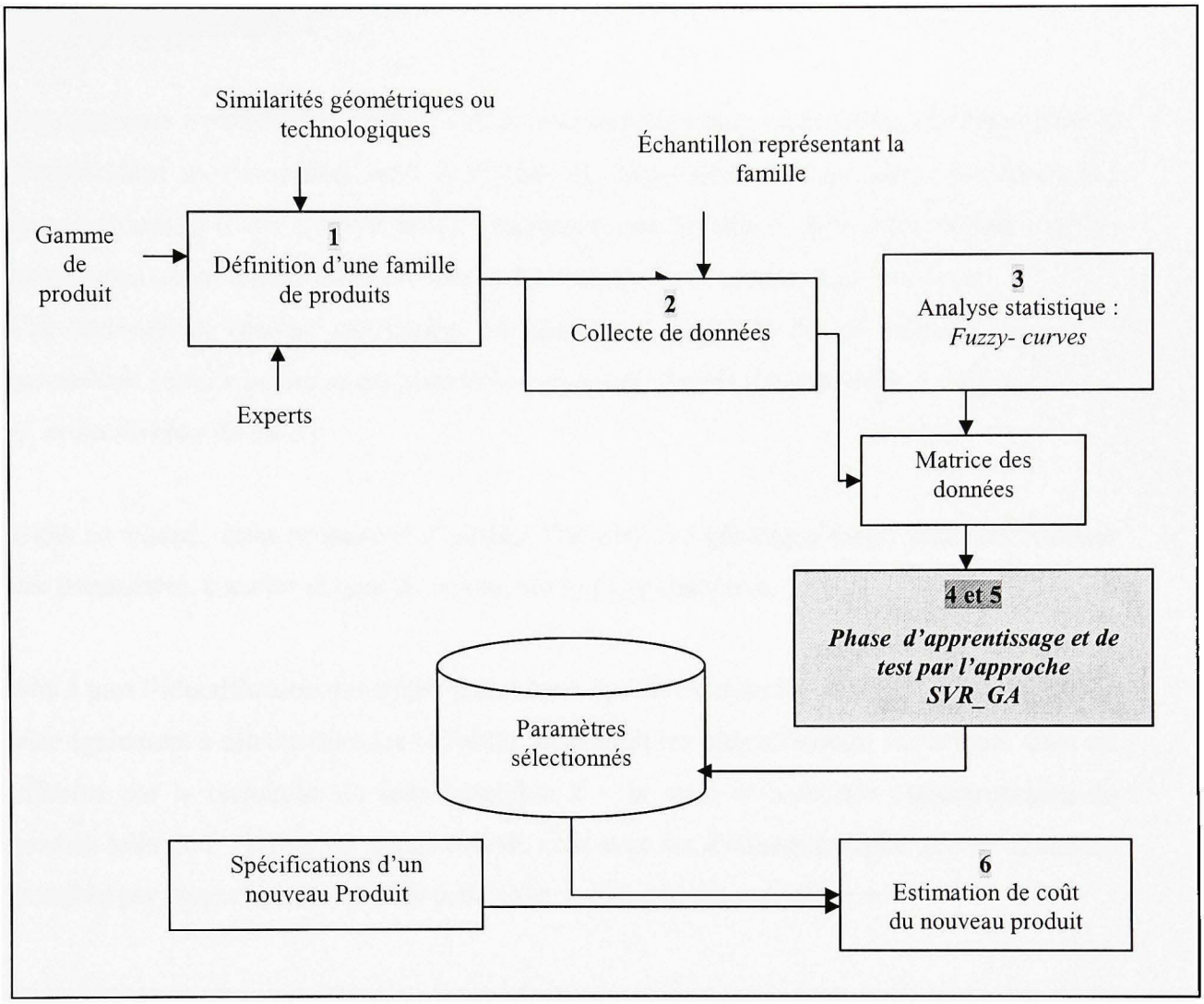

Figure 3.1 ***Approche globale d'estimation de coût et de sélection des paramètres les plus pertinents.***

Dans ce qui suit, nous détaillons la procédure d'apprentissage et de test par l'approche hybride de régression par les vecteurs de support (SVR) et les algorithmes génétiques (AG) que nous appellerons dorénavant SVR-GA. Nous présenterons aussi l'algorithme retenu.

3.1.1 L'étape SVR-GA

Les machines à vecteurs de support ont été utilisées dans bien de travaux que ce soit pour la classification ou récemment pour la régression. Cependant, bien qu'elles démontrent une bonne efficacité et des performances attrayantes, une définition appropriée de leurs hyper-paramètres, au préalable demeure une tâche cruciale pour assurer leur bon fonctionnement. Ces paramètres, comme mentionné, au chapitre 2 page 29 de ce mémoire, sont Ces paramètres sont *Le noyau et ses paramètres (largeur, degré), La constante de régularisation C, et La largeur du tube* ε.

Dans ce travail, nous proposons d'utiliser l'heuristique génétique (AG) pour sélectionner ces paramètres, à savoir le type du noyau, ses hyper-paramètres.

Mis à part l'identification des hyper-paramètres optimaux associés aux SVR, notre approche vise également à sélectionner les variables du produit les plus influentes sur le coût. Cela est effectué par la recherche du sous ensemble $d < m$ avec m taille des caractéristiques du produit telle que l'erreur de prédiction du coût avec les d caractéristiques soit la minimum possible par rapport aux erreurs de prédiction avec les m caractéristiques.

Cette recherche des variables les plus pertinentes est une recherche à posteriori, c'est-à-dire que le résultat est trouvé après une phase d'apprentissage et d'évaluation. Par analogie à la méthode des *courbes floues (fuzzy- curves)*, qui, elle, est une méthode à priori aboutissant à la même finalité. Nous l'appelons à priori, puisqu' elle s'effectue sur la base de données brute avant de les injecter dans le processus d'apprentissage et d'évaluation.

Le but d'effectuer ces deux phases, à priori et à posteriori de recherche de variables influentes sur le coût final, est de vérifier et, par conséquent, de confirmer leurs résultats mutuels. En effet, les deux résultats coïncident pour les applications de ce travail.

3.2 Sélection des variables les pertinentes et recherche des paramètres optimaux associés aux SVR par les algorithmes génétiques (AG)

En général, lors de la sélection de paramètres de modèles, plusieurs recherches ont recours à la technique par essais-erreurs. Cette procédure a bel et bien été utilisée pour les hyper-paramètres des SVM, tout comme l'architecture des réseaux de neurones. La sélection de variables à posteriori a aussi été réalisée dans bien des travaux par les procédures de sélection ascendante, descendante et sélection pas à pas. Ce qui ressemble a la technique d'essais-erreurs précédemment évoquée. Ceci demande en effet, un long temps de calcul, et requiert, en particulier pour la procédure d'essais-erreurs, des *coups de chance* pour que l'utilisateur se retrouve dans l'intervalle propice dans le grand espace de recherche des solutions possibles.

Dans ce travail, l'approche proposée, vise à optimiser simultanément les deux (2) sous ensembles cités plus haut, à savoir les paramètres des SVM et les caractéristiques du produit ayant le plus d'influence sur son coût. En général, le choix de variables d'entrée pour les SVM, (caractéristiques du produit dans notre cas) a une influence sur le design approprié de la machine SVR (combinaison des hyper-paramètres) et vice versa. De ce fait, l'optimisation simultanée de ces deux sous-ensembles est nécessaire pour une meilleure performance de prédiction. Ceci fut confirmé, dans ce travail, par le fait que la combinaison optimale des hyper paramètres de la machine SVR n'est pas la même dans le cas ou toutes les caractéristiques sont prises en considération ou non.

Deux modèles de prédiction, pour la méthodologie proposée, ont été appliqués. L'un tenant compte de toutes les caractéristiques du produit, et donc sans faire recours à la recherche par AG pour la sélection des variables (seule l'optimisation des paramètres de la machine a été effectuée), et l'autre avec la recherche simultanée des deux sous ensembles (variables du produit ***et*** paramètres de la machine). Les résultats trouvés sont présentés dans la section *Résultats, expérimentation et analyse* de ce chapitre pages 67 à 70 de ce mémoire.

3.2.1 L' algorithme SVR-GA

La procédure suivie pour établir le modèle de l'approche hybride SVR-GA comprend plusieurs étapes tel qu'illustré par la figure 3.5. Ces étapes sont les suivantes :

1-Prétraitement des données : Normalisation

L'avantage majeur de la normalisation des données est d'éviter la dominance d'attributs de grandes valeurs numériques par rapport à ceux de petites valeurs. L'autre avantage est d'éviter des difficultés de calcul pendant la phase d'apprentissage. Les auteurs Hsu et al. (2003) et Chang et al. (2007) soulèvent ce point dans leurs travaux. Nous avons opté pour le seuillage des données selon l'intervalle *[0,1]* par la formule suivante :

$$X_{normalisé} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

$X_{normalisé}$ est la valeur normalisée de l'attribut X.

X_{min} est la valeur minimale de l'attribut X parmi tous les échantillons.

X_{max} est la valeur maximale de l'attribut X parmi tous les échantillons.

Il est à noter que seules les valeurs des attributs (ou de variables) ont été normalisées. Le coût final est laissé à sa valeur réelle. Ainsi, nous avons obtenu des prédictions de coût ayant le même ordre de grandeur que les valeurs réelles. Nous avons évité ainsi de perdre la précision de la prédiction en voulant revenir aux valeurs réelles si on avait obtenu des valeurs de coût normalisées.

2- Codage et représentation des chromosomes

Les chromosomes vont jouer le rôle des individus des populations de l'algorithme génétique, ainsi il est très important de coder de façon judicieuse ces chromosomes afin de pouvoir balayer tout l'espace de recherche.

Les chromosomes sont composés de plusieurs gènes, dont chacun représente l'un des paramètres recherchés. Dans notre cas, les paramètres recherché seront les hyper-paramètres des SVR. Nous avons opté pour un codage binaire, et nos gènes représentent les hyper-paramètres de la machine ainsi que le sous-ensemble des variables sélectionnées. Nous codons donc, 5 gènes définis comme illustré par la figure 3.2.

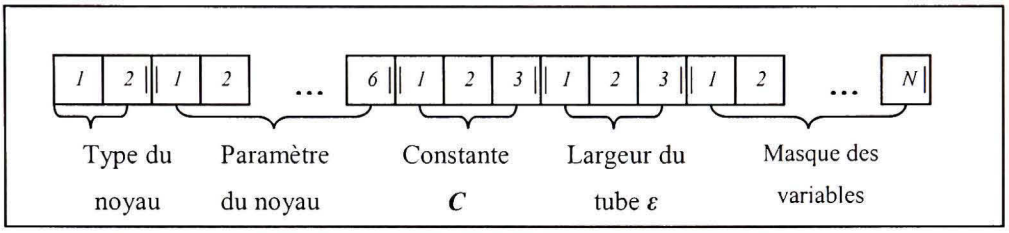

Figure 3.2 ***Codage et représentation du chromosome en 5 gènes.***

Génotype du noyau : composé par 2 bits, il code 3 types de noyaux : le gaussien, le noyau RBF (*Radial Based Function*) et le polynomial. Les équations de ces 3 types de noyaux sont données par le tableau 3.2.

Génotype du paramètre du noyau : composé par 6 bits, il code les valeurs allant de 0 à 6.3 par pas de 0.1. Nous avons décidé de nous arrêter à la valeur 6.3 puisque par exemple pour le noyau gaussien et selon Cherkassky et al. (2003), la largeur du noyau s'étend entre 0.1et 0.5 pour la valeur normalisée d'entrée. Pour le noyau polynomial, un degré 7 est suffisant pour des fonctions modélisant des régressions. Il est à noter que les degrés sont des nombres entiers, donc la valeur du phénotype, si elle est jumelée à un noyau polynomial, sera arrondie à l'entier suivant.

Génotype de la constante C : composé par 3 bits, il code des valeurs entre 10 et 10^7. La constante C est en général en puissance de 10.

Génotype de la largeur du tube ε : : composé par 3 bits, il code des valeurs entre 10^{-6} et 10. La largeur du tube ε est en général en l'inverse d'une puissance de 10.

Génotype ou Masque des variables d'entrée : composé par N bits. N représente le nombre total de variables ou d'attributs. Ce génotype est utilisé comme masque puisque, si la valeur du bit est 1, la variable est considérée dans le processus d'apprentissage, sinon, le bit étant 0, la variable en question sera éliminée et l'apprentissage se fera sans tenir compte de cet attribut.

Tableau 3.2

Les types de noyaux des SVR et leurs fonctions mathématiques

Noyau Gaussien : $K(x.x') = e^{-\frac{\lvert x-x'\rvert^2}{2\sigma^2}}$; le paramètre étant sa largeur σ	(1)
Noyau RBF : $K(x.x') = e^{-\gamma\lVert x-x'\rVert^2}$; le paramètre étant sa largeur γ	(2)
Noyau polynomial : $K(x.x') = (1 + x.x')^d$; le paramètre étant son degré d	(3)

3- Conversion des génotypes en phénotypes : Le génotype est la représentation binaire du gène tandis que le phénotype est sa traduction en valeur réelle considérée en phase d'apprentissage. Voici les conversions considérées dans ce travail.

Phénotype du noyau : étant codé sur 2 bits, les valeurs que peut prendre ce phénotype sont de 0 à 3.

-Si la valeur est 0 ou 1, le noyau est gaussien, (c'est le noyau le plus utilisé en régression par SVM).

-Si la valeur est 2, le noyau est un RBF;

-Si la valeur est 3, le noyau est polynomial.

Nous nous sommes arrêtés à ces 3 types de noyaux pour des raisons de simplification. D'autres types peuvent être pris en considération.

Phénotype du paramètre du noyau : étant codé sur 6 bits, les valeurs s'étendent entre 1 et 63. La valeur 0 sera écartée de la liste. La valeur donnée par le génotype sera divisée par 10, de cette manière; nous obtenons un phénotype variant entre 0.1 et 6.3.

Phénotype de la constante C **:** étant codé sur 3 bits, les valeurs s'étendent entre 0 et 7. La valeur V donnée par le génotype sera la puissance dans 10^V; de cette manière, nous obtenons un phénotype variant entre 1 et 10^7. (Ex : génotype : 011 → phénotype : $10^3 = 1000$).

Phénotype de la largeur du tube ε **:** étant codé sur 3 bits, les valeurs s'étendent entre 0 et 7. La valeur V donnée par le génotype sera diminuée de 6 et la valeur (V-6) sera la puissance dans 10^{V-6}; de cette manière, nous obtenons un phénotype variant entre 10^{-6} et 10. (Ex : génotype : 011 → phénotype : $10^{-3} = 0.001$).

Variables ou attributs considérés **:** Le génotype des variables nous donne directement quelles sont les variables à prendre en considération pour entraîner la machine à chaque phase d'apprentissage. (Ex : 010100....1 : les $2^{ème}$, $4^{ème}$ et dernière variables seulement sont considérées).

4-Définition de la fonction fitness

Pour chaque chromosome, représentant les paramètres de la machine SVR et les variables sélectionnées, la base d'apprentissage est utilisée pour entrainer le régresseur (ou le prédicateur) et la base de test est utilisée pour vérifier la performance de la prédiction. Cette performance de prédiction est évaluée par la fonction fitness. À chaque problème, la fonction fitness est différente selon l'objectif à atteindre. Dans notre cas, l'objectif à atteindre est la minimisation de l'erreur de prédiction.

$$fitness = Min(f)$$

La fonction f en question peut être traduite par l'une des mesures de performance du tableau 3.3 suivant. (Mesures de performance les plus utilisées pour la régression en littérature).

Tableau 3.3

Les différentes mesures de performance appliquées en cas de régression

(1) ***MAPE*** (Mean Absolute Percentage Error) : $\frac{\sum_{i=1}^{N}|R_i - E_i|}{N}.100\%$

(2) ***MSE*** (Mean Squared Error) : $\frac{1}{N}\sum_{i=1}^{N}(R_i - E_i)^2$

(3) $\boldsymbol{PE_i}$ (le pourcentage d'erreurs de chaque échantillon) : $\frac{R_i - E_i}{R_i}.100\%$ qui doit être $< \pm 15\%$

Dans un premier temps, nous avons opté pour la première mesure de performance, à savoir, le *MAPE* comme fonction de fitness de notre algorithme génétique. Cependant, une simple valeur de *MAPE*, ne pourra pas nous traduire, de façon *concrète* ou *explicite*, si les termes nous permettent de le dire, la qualité de notre estimation sans tracer les courbes des coûts réels et estimés respectifs. Effectivement, le tracé des courbes, et surtout, si elles se superposent bien, est une bonne preuve de la performance de notre estimation.

Pour avoir une traduction *concrète* ou *explicite* de la qualité de notre estimation, nous avons pensé à une autre fonction fitness, qui cette fois est une maximisation de nombre de points bien prédits. Elle est tirée de la mesure de performance (3) du tableau 3.3. Elle consiste à calculer le nombre de points ayant une déviation relative de $\pm$ 15% et de trouver la prédiction dans la base de test qui maximise ce critère.

$$fitness2 = Max(f)$$

La fonction f sera : $\dfrac{Nombre\ \ de\ \ point s\ \ P_i\ \ telque\ \ \dfrac{R_i - E_i}{R_i}.100\% \quad \leq \pm 15\%}{N} * 100\%$

Tel que N est le nombre d'échantillons dans la base de test.

Nous appellerons ce critère : ***pourcentage de points correctement prédis.*** Nous avons donc choisi finalement d'adopter ce deuxième critère comme fonction fitness dans notre travail.

Il est plus légitime de se fier à une estimation qui donne un *pourcentage de points correctement prédits* de 10 points sur un total de 12 (83%), qu'à celle qui donne un *MAPE* de 6.32. Le simple nombre 6.22 ne pourra pas nous communiquer la qualité de notre estimation à l'encontre du pourcentage de 88% qui nous prouve une bonne estimation.

En fait, ces deux mesures de performance *MAPE* et *pourcentage de points correctement prédis* vont de paire, c'est-à-dire qu'une estimation ayant un «*bon*» pourcentage de points bien prédits, aura nécessairement un MAPE faible. Cependant, dans certains cas, il se trouve que 2 estimations ayant un même *pourcentage de points correctement prédits* puissent avoir 2 *MAPE* différents, et dans ce cas, nous choisirons l'estimation qui a le *MAPE* minimal. Il est donc, intéressant de garder la mesure de performance *MAPE* comme second critère complémentaire à notre fonction fitness pour le choix de notre meilleure estimation.

5- Initialisation de la population

La population initiale est un ensemble d'individus choisis aléatoirement. Le nombre de chromosomes ou d'individus représente la taille de la population. Cette taille initiale sera maintenue durant les générations. Il est important de bien choisir ce premier paramètre de l'algorithme génétique puisqu'un choix non judicieux peut entraîner une convergence rapide vers un optimum local sans la possibilité d'exploration de l'espace de recherche. La taille de la population peut dépendre de la longueur du chromosome et du nombre de gènes qui le composent.

Dans ce travail, nous avons essayé plusieurs tailles de populations initiales et le choix s'est arrêté à 100.

6 - Partage de la base de données en base d'apprentissage et de test

Durant un processus d'apprentissage artificiel, la base de données est divisée en base d'entraînement et en base de test. La première est utilisée pour apprendre à la machine SVR comment prédire les résultats et la seconde est utilisée pour vérifier si le modèle de prédiction est bien adapté au problème donné.

Il est donc primordial que les deux bases soient complètement disjointes et qu'aucun échantillon de l'une n'interfère dans l'autre, sinon notre apprentissage est biaisé et le modèle de prédiction n'est pas tout à fait apte à prédire de nouvelles données. En général, la division entre les bases se fait d'une façon aléatoire, et la proportion est de 3/5 pour la base d'apprentissage par rapport à 1/5 pour la base de validation et 1/5 pour la base de test. Nous avons donc, dans un premier temps, procédé à cette division avant de lancer les machines SVR-GA dans leur procédure de prédiction.

Pour améliorer la performance de généralisation de notre estimateur (prédicteur) face à de nouvelles données, nous avons procédé, dans un deuxième temps, à un partage de la base de données en base d'entraînement et un ensemble indépendant en bases de test à travers une technique *de k-fold cross-validation.* Elle consiste à diviser la base de données en k ensembles, et de prendre comme base de test le k^{eme} ensemble pour le modèle entraîné par les k-1 ensembles restants. Le processus est répété k fois, pour que les k ensembles de test aient le tour d' être testés.

Pour chaque modèle, la moyenne des erreurs de prédiction des k bases de test est prise en considération comme mesure de performance. Nous avons retenu cette méthode de partage de la base de données pour notre travail puisqu'elle nous garanti que tous les échantillons

ont pris leur tour à être en phase de test, d'où une meilleure généralisation du modèle face à de nouvelles données.

7 -Entrainement des différents SVR (de la population initiale)

À cette phase, tout est prêt pour lancer le processus d'entraînement des SVR. Rappelons qu'un modèle de prédiction défini par une seule machine SVR avec ses différents paramètres est représenté par un individu de la population initiale. De ce fait, la population initiale est un ensemble de modèles de prédiction dont les paramètres (*noyau, paramètre du noyau, constante de régularisation C, largeur du tube et nombre de variables considérée*) sont choisis aléatoirement. Le processus est amorcé et les différents modèles accomplissent leur tâches, à savoir apprendre à dessiner la fonction de régression entre le coût final et les différentes caractéristiques du produit (phase d'apprentissage) et, par la suite, prédire le coût pour de nouveau produits (phase de test)

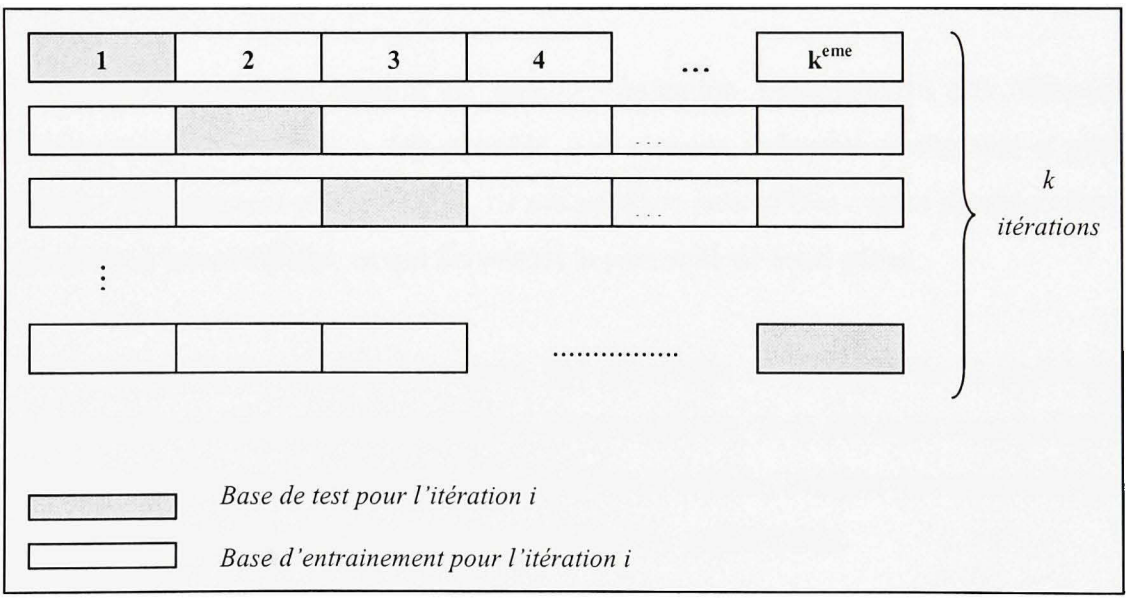

Figure 3.3 ***Technique de k fold cross-validation.***

8 - Évaluation des individus de la population initiale

Cette évaluation sera faite sur chaque chromosome de la base de test selon la fonction objectif choisie (mesure de performance). Un individu performant aura une meilleure mesure de performance par rapport aux autres et sera élu pour faire partie de la génération suivante. Un individu est un modèle de régression, évalué avec l'une des mesures choisies.

9- Sélection des individus pour le passage à la génération suivante (population intermédiaire ou mating-pool)

Dans la littérature sur les algorithmes génétiques, plusieurs méthodes de sélections ont été recensées. Dans ce travail, nous avons opté pour la méthode de ***sélection par tournoi*** qui va sélectionner les $N/2$ meilleurs individus de la population initiale. Cette technique consiste à sélectionner aléatoirement un nombre de sous ensembles (taille du tournoi) d'individus qui seront combattus entre eux et le meilleur individu de chaque sous ensemble sera retenu (selon la valeur de sa fonction *fitness*).

Cette étape est répétée jusqu'à ce que la génération intermédiaire soit remplie $(N/2$ chromosomes). Il est tout à fait possible que certains individus participent à plusieurs tournois : s'ils gagnent plusieurs fois, ils auront donc droit d'être copiés plusieurs fois dans la génération intermédiaire, ce qui favorisera la pérennité de leurs gènes.

10- Croisement : Le croisement a pour but d'enrichir la diversité de la population en manipulant les composantes des individus (chromosomes). Il se fait entre deux parents de la population initiale pour que les nouveaux individus (enfants) héritent des caractéristiques de leurs antécédents, déjà sélectionnés comme individus performants.

Dans notre travail, comme la population intermédiaire est composée de $N/2$ individus, le croisement se fait aléatoirement entre les paires de cette population sans remise. Il sera important alors, dans notre cas, de choisir la taille N comme entier divisible par 4 pour que les paires de chromosomes de la population intermédiaire puissent se former.

La probabilité de croisement, P_{cross}, intervient alors à ce moment. Elle permet de trouver le nombre de paires d'individus sur lesquelles le croisement va s'effectuer. En général, cette probabilité P_{cross} est assez élevée pour permettre une diversité de population considérable entre les générations. Nous avons choisi de croiser les paires de chromosomes aléatoirement selon les générations par la technique du ***croisement en un point.*** On choisit au hasard un point de croisement pour chaque couple de chromosomes. Notons que le croisement s'effectue directement au niveau binaire, et non pas au niveau des phénotypes.

11-Insertion : Après les deux opérateurs de sélection et de croisement, nous parviendrons à une nouvelle population de taille inferieure ou égale a N, selon que la probabilité P_{cross}, soit égale à 1 ou non. Nous avons déjà mentionné que la taille de la population demeure inchangée tout au long des générations, soit une taille de N chromosomes. Pour remplir la nouvelle génération de $N/2$ chromosomes restants (puisque la population intermédiaire compte seulement $N/2$), nous avons choisis pour ce travail de copier des individus de la population intermédiaire selon le nombre nécessaire pour que la taille de la nouvelle population atteigne N chromosomes. Cette technique d'insertion s'appelle ***sélection par Élitisme***. Par cette insertion, nous garantissons que les meilleurs chromosomes des populations précédentes ne périssent pas durant les générations suivantes.

12- Mutation : Une fois que la nouvelle population a atteint sa taille maximale souhaitée, soit N chromosomes, nous cherchons à garantir que notre algorithme soit susceptible d'atteindre tous les points de l'espace de recherche. Ceci est réalisé par des mutations aléatoires sur les bits des chromosomes de cette population. Le nombre de mutations de bits d'un chromosome dépend d'une certaine probabilité $P_{mut.}$ qui a été définie auparavant. Il est égal à $L \times P_{mut.}$ *(L étant la taille du chromosome)* En général cette probabilité varie entre 0.1% et 1%.

Nous avons choisi la technique de mutation aléatoire uniforme qui consiste à assigner un masque de mutation selon $L \times P_{mut}$ *à* chaque chromosome et à muter les bits qui

correspondent aux positions des bits du masque dont la valeur est 1. Si la valeur du bit du masque est 0, le bit correspondant à la position de ce bit dans le chromosome reste inchangé.

Muter (inverser) un bit, c'est changer sa valeur binaire par la valeur binaire complémentaire (*1 devient 0 et vice versa).* La position du bit à muter est choisie aléatoirement selon les chromosomes.

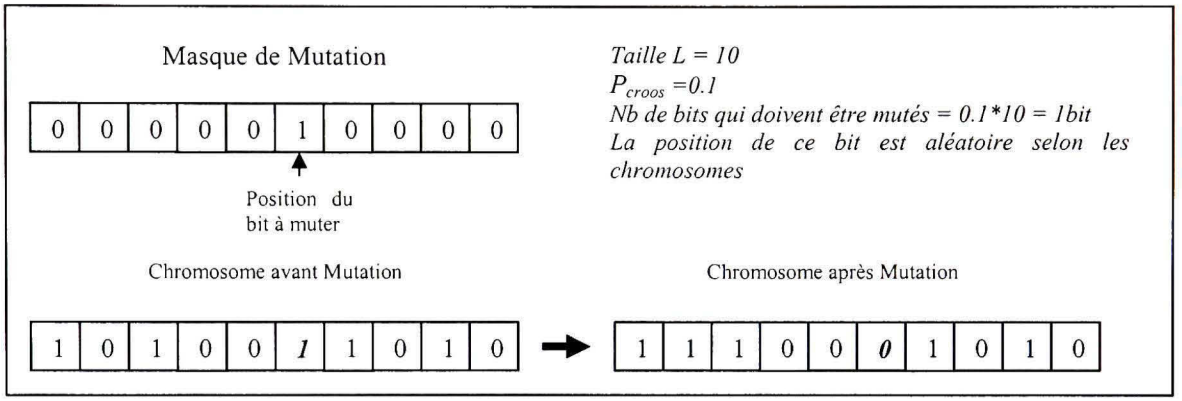

Figure 3.4 ***Exemple de mutation uniforme aléatoire.***

13- Critère d'arrêt : Si nous ne nous fixons pas un critère d'arrêt pour notre algorithme, notre évolution se fera à l'infini. C'est pour cette raison que nous nous fixons ce critère pour arrêter l'évolution des populations suivant les générations. En général, ce critère d'arrêt dépend généralement soit du temps (nombre de générations), soit de la valeur de la fonction fitness qu'on désire atteindre.

En classification, en général, ce critère dépend du taux de bonnes classifications dans la base de test. Dans le cas de la prédiction, comme nous ne pouvons pas connaître à l'avance l'ordre de grandeur de notre *taux d'erreurs* de la base de test, nous avons choisi d'arrêter selon un nombre maximal de générations G_{max}. Plusieurs essais ont été effectués afin de déterminer ce nombre G_{max}. Finalement, il sera choisi, à chaque cas, selon un compromis entre la vitesse de convergence et la valeur de la fonction fitness.

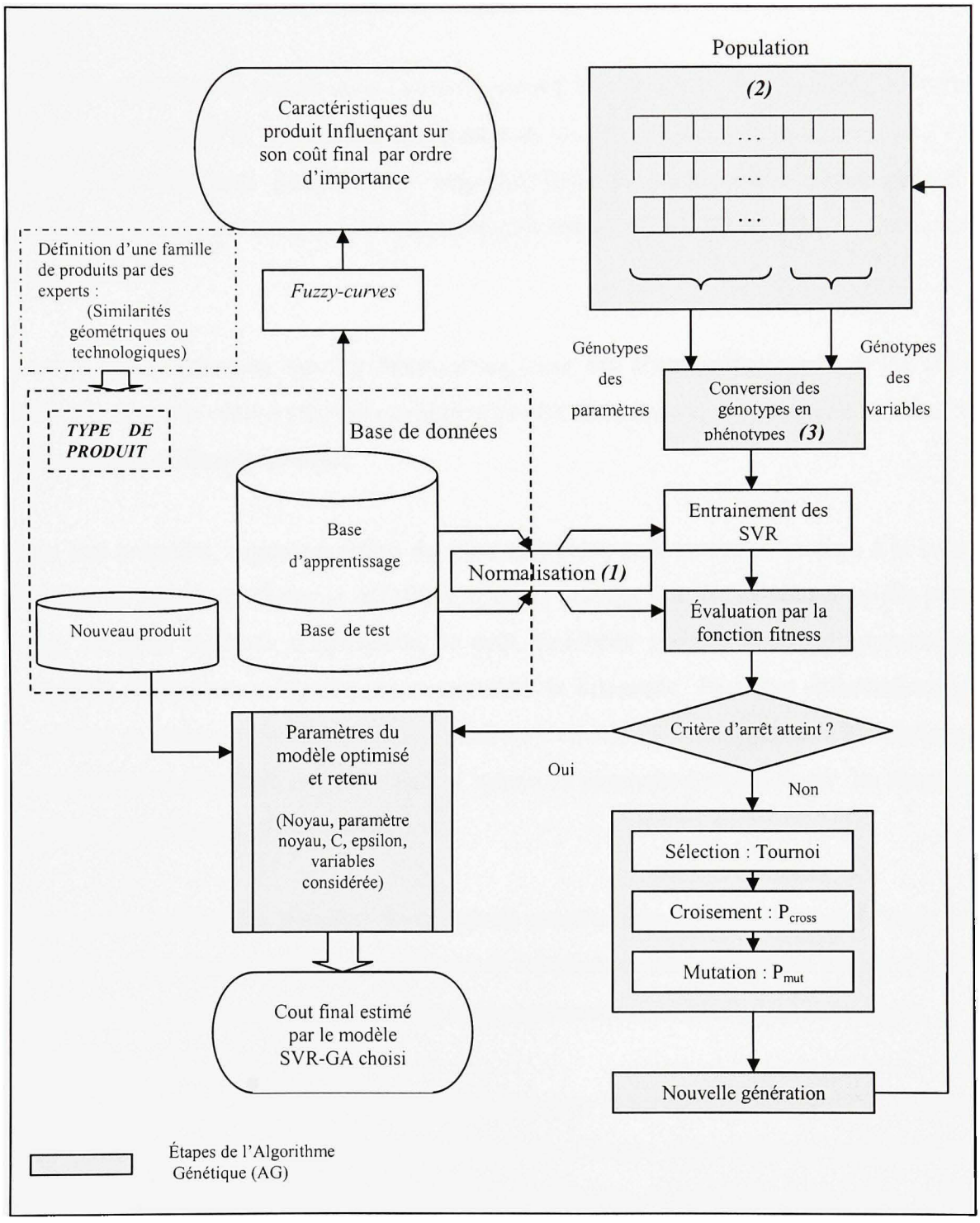

Figure 3.5 ***Architecture de l'approche SVR-GA proposée.***

3.2.2 Résultats, expérimentations et analyse

Nous avons réalisé ce travail sous l'environnement Matlab 2007. Pour la partie concernant la régression par SVR, nous avons eu recours au toolbox SVM, développé par *Canu et al.* (2005). La référence complète se trouvant dans la partie bibliographie. Nous le recommandons vivement pour d'éventuelles recherches sur les SVM, pour sa simplicité et sa robustesse.

Pour la partie *AG* ainsi que les *fuzzy-curves*, nous les avons programmés en MATLAB 2007. Nous avons réalisé plusieurs expériences afin d'analyser notre approche hybride et de comparer les différents modèles.

Pour une première expérimentation de notre approche, nous avons eu recours à la base de données du travail de *Zhang et al.*(1998). C'est le premier travail qui nous a guidés pour le choix de notre méthode d'estimation de coût, (méthode paramétrique par apprentissage) parmi les différentes méthodes existantes dans la littérature. Pour des informations plus détaillées, le lecteur pourrait se référer à l'article en question. L'application a été réalisée sur des produits d'emballage en production. L'approche proposée était basée sur les réseaux de neurones pour l'apprentissage artificiel.

Nous avons utilisé les données de ce travail comme base d'expérimentation pour notre approche, étant donné la largeur de la base de données présentée. Dans le chapitre 4 de ce travail, nous allons nous intéresser à des exemples concrets de fabrication mécanique.

La base de données se compose de 32 exemples pour la base d'apprentissage, 17 exemples pour chacune des bases de validation et de test. Chaque échantillon se compose de 21 caractéristiques qui définissent : la *forme, les dimensions, le matériau, la couleur, taille du lot de production, etc.* du produit d'emballage. Ces caractéristiques ont été choisies par un expert du procédé de fabrication qui connaissait, plus ou moins, les caractéristiques pouvant

contribuer au coût final du produit. Les coûts réels et estimés de la base de test (puisque c'est la base qui permet de juger de la pertinence de la méthode d'estimation) sont montrés dans le tableau 3.4. La 2ème colonne présente les valeurs réelles des produits. La 3ème présente les valeurs estimées par l'approche proposées par les auteurs. Les colonnes suivantes présentent les valeurs estimées par notre approche selon les 2 modèles proposés ci-dessous.

Modèle I : La phase de sélection à posteriori *(par les AG)* des variables les plus influentes n'est pas appliquée, toutes les caractéristiques du produit entrent en considération.

Modèle II : La phase de sélection à posteriori *(par les AG)* des variables les plus influentes est appliquée. Seules les caractéristiques choisies par l'AG sont retenues.

Dans un premier temps et pour les 2 modèles, une partition de la base de données, comme effectué par les auteurs a été faite. Pour chacun des deux modèles, plusieurs essais ont été effectués en changeant à chaque fois la taille de la population, le nombre de générations maximal, P_{cross} *et* $P_{mut.}$ et la forme de la fonction *fitness*. Nous avons pu ainsi choisir le meilleur modèle permettant d'avoir la meilleure prédiction du coût possible. Bien évidemment, en définissant la fonction fitness de notre modèle, nous calculons aussi la valeur de cette même fonction fitness pour l'approche des réseaux de neurones, à laquelle nous comparons nos résultats.

Dans un deuxième temps, nous avons utilisé la partition de la base de données par *validation-croisées* pour distinguer entre les données d'apprentissage et celles de test. Dans ce cas précis, nous avons comparé les résultats trouvés pour la base de test des auteurs à la portion équivalente de notre base de test, puisque cette dernière ne contiendra plus 17 exemples seulement, mais la totalité des exemples de la base d'apprentissage et de la base de test.

1- Identification des variables les plus pertinentes par ordre d'importance

Pour cet exemple, nous avons 21 caractéristiques définissant un produit. Par la méthode des *fuzzy-curves*, nous avons obtenu le résultat suivant :

Les caractéristiques du produit sont classées par ordre d'importance selon leur influence sur le coût final du produit :

9 ,3 , 4 , 1 , 6 , 8 , 5 , 10 , 21 , 11 , 18 , 16 , 19 , 17 , 2 , 14 , 15 , 20 , 13 , 7 et finalement 12.

Dans ce cas, la caractéristique 9 est la plus influente, la $12^{ème}$ est la moins influente.

2- Résultats de l'approche par réseaux de neurones

La 3ème colonne du tableau 3.4 montre les résultats obtenus par les auteurs, à travers leur approche par les réseaux de neurones. Nous avons calculé leur *MAPE* conformément à l'équation (1) de notre tableau 3.1, mesure de performance que nous avons retenue comme critère de comparaison. La valeur *MAPE* de leur approche est de ***7.45%*** et le pourcentage de points bien prédits, à savoir le pourcentage de points de la base de test dont l'erreur relative de prédiction ne dépasse pas ± *15%* est de ***100%.*** Cela veut dire que tous les coûts estimés de leur base de test (17 points) sont à ± *15%* de la valeur du coût réel. La figure 3.6 montre les courbes des coûts réels (bleue) vs les coûts estimés (rouge), ainsi que l'erreur relative de prédiction qui ne dépassent pas les 15%.

3- Résultats de l'approche par SVR-modèle I (base de données classique)

La 4ème colonne du tableau 3.4 montre les résultats obtenus par notre approche avec le modèle I, c'est-à-dire que toutes les caractéristiques du produit (*21 variables*) sont prises en considération durant la phase d'apprentissage des *SVR-GA*. La base de données est répartie d'une façon classique telle qu' effectuée par les auteurs. La valeur de notre *MAPE*, dans ce cas, est de ***6.2295%*** soit *16%* d'amélioration par rapport à l'approche *ANN*. Notre pourcentage de points bien prédits est aussi à ***100%.*** Ces résultats sont obtenus par les paramètres de la machine SVR suivants : ***(noyau polynomial de degré 2 ; C égal à 100, et epsilon égal à 0.01). Le***s paramètres de l'algorithme génétiques se présentent comme suit :

(taille de la population = 100, nombre de générations =10, P_{cross} =0.8 et P_{mut}=0.01). La figure 3.7 montre les courbes des coûts réels (bleue) vs les coûts estimés (rouge), ainsi que l'erreur relative de prédiction qui ne dépasse pas les 15%, mieux encore, elle est au dessous de 12%.

4- ***Résultats de l'approche par SVR modèle II (base de données classique)***

La 5ème colonne du tableau 3.4 montre les résultats obtenus par notre approche avec le modèle II, (c'est-à-dire que ***seulement*** quelques caractéristiques du produit sont prises en considération durant la phase d'apprentissage), toujours avec une répartition classique de la base de données. Les attributs significatifs, ressortis par la méthode des algorithmes génétiques (à posteriori) sont au nombre de 11 parmi 21, et sont les suivantes : (1ère, 2ème, 3ème, 4ème, 6ème, 9ème, 13ème,14ème, 17ème,18ème et 20ème). La combinaison de ces variables, nous a fourni le meilleur résultat de prédiction. Nous remarquons bien que ces caractéristiques font partie des attributs classés en premières positions par la méthode des *fuzzy-curves*. Nous remarquons par exemple, que la caractéristique 12, classée en dernier lieu par les *fuzzy-curves*, n'a pas été retenue. Cela confirme qu'elle n'est pas très significative pour l'inclure dans notre modèle de coût.

La valeur de notre *MAPE* dans ce cas, est de ***5.4819%*** soit *26%* d'amélioration par rapport à l'approche *ANN*. Notre pourcentage de points bien prédits est toujours à ***100%.***

Ces résultats sont obtenus par les paramètres de la machine SVR suivants : ***(noyau gaussien de largeur 2.3 ; C égal à 1000, et epsilon égal à 10^{-8}).*** Les paramètres de l'algorithme génétiques sont comme suit : ***(taille de la population = 100, nombre de générations =10, P_{cross} =0.8 et P_{mut}=0.01).***

5- *Résultats de l'approche par SVR modèle I (base de données repartie par cross validation)*

La 6ème colonne du tableau 3.4 montre les résultats obtenus par notre approche avec le modèle I, base de données répartie selon la technique de la validation croisée. La valeur de notre *MAPE* dans ce cas, est de ***6.9964%*** soit *6%* d'amélioration par rapport à l'approche *ANN*. Cette valeur est supérieure à celle trouvée par notre approche avec le modèle I et cela est dû au fait, que notre base de test comprend 49 (32+17) échantillons cette fois et non 17, c'est tout à fait logique que notre *MAPE* augmente. Par contre, *MAPE* est toujours inferieure à celui de l'approche par *ANN*, qui a été trouvée avec une base de test de 17 échantillons seulement. Notre pourcentage de points bien prédis est toujours à ***100%.***

Ces résultats sont obtenus par les paramètres de la machine SVR suivants : ***(noyau gaussien de largeur de 5.6; C égal à 10^6, et epsilon égal à 10^{-2}).*** Les paramètres de l'algorithme génétiques sont comme suit : ***(taille de la population = 100, nombre de générations =10, P_{cross} =0.8 et P_{mut}=0.01).***

6- *Résultats de l'approche par SVR modèle II (base de données repartie par cross validation*

La 7ème et dernière colonne du tableau 3.4 montre les résultats obtenus par notre approche avec le modèle II avec répartition des données par *cross-validation*. Les caractéristiques qui ont été prises en considération durant la phase d'apprentissage et de test, pour donner la meilleure prédiction, sont au nombre de 9 et sont les suivantes : (2ème, 4ème, 5ème, 9ème, 10ème, 10ème, 15ème, 18ème et 19ème). Nous avons également remarqué que cette combinaison n'est pas l'unique qui donne un *pourcentage de points correctement prédits* de *100%,* mais d'autres combinaisons d'attributs telles que par exemple (1ère, 2ème, 3ème, 4ème,6ème, 9ème, 13ème, 14ème, 17ème, 18ème et 20ème) le font à quelques chiffres près pour ce qui est de *MAPE.* Encore une fois, nous remarquons bien que la combinaison choisie par les algorithmes génétiques vient confirmer le résultat trouvé par les *fuzzy-curves*, pour ce qui est de l'identification des variables les plus pertinentes sur le coût. Le fait d'avoir plusieurs

combinaisons possibles de variables, nous indique que non seulement les variables en question, prises individuellement sont significatives sur le coût final, mais aussi que certaines interactions entre-elles le sont aussi. Nous faisons remarquer au lecteur que la phase d'identification des attributs les plus significatifs par les algorithmes génétiques (à posteriori) n'ordonne pas ces attributs par ordre d'importance comme le font les *fuzzy-curves*, mais nous fourni la combinaison de ces attributs qui modélise le coût final avec l'erreur minimale par rapport au coût réel.

La valeur de notre *MAPE* dans ce cas, est de ***5.7035*** soit ***23.4 %*** d'amélioration par rapport à l'approche *ANN*. Notre pourcentage de points bien prédits est toujours à ***100%.*** Ces résultats sont obtenus par les paramètres de la machine SVR suivants : ***(noyau gaussien de largeur 4.1 ; C égal à 10^4, et epsilon égal à 10^{-5}).*** Les paramètres de l'algorithme génétiques sont comme suit : ***(taille de la population = 100, nombre de générations =10, P_{cross} =0.08 et P_{mut}=0.01).***

Les figures 3.9 et 3.10 montrent, respectivement, les allures des moyennes des *MAPE et des pourcentages de points correctement prédits* durant les 10 générations. Il est bien clair que la moyenne des *MAPE* diminue tout le long des générations, ce qui démontre une convergence de l'algorithme vers le minimum de la fonction fitness (*MAPE*). La moyenne *des pourcentages de points correctement prédits*, quant à elle, augmente progressivement le long des générations, pour converger vers le maximum, ce qui confirme l'hypothèse émise plus haut et qui stipule que les deux mesures de performance vont de paires, et que, lorsque l'une diminue, l'autre augmente.

Pour cet exemple, nous avons opté pour une taille de population de ***100*** et nous nous sommes arrêtés à ***10*** générations, les résultats étaient satisfaisants et augmenter le nombre de ***P*** ou G_{max} n'aurait pas était nécessaire. Les temps de calcul ne dépassaient pas les quelques minutes (4 à 5) avec un bon processeur de *2GHertz*, ce qui est assez rapide et efficace pour une estimation de coûts.

Nous pouvons remarquer que la bonne collecte de données et leur prétraitement jouent un rôle important pour une bonne estimation de coûts. En effet, la base de données crées par les auteurs (Zhang et al., 1998) était déjà normalisée d'une certaine manière. Notre seuillage dans l'intervalle [0,1] a aussi été un facteur d'amélioration des résultats.

Nous avons également constaté que le type de noyau *Gaussien* est le mieux à choisir pour faire de telles estimations. Nous sommes en fait en train d'effectuer des modélisations et des approximations de fonctions, et le noyau *Gaussien* est le plus utilisé pour ce genre d'applications. Le noyau *Polynomial,* quant à lui, est moins adaptable à ce genre d'applications, surtout si les fonctions à modéliser ne sont pas polynomiales; et dans notre cas, il est difficile de savoir le type de fonctions modélisant nos coûts.

La séparation de la base de données en base d'apprentissage et en base de test est plus pertinente en utilisant la technique de la validation croisée (ou *cross-validation*). En effet avec cette technique, nous garantissons que tous les échantillons sont passés en phase de test et donc, d'où une meilleure généralisation face à de nouvelles données.

La comparaison des résultats fournis par les deux modèles I et II de l'approche proposée (voir tableau 3.4 p.72-73), montre que le modèle II donne de meilleures performances de prédiction que le modèle I, ainsi qu'il nous fourni les paramètres du produit les plus influents sur le coût total.

Finalement, nous mettons l'accent sur la meilleure performance de notre approche *SVR-AG* par rapport à celle des *ANN* en se basant sur les résultats de prédiction fournis par l'exemple traité tel que montré dans le tableau 3.4 de ce chapitre.

Tableau 3.4

Tableau des résultats des coûts estimés par les différentes approches

		Base de test classique			Base de test par Cross-Validation	
Échantillon	**Coût Final Réel ($)**	**Coût estimé par ANN**	***Coût estimé par SVR-GA Modèle I***	***Coût estimé par SVR-GA Modèle II***	***Coût estimé par SVR-GA Modèle I***	***Coût estimé par SVR-GA Modèle II***
1	***0,701***	*0,779*	0.7550	0.7078	0,7949	0.7694
2	***1,47***	*1,411*	1.5795	1.4437	1,4408	1.4724
3	***1,422***	*1,435*	1.5795	1.4169	1,3414	1.4410
4	***1,489***	*1,628*	1.4916	1.5708	1,5024	1.5853
5	***1,475***	*1,547*	1.5590	1.5638	1,4968	1.5509
6	***1,75***	*1,848*	1.7496	1.7584	1,6941	1.8656
7	***1,016***	*1,059*	1.0762	1.0327	0,958	1.0022
8	***0,566***	*0,552*	0.5425	0.5982	0,5375	0.5391
9	***0,565***	*0,562*	0.5421	0.5808	0,542	0.5467
10	***0,955***	*0,881*	0.8467	0.8700	0,8736	0.8936
11	***1,327***	*1,431*	1.4999	1.3257	1,356	1.4946
12	***1,488***	*1,37*	1.5026	1.3246	1,3297	1.4997

13	*1,011*	*0,98*	0.9135	1.0847	1,0976	0.9526
14	*0,878*	*0,992*	0.9417	0.9326	0,9041	0.9378
15	*2,374*	*2,243*	2.3166	2.2183	2,1532	2.3675
16	*1,384*	*1,294*	1.3093	1.3600	1,3789	1.3517
17	*1,564*	*1,464*	1.4726	1.4578	1,3862	1.4287
MAPE		***7.4529***	***6.2295***	***5.4819***	***6.9964***	***5.7035***
% d'amélioration (*)		-	***16.41%***	***26.44%***	***6.13%***	***23.45%***
% des points correctement prédits		***100%***	***100%***	***100%***	***100%***	***100%***
Paramètres SVR		-	***Noyau =Polynomial Degré = 2 C = 100 Epsilon =0.01***	***Noyau =Gaussien Degré = 2.3 C = 1000 Epsilon =10^{-8}***	***Noyau =Gaussien Degré = 5.6 C = 10^{6} Epsilon =10^{-1}***	***Noyau =Gaussien Degré = 4.1 C = 10^{4} Epsilon =10^{-5}***
Variables considérées :		***toutes***	***toutes***	***11 parmi 21***	***toutes***	***9 parmi 21***

(*) Calculé par rapport aux résultats de la 1ère colonne, obtenus avec la technique *ANN*.

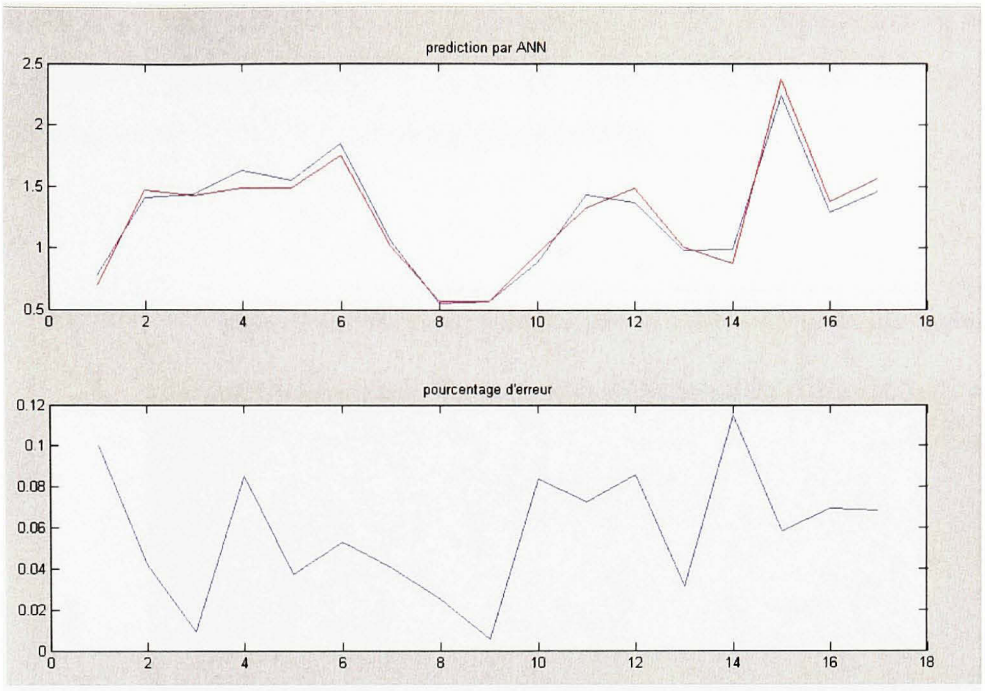

Figure 3.6 ***Courbe du coût réel VS le coût estimé par l'approche ANN et courbe de l'erreur absolue relative pour chaque échantillon.***

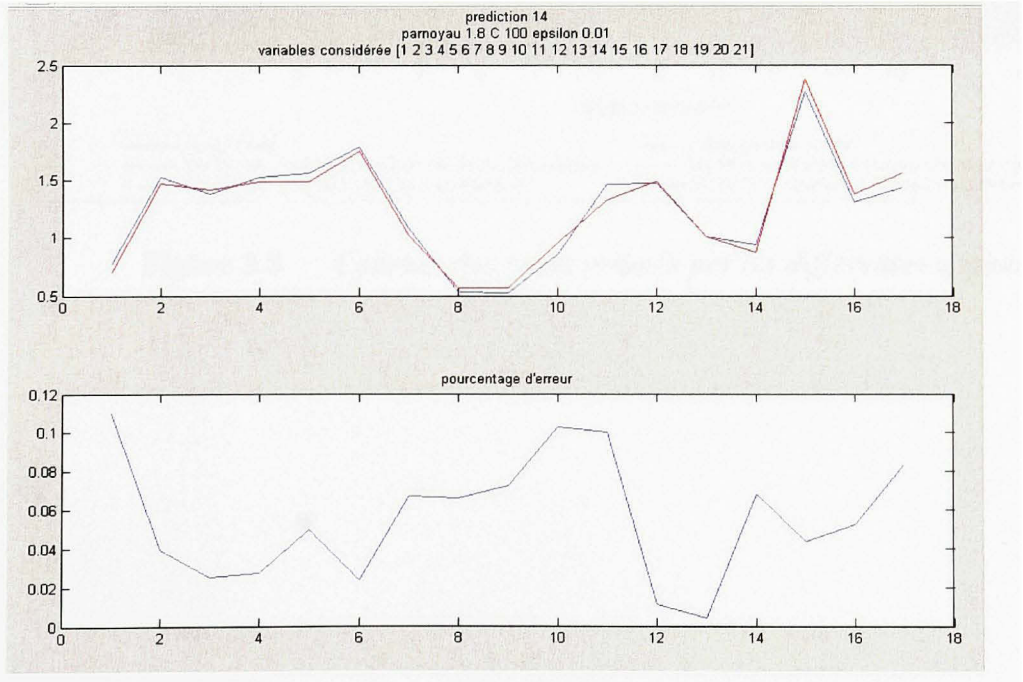

Figure 3.7 ***Courbe du coût réel VS le coût estimé par l'approche SVR-GA modèle I et Courbe de l'erreur absolue relative pour chaque échantillon***

La figure 3.8 suivante, regroupe les différentes courbes d'estimations de coût effectuées, les 4 réalisées par notre approche, celle que nous avons prise comme base de comparaison (approche par ANN) et finalement les coûts réels.

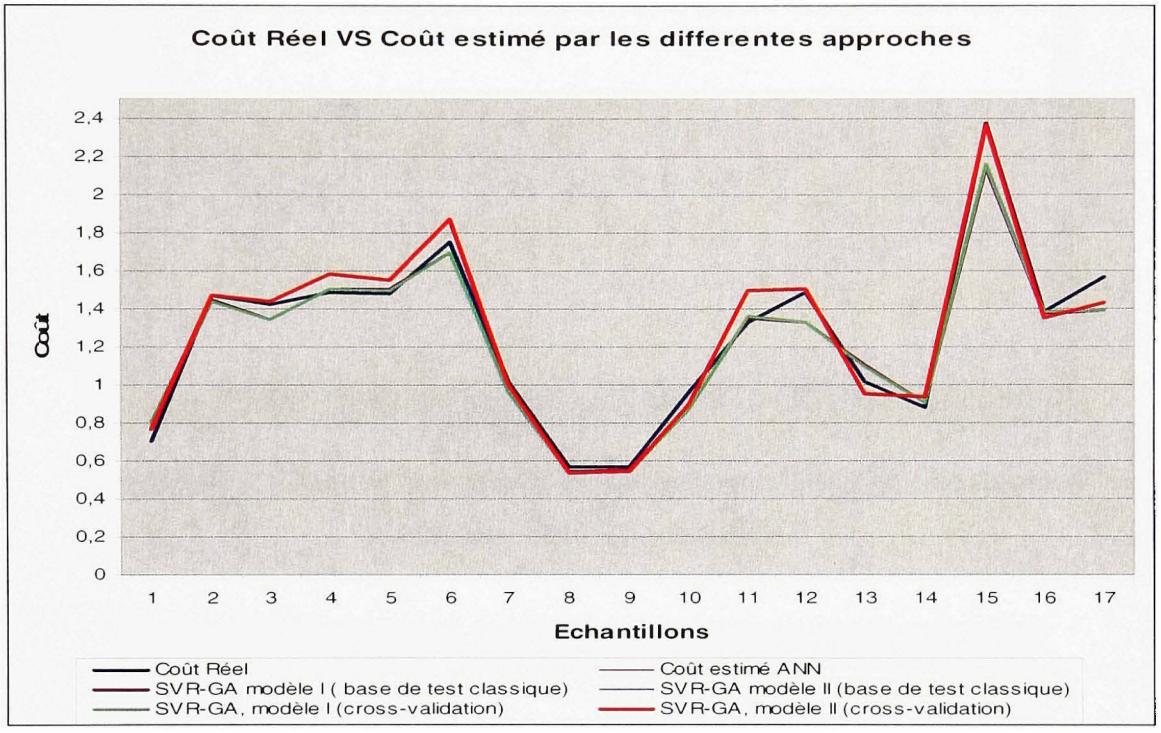

Figure 3.8 ***Courbes des coûts estimés par les différentes approches.***

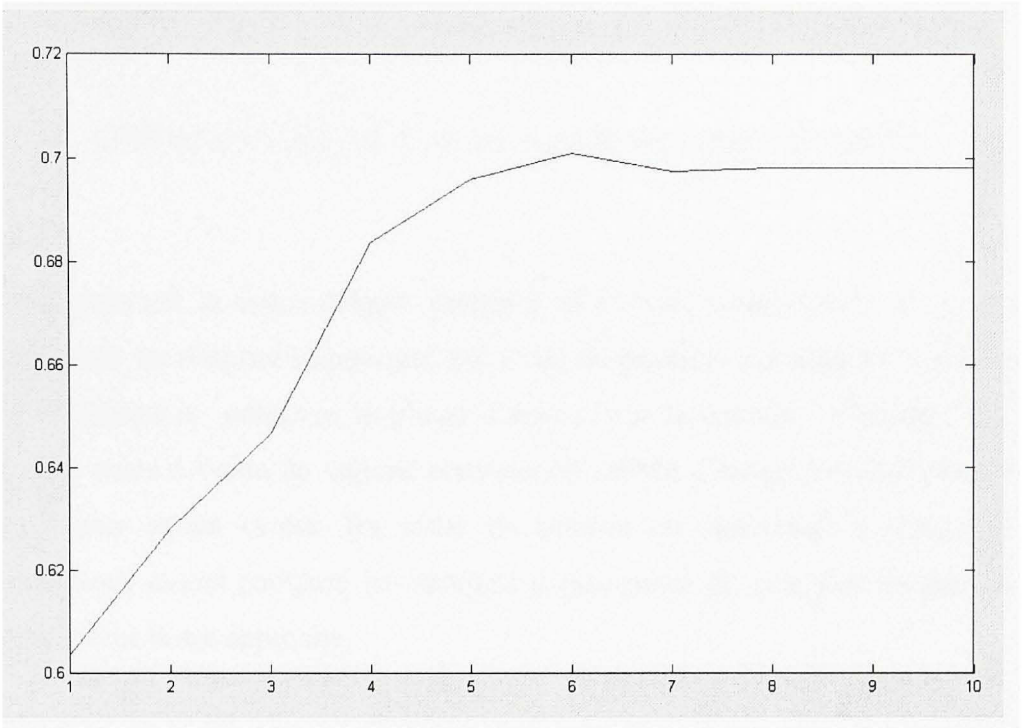

Figure 3.9 ***Allure de la moyenne de pourcentages de points correctement prédits durant les 10 générations.***

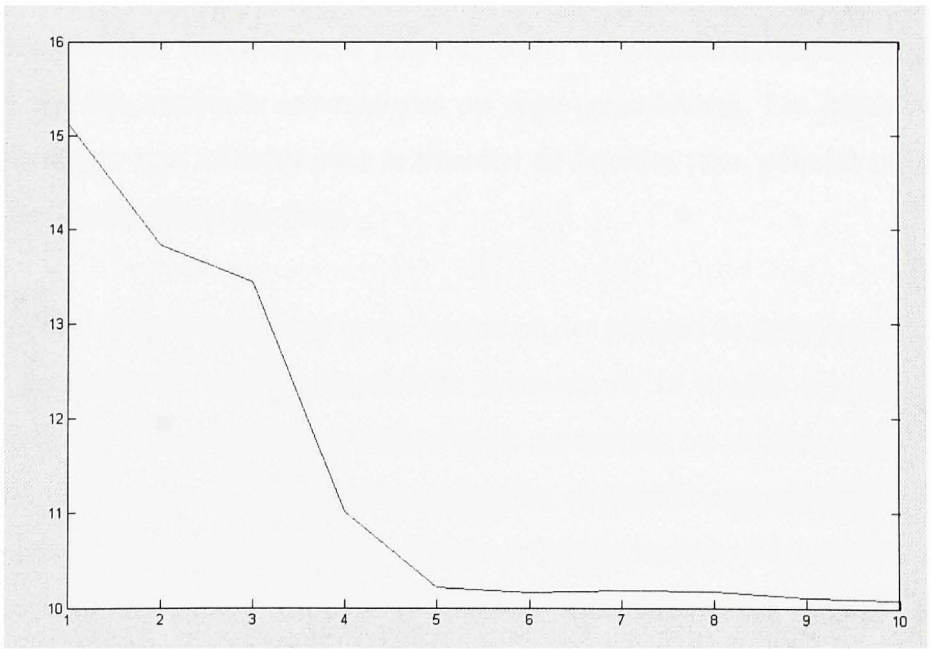

Figure 3.10 ***Allure de la moyenne de MAPE durant les 10 générations.***

CHAPITRE 4

APPLICATIONS AU CAS DE PRODUITS MÉCANIQUES

Après avoir présenté la méthodologie proposée et l'avoir expérimentée sur un exemple, nous allons dans ce chapitre l'appliquer sur 2 cas de produits mécaniques. Le premier est tiré de la littérature et concerne le pliage d'acier. Pour le deuxième exemple, la base de données a été créée à l'aide du logiciel commercial *DFMA (Design for Manufacturing and Assembly)*, nous avons évalué les coûts du produit en changeant à chaque fois ses paramètres. Nous avons comparé les résultats d'estimation de coût fournis par *DFMA* et ceux obtenus avec notre approche.

4.1 Exemple du « *steel pipe bending* »

La base de données de cet exemple est tirée de la littérature Shtub et Versano. (1999). L'estimation du coût de cet exemple est surtout une estimation du coût du procédé même, à savoir le recourbement (ou pliage) de pipes en acier. Le procédé de recourbement (pliage) est effectué par des machines automatisées ou semi-automatisées. Les pipes issues de ce processus de pliage sont utilisées pour le transfert de liquides (eau, pétrole) ou gaz ou bien pour la protection de câbles flexibles.

L'estimation de coûts a été élaborée avec l'approche des réseaux de neurones combinée à un algorithme de recherche pour le nombre de neurones de la couche cachée. La base de données comprend 36 échantillons et le nombre de paramètres ou attributs est de 5 (*nombres de courbures, dimension de l'espace dans lequel les courbures sont effectuées, le diamètre interne des pipes, le diamètre externe des pipes, un certain nombre d'opérations nécessaires selon la longueur des pipes).* La base de données complète de cet exemple se trouve à l'annexe I de ce mémoire.

Le tableau 4.1 présente les coûts réels, estimés par *ANN* (Shtub et Versano, 1999) ainsi que les coûts obtenus par notre approche. Vu que nous ne disposons que d'un nombre limité d'échantillons, nous avons choisi de travailler avec la technique de *cross-validation* pour la séparation entre base d'apprentissage et base de test. Nous avons estimé les coûts avec chacun des modèles I et II que nous avons précédemment présentés.

La première étape de l'approche permet de classer les attributs ou *cost-drivers* par ordre d'importance par rapport au coût par la méthode des *fuzzy-curves*, le résultat donné est : 4, 5, 3 ,2 et 1. Ce qui correspond à :

4 : ***Nombre de points recourbement***

5 : ***Nombre d'opérations requises pour effectuer le recourbement dépendamment de la longueur totale de la pipe***

3 : ***Dimension de l'espace de recourbement***

2 : ***Diamètre extérieur de la pipe***

1 : ***Diamètre intérieur de la pipe***

Ainsi, le paramètre le plus influent sur le coût du pliage des pipes serait le nombre de points de recourbement et le moins influent est le diamètre interne. Cet ordre d'importance nous semble tout à fait logique.

Nous avons calculé le *MAPE* et *le pourcentage de points correctement prédits* donnés par l'approche *ANN* à laquelle nous comparons nos résultats. Le *MAPE* trouvé est de 2890 et *le pourcentage de points correctement prédits* est de *61.1 %* soit 21 points sur 36. Le résultat trouvé par l'approche *ANN* paraissait assez satisfaisant pour les auteurs.

Nous avons amélioré ces 2 mesures de performance par notre approche (pour les deux modèles utilisés) et nous avons obtenus des *MAPE* de 2488 et 2455 (jusqu' à 15% d'amélioration par rapport à 2890), ainsi que des *pourcentages de points correctement prédits* de 69.9 % (25 points sur 36) et 77.7% (28 points sur 36) respectivement aux

modèles I et II, à savoir, en considérant tous les attributs pour l'estimation de coût, ou en n'en considérant que quelque uns.

Le modèle II qui réalise à part l'estimation du coût, une sélection des variables les plus pertinentes, a donné comme résultat *un pourcentage de points bien prédits* de 77.7 % (28 points sur 36) et n'a considéré que les 4 premiers attributs importants trouvés par la méthode des *fuzzy-curves*, soit, *le nombre de points recourbement, le nombre d'opérations requises, la dimension de l'espace de recourbement et le diamètre extérieur de la pipe.*

Tableau 4.1

Tableau des résultats d'estimation de coût pour l'exemple « *pipe bending*»

Échantillon	*Coût Réels*	*Coût estimé par ANN*	*Cout estimé par SVR-GA Modèle I*	*Cout estimé par SVR-GA Modèle II*
1	***239,5***	157	162,0266	180,7021
2	***106,2***	124	117,6388	111,3817
3	***408***	419	405,5496	431,5959
4	***156,6***	150,8	152,3117	151,5799
5	***433,5***	365,4	420,3209	392,4599
6	***109***	109,3	121,8371	123,4611
7	***157,2***	140,9	164,3805	158,0787
8	***101,6***	117,8	95,0925	87,59
9	***170,4***	154	147,9937	146,4929
10	***210***	187,1	226,5032	226,6937
11	***148,6***	163,3	156,0572	149,0238
12	***289,4***	304,5	344,7457	315,3189
13	***166***	97,5	115,0658	103,7916
14	***306***	275	329,639	316,0189
15	***127,2***	156,5	117,0362	112,02
16	***153,8***	159,7	159,9091	156,5122
17	***76,1***	71	118,9324	141,0928
18	***84***	93,9	105,1864	120,6756
19	***170,7***	159,2	156,0482	175,1025
20	***237,8***	183,1	272,0796	299,0516
21	***199,3***	175,6	197,9553	184,5052
22	***425***	358,7	362,4556	363,3533
23	***118,5***	96	128,5605	94,5814
24	***162,5***	170,3	177,3034	170,0221
25	***221,3***	241,8	259,5376	222,2414
26	***182,4***	163,5	151,1885	169,7087

27	***204***	144,4	173,9058	177,9479
28	***145,1***	163,1	171,9776	152,3773
29	***207,8***	276,2	314,1095	288,44
30	***151***	172,2	154,713	162,5781
31	***113,9***	131,9	122,1834	119,5375
32	***112,7***	195,9	173,0103	157,5428
33	***412***	377,1	392,0862	369,0791
34	***170,8***	153,9	161,6903	152,5564
35	***200,1***	175,4	182,6462	175,0945
36	***88***	120,9	73,0936	75,1774
MAPE		***2890***	***2488***	***2455***
% des points correctement prédits		***61.1% 21 points sur 36***	***69.9 % 25 points sur 36***	***77.7 % 28 points sur 36***
Paramètres SVR		-	***Noyau = gaussien Degré = 17 C =10^8 Epsilon = 0.01***	***Noyau = gaussien Degré = 22 C = 10^{11} Epsilon =0.1***
Nb d'attributs considérés		-	***Tous***	***4 parmi 5***

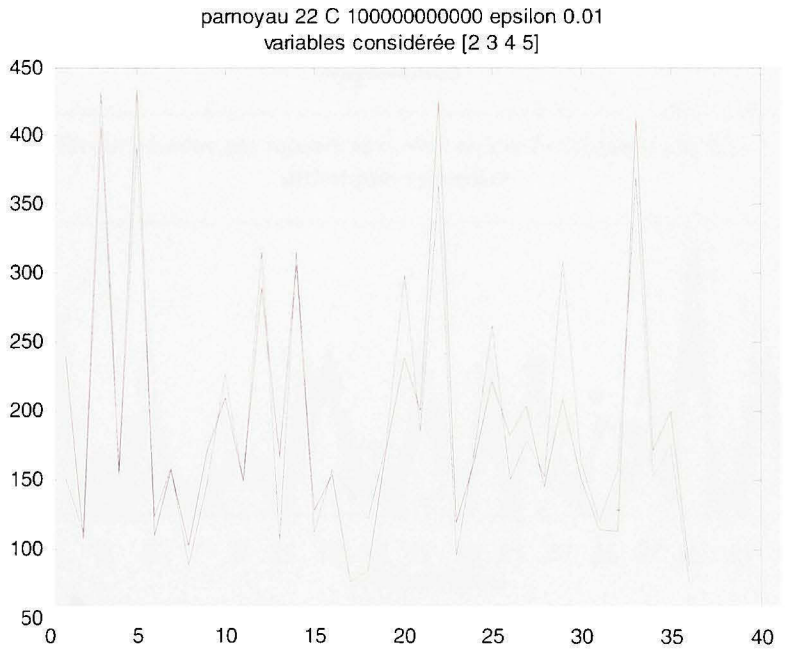

Figure 4.1 ***Courbes des coûts réels versus les coûts estimés de l'exemple «pipe-bending» par SVR-GA modèle II.***

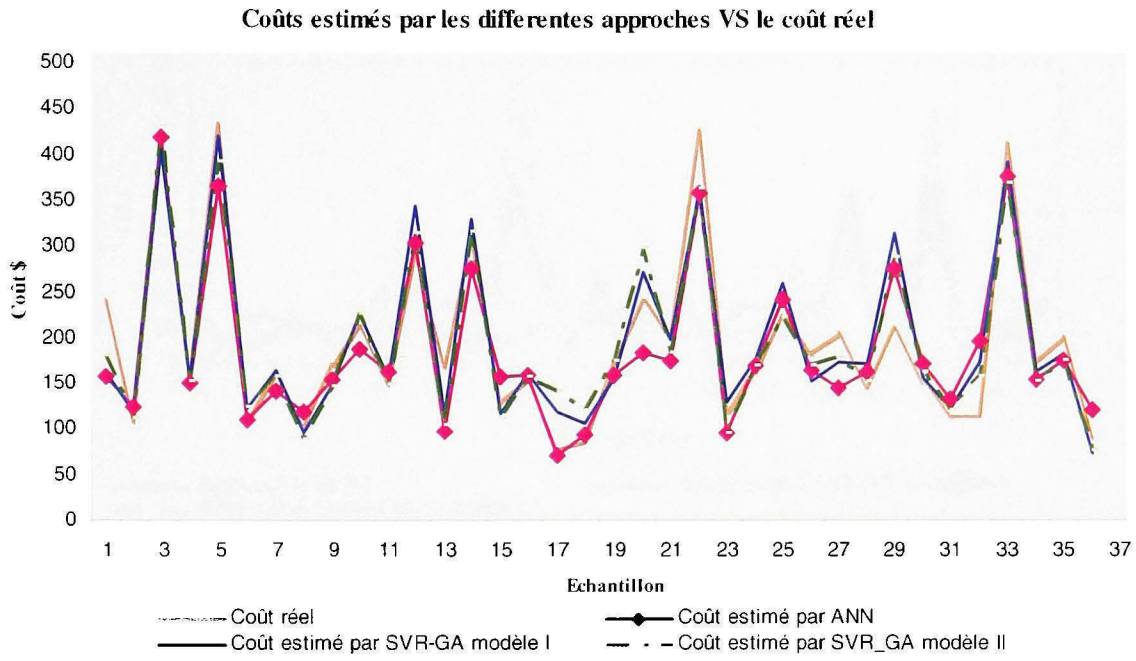

Figure 4.2 *Courbes des coûts estimés de l'exemple «pipe-bending» par les différentes approches*

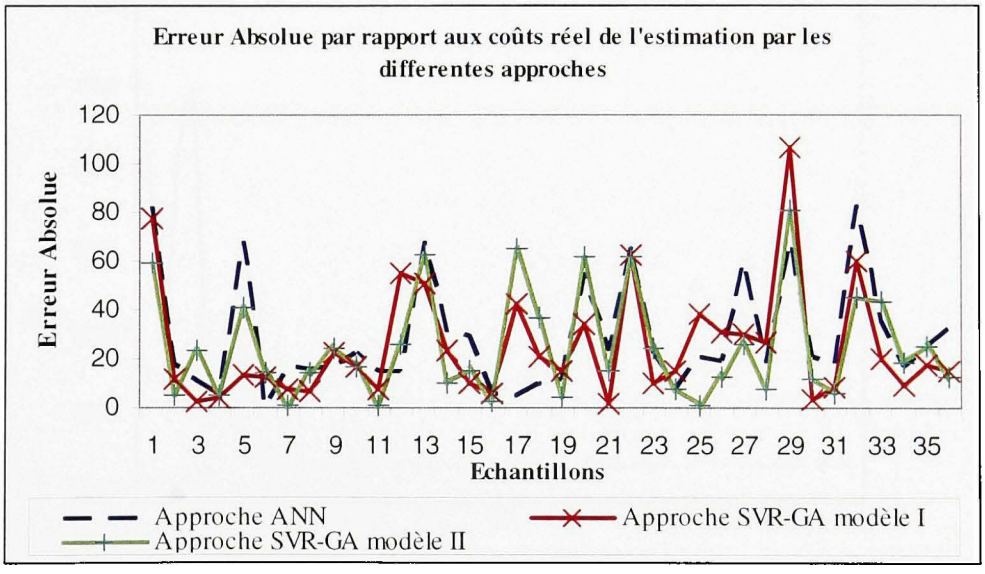

Figure 4.3 *Courbes des Erreurs absolues par rapport aux coûts réels de l'estimation des coûts de l'exemple «pipe-bending» par les différentes approches.*

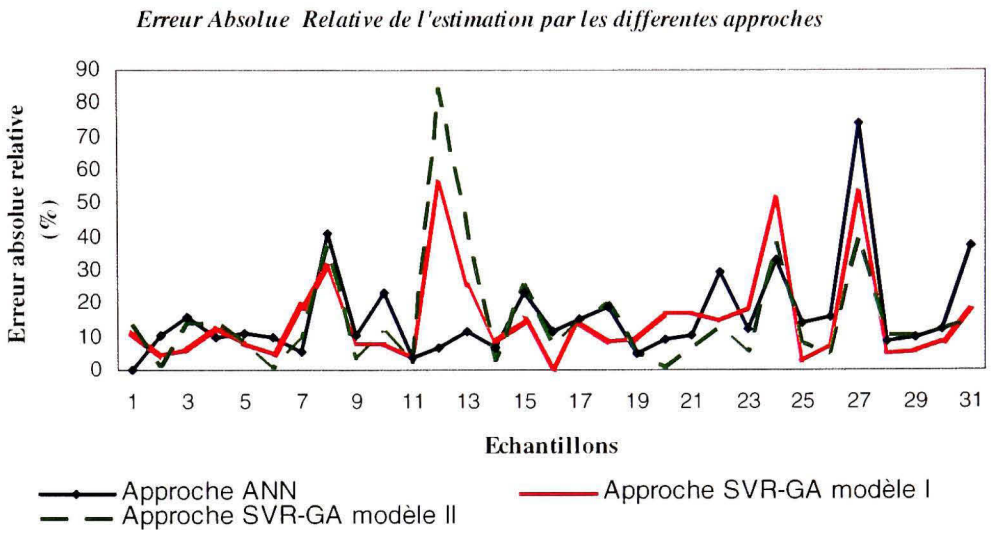

Figure 4.4 *Courbes des Erreurs absolues relatives de l'estimation des coûts de l'exemple «pipe-bending» par les différentes approches.*

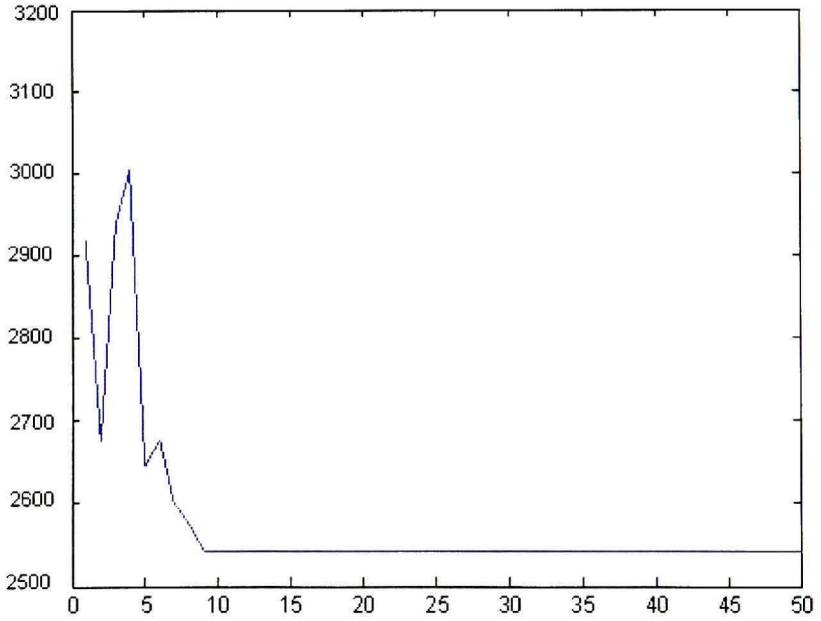

Figure 4.5 *Allure de la moyenne de MAPE durant les générations de l'exemple «pipe-bending».*

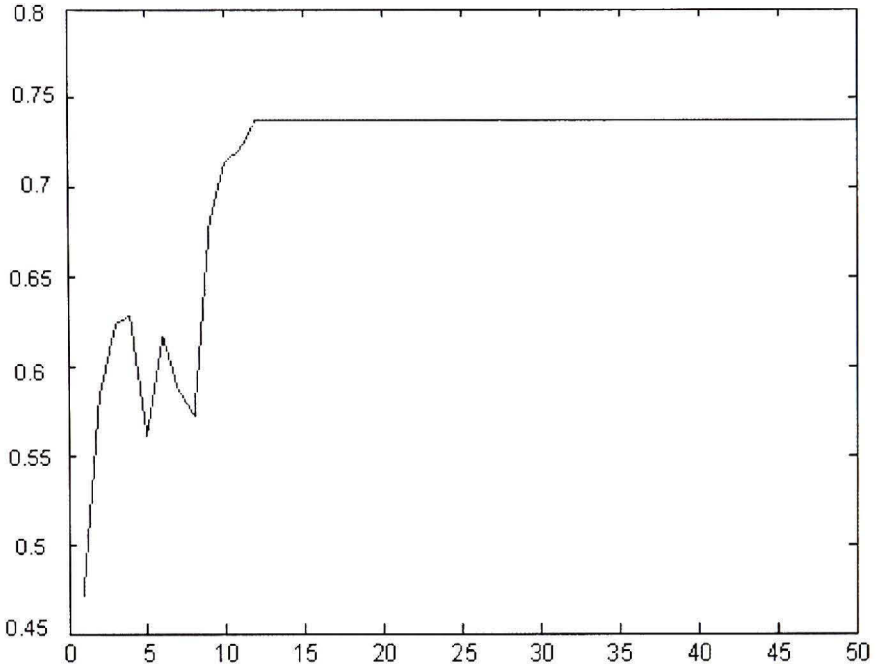

Figure 4.6 ***Allure de la moyenne de pourcentages de points correctement prédits durant les générations de l'exemple «pipe-bending».***

Les figures 4.5 et 4.6 montrent, respectivement, les convergences des 2 mesures de performances *MAPE* et *pourcentage de points correctement prédits.* Nous avons choisi, pour cet exemple d'augmenter le nombre de générations à 50 et de laisser la taille de la population à 100 chromosomes.

Le noyau gaussien est toujours le plus adapté à la modélisation de la fonction du coût final. Pour cet exemple, notre coût final peut s'écrire :

Coût final = $\Im$ *(Nombre de points de recourbement, Nombre d'opérations requises, Dimension de l'espace, et Diamètre extérieur de la pipe).*

4.2 Exemple « *Spindle de DFMA*»

Pour cet exemple, nous avons eu recours au logiciel commercial *Design for Manufacturing & Assembly* ***(DFMA)*** et plus précisément le module ***Dfm Concurrent Costing 2.0*** pour générer la base de données. Nous avons utilisé la pièce ***Spindle*** se trouvant dans la librairie du logiciel à partir de laquelle nous avons généré diverses configurations possibles en changeant les paramètres géométriques de la pièce (attributs).

Comme le montre la figure 4.1, la pièce en question a été fabriquée par le procédé d'usinage à partir d'une pièce brute de forme cylindrique.

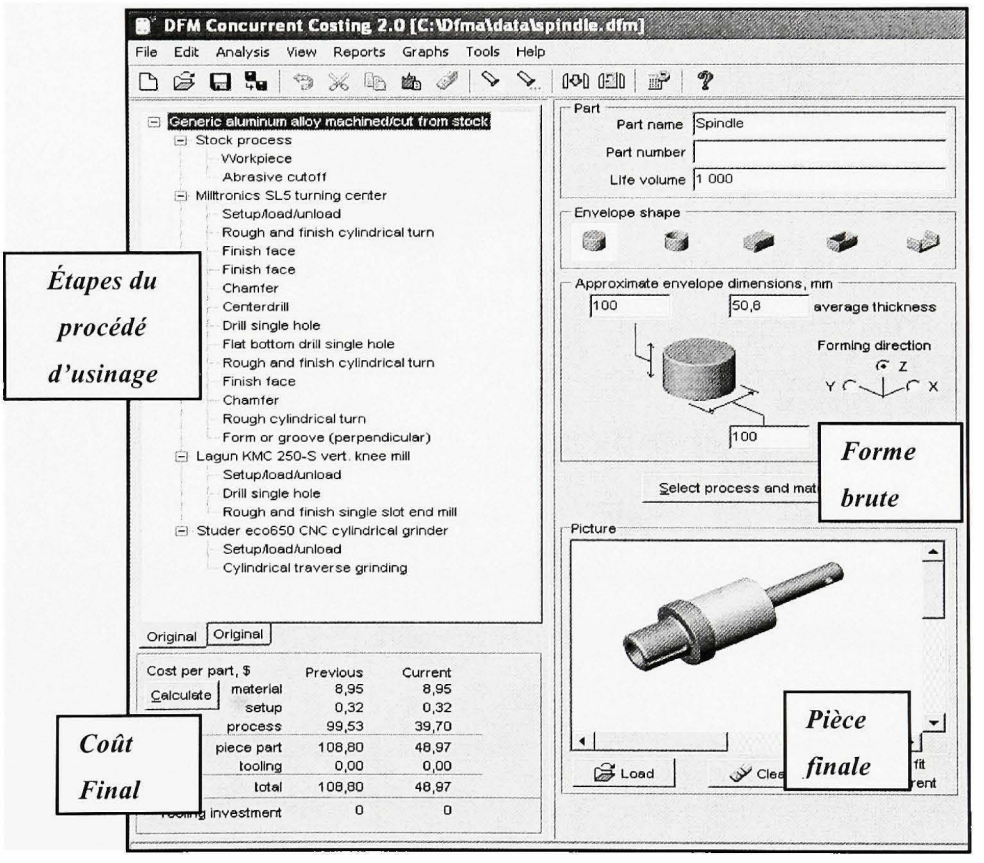

Figure 4.7 ***Exemple de la pièce mécanique «Spindle» traité avec DFM Concurrent Costing.***

Nous avons choisis 19 paramètres dont les différentes dimensions de la pièce (*diamètres des cylindres et des trous, longueur, etc.),* le type de matériau utilisé ainsi que le fini de surface désiré. La figure 4.8 montre la pièce en question avec les différents attributs géométriques. Le fini de surface a été choisi à chaque fois entre les 3 valeurs (*1.6 μm, 6.3 μm et 0.1μm*). Le matériau utilisé a été choisi parmi les 3 matériaux suivants : (*Generic Aluminium Alloy, Generic Stainless Steel ou Generic Titatium Alloy).*

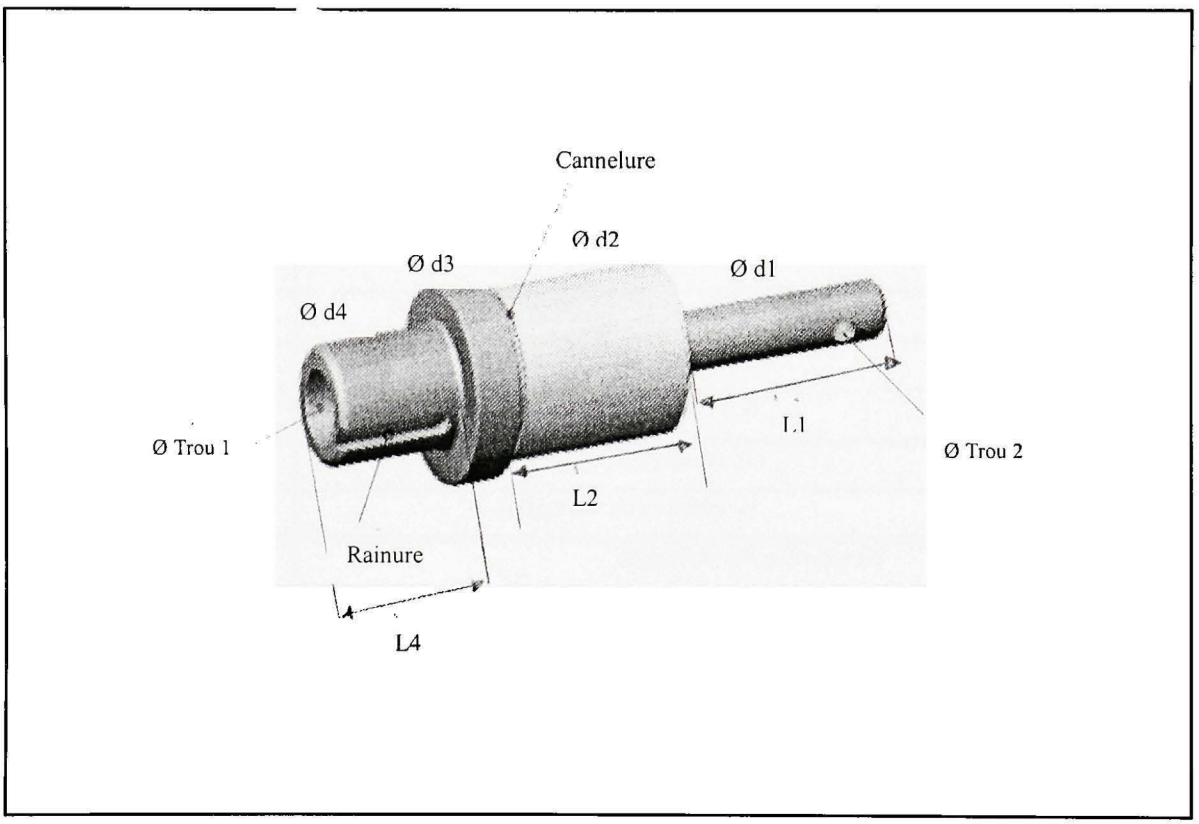

Figure 4.8 ***Pièce de l'exemple «Spindle de DFMA».***

Nous avons varié les dimensions de la pièce allant d'une trentaine de millimètres (3 cm de longueur et de largeur) jusqu'à 500 millimètres pour les 2 dimensions de la pièce brute. Cela donne un éventail assez varié de la possibilité d'avoir à fabriquer des pièces à faibles dimensions (usinage de précision) et d'autres de grandes dimensions. Nous avons également varié les possibilités d'avoir ou non des détails sur la pièce en questions (comme les trous,

la rainure ou la cannelure), en effet, quelques pièces de la base de données en ont, à dimensions variées, et d'autres en sont dépourvues (échantillons 16, 20, 21, 45 et 46).

Voici dans le tableau 4.2, les 19 attributs considérés, agencés selon l'ordre présenté dans la base de données crées pour cet exemple (Voir Annexe II).

Tableau 4.2

Tableau des attributs de coût de la pièce «*Spindle de DFMA*»

Numéro de l'Attribut	***Qualification de l'Attribut***	
1	*Longueur de la pièce brute*	*Enveloppe (Pièce* Brute)
2	*Largeur de la pièce brute*	
3	*Type de matériau*	
4	*Diamètre du cylindre I*	*Cylindre I*
5	*Longueur du cylindre I*	
6	*Diamètre du cylindre II*	*Cylindre II*
7	*Longueur du cylindre II*	
8	*Diamètre du cylindre III*	
9	*Diamètre du cylindre IV*	*Cylindre IV*
10	*Longueur du cylindre IV*	
11	*Diamètre du trou I*	*Trou I*
12	*Profondeur du trou I*	
13	*Diamètre du trou II*	
14	*Longueur de la Rainure*	*Rainure*
15	*Largeur de la Rainure*	
16	*Profondeur de la Rainure*	
17	*Largeur de la Cannelure*	*Cannelure*
18	*Profondeur de la Cannelure*	
19	*Fini de surface*	

La base de données complète se compose de 52 échantillons et se trouve à l'annexe II de ce mémoire. Le tableau 4.3 présente les coûts calculés par *DFMA* ainsi que les coûts estimés par notre approche. Les mesures de performance à savoir le *MAPE* et le *pourcentage des points correctement prédits* sont aussi montrés.

Avant de procéder à l'estimation des coûts par l'approche *SVR-GA*, le premier résultat ressorti par la méthode des *fuzzy-curves* est le classement des attributs du produit par ordre d'importance relativement à leur signification par rapport au coût. Le résultat donné, dans le cas de la pièce *Spindle* est le suivant : $3^{ème}$, $2^{ème}$, $8^{ème}$, $4^{ème}$, $6^{ème}$, $9^{ème}$, $11^{ème}$, $15^{ème}$, $18^{ème}$, $14^{ème}$, $1^{ère}$, $13^{ème}$, $17^{ème}$, $7^{ème}$, $12^{ème}$, $10^{ème}$, $16^{ème}$, $5^{ème}$ et finalement la $19^{ème}$.

Nous remarquons que c'est le type de matériau qui est l'attribut le plus influent sur le coût. Cela concorde tout à fait avec la réalité de l'exemple et du contexte choisis. En effet, il est confirmé qu'en fabrication mécanique, un bon pourcentage du coût final est attribué au coût du matériau. D'un autre coté, l'usinabilité du matériau choisi a un impact direct sur le coût de fabrication de la pièce. Ces deux facteurs font que le type matériau est très significatif par rapport au coût final d'une pièce usinée. Et nous avons la preuve par notre exemple; en effet, le *Titane* (l'un des 3 matériaux choisis) est très dur et très cher, et nous remarquons que les pièces fabriquées avec ce matériau sont celles dont le coût final est exorbitant même si la pièce usinée est de faibles dimensions. (Voir par exemple échantillons 3, 10 ou 47).

Vient ensuite en deuxième position, la largeur de la pièce brute comme second attribut plus influent sur le coût final de la pièce usinée. Nous pouvons très bien expliquer ce résultat par le fait que c'est exactement sur cette dimension que la majorité des tâches d'usinage seront appliquées (enlèvement de la matière), et plus la pièce brute est large, plus nous avons besoin de temps pour usiner les différents cylindres de la pièce, dépendamment aussi de l'usinabilité du matériau. Les diamètres des différents cylindres, quant à eux, prennent les positions suivantes dans l'ordre d'importance par rapport au coût.

Le fini de surface vient en dernière position, ce qui signifie que sa contribution de coût n'est pas très significative dans le coût final, nous l'avons aussi remarqué en utilisant ***Dfm Concurrent Costing***, lors de la création de notre base de données.

Tableau 4.3

Tableau des résultats d'estimation de coût pour l'exemple *«Spindle de DFMA»*

Échantillon	*Coût Calculé par DFMA ($)*	*Cout estimé par SVR-GA Modèle I ($)*	*Cout estimé par SVR-GA Modèle II ($)*
1	***58.15***	65,36	62,35
2	***48.61***	52,41	49,26
3	***416.39***	390,265	421,56
4	***526.05***	496,32	520,369
5	***72.91***	80,06	69,52
6	***205.61***	300,258	214,698
7	***32.09***	37,69	36,58
8	***39.67***	32,379	44,569
9	***62.01***	66,31	62,358
10	***1387.24***	989,99	1258,748
11	***224.09***	212,653	230,25
12	***1066.30***	967,123	1200,02
13	***74.61***	72,356	69,52
14	***99.99***	105,963	101,639
15	***106.79***	121,02	112,036
16	***110.49***	99	116,84
17	***110.76***	100,356	115,99
18	***206.65***	215,36	241,63
19	***39.63***	58,36	38,65
20	***37.62***	56,52	35,25
21	***27.49***	23,15	27,15
22	***34.69***	34,25	39,257
23	***39.71***	44,156	45,12
24	***101.64***	100,69	120,39
25	***201.74***	215,02	189,63
26	***569.81***	620,896	700,69
27	***60.79***	69,05	58,269
28	***81.59***	89,63	87,125
29	***116.52***	118,52	116,87

30	***200.09***	200,789	189,99
31	***1226.74***	1356,489	1400,35
32	***108.01***	112,223	107,56
33	***193.99***	201,69	200,356
34	***48.90***	54,385	49,26
35	***50.33***	47,12	52,4
36	***57.42***	54,23	60,256
37	***424.35***	450,69	431,11
38	***315.52***	345,189	325,48
39	***37.84***	32,15	37,29
40	***50.17***	50,01	40,89
41	***1676.87***	2002,89	1986,223
42	***379.69***	400,58	412,25
43	***44.17***	41,25	45,24
44	***45.58***	41,56	46,23
45	***38.30***	36,05	36,66
46	***27.97***	22,89	29,52
47	***2242.91***	2500,236	2500,998
48	***382.62***	389,54	401,25
49	***296.59***	340,852	302,65
50	***43.62***	48,12	45,65
51	***38.82***	36,12	35,258
52	***584.03***	600,53	568,988
MAPE		***3436,04615***	***2733,12115***
% des points correctement prédits		***80.7%*** ***42 points sur 52***	***90.38%*** ***47 points sur 52***
Paramètres SVR		***Noyau = gaussien*** ***Degré = 9*** ***C =10^9*** ***Epsilon =0.1***	***Noyau = gaussien*** ***Degré = 7.4*** ***C = 10^7*** ***Epsilon = 0.01***
N_b d'attributs considérés		***tous***	***12 Parmi 19***

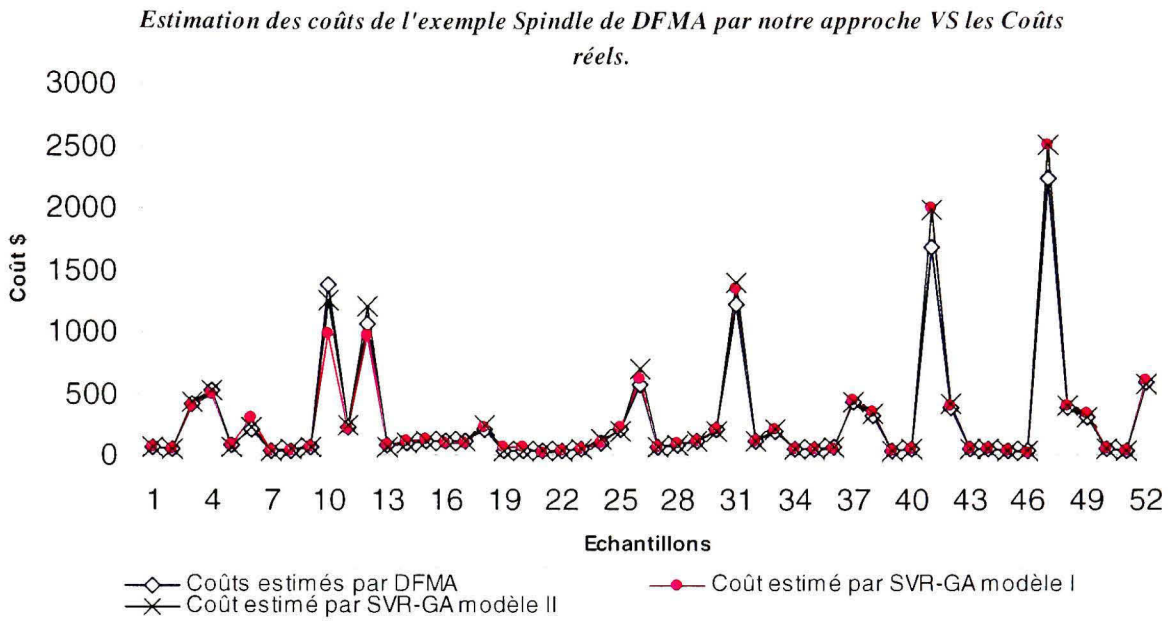

Figure4.9 *Courbes des coûts estimés de l'exemple «Spindle de DFMA» par les deux modèles de l'approche SVR-GA VS le coût réel.*

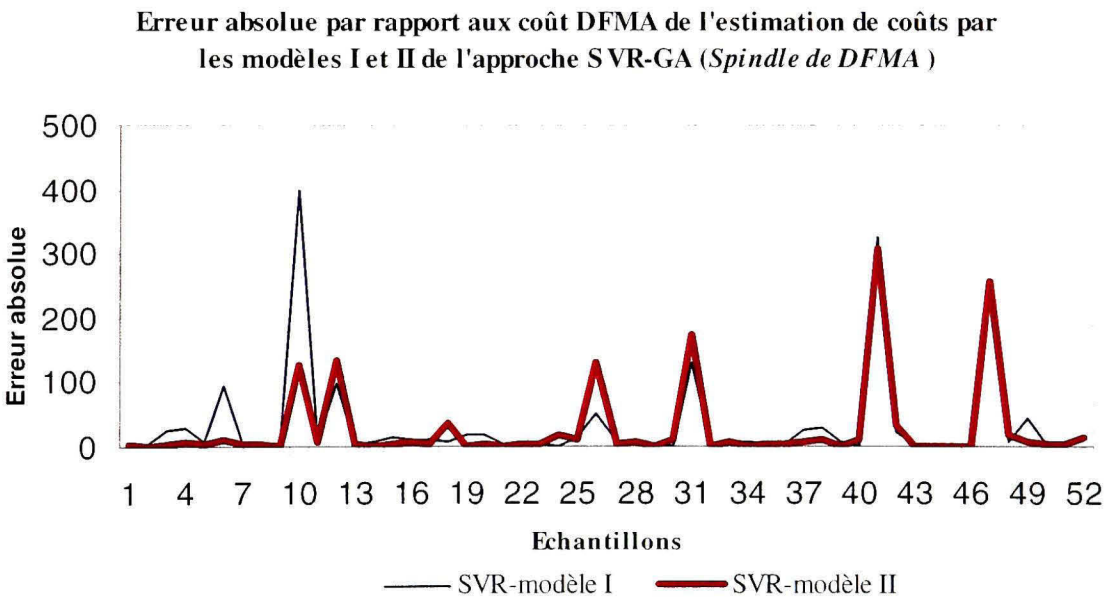

Figure 4.10 *Courbes des Erreurs absolues de l'estimation des coûts de l'exemple «Spindle de DFMA » par les 2 modèles de l'approche SVR-GA.*

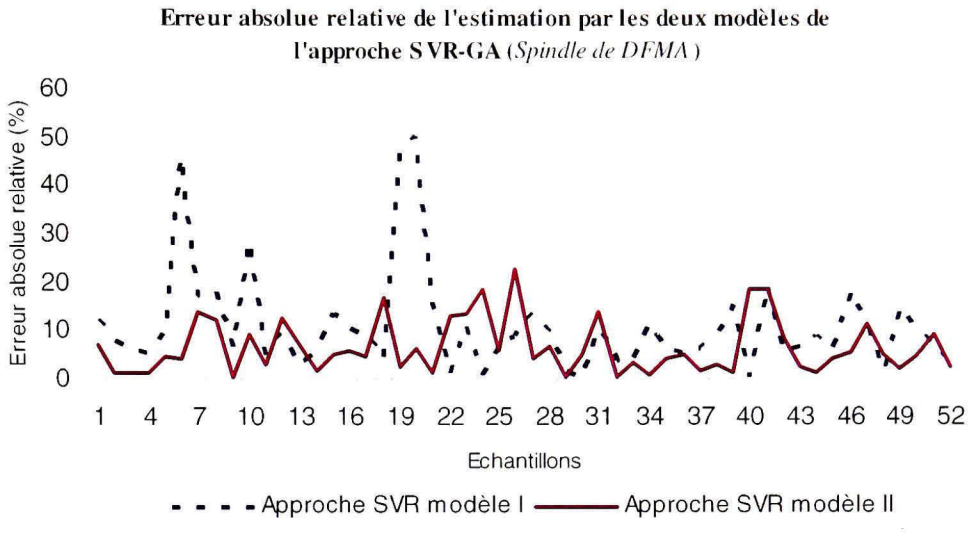

Figure 4.11 *Courbes des Erreurs absolues relatives de l'estimation des coûts de l'exemple «Spindle de DFMA» par les 2 modèles de l'approche SVR-GA.*

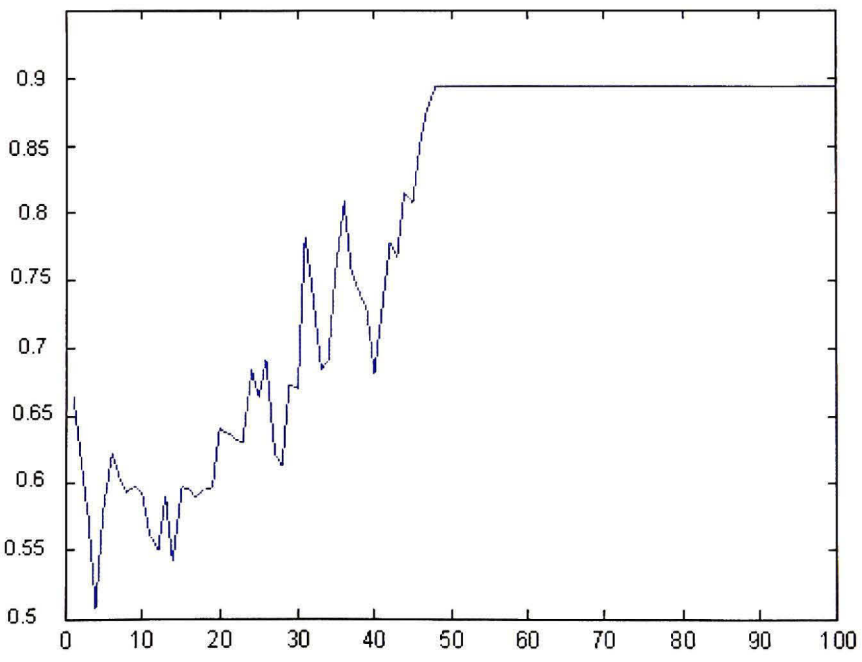

Figure 4.12 *Allure de la moyenne de pourcentages de points correctement prédits durant les générations de l'exemple «Spindle de DFMA».*

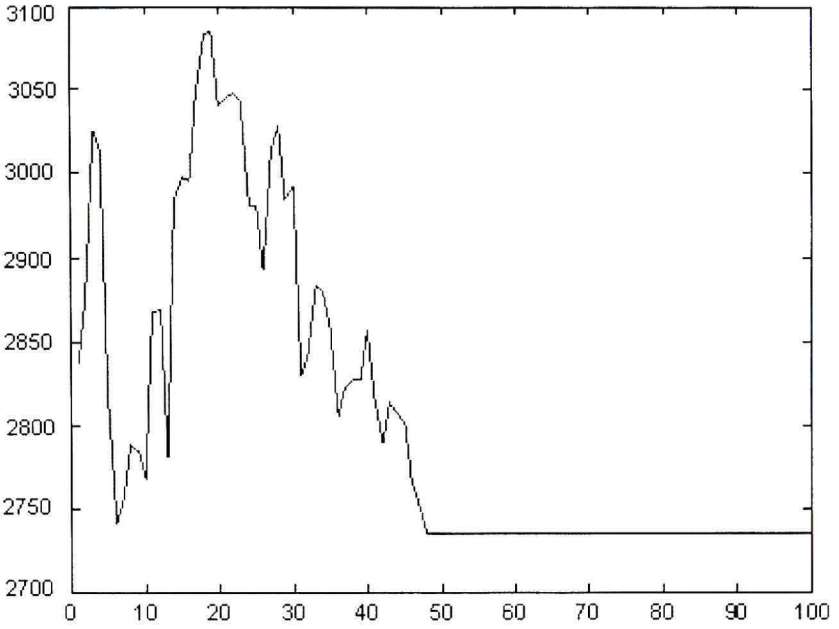

Figure 4.13 ***Allure de la moyenne de MAPE durant les générations de l'exemple «Spindle de DFMA».***

Pour cet exemple, nous avons obtenu des résultats de prédiction très satisfaisants, à savoir *des pourcentages de points bien prédits* de *80.7 %* et 90.38% respectivement selon les modèles I et II de l'approche *SVR-GA*. Ces pourcentages sont meilleurs que l'exemple de *pliage de pipes* présenté dans la première partie de ce chapitre. Nous avons essayé d'expliquer la raison dans la partie *Analyse et discussion* de ce chapitre.

Nous avons appliqué pour ce cas, la technique de *cross-validation* pour la séparation des bases d'apprentissage et de test, puisque cette façon de partager les données est plus pertinente de point de vue résultat.

Vu le nombre assez élevé d'attributs pour cet exemple, nous avons dû augmenter le nombre de générations pour atteindre le meilleur résultat. Nous avons choisi G_{max} égal à 100 ainsi qu'une taille de population de 100 chromosomes. Les figures 4.12 et 4.13 montrent l'allure

des courbes des moyennes des 2 mesures de performances choisies durant les générations. Il est bien clair, que la convergence n'ait été atteinte qu'au bout d'une cinquantaine (50) de générations. La figure 4.11 montre les erreurs absolues relatives de l'estimation de notre coût pour cet exemple, et nous remarquons très bien que la majorité des échantillons ont une erreur relative inferieure à 15%.

Le modèle II de notre approche a ressorti, à posteriori, et, à part la prédiction des coûts, la combinaison des attributs qui a servi à estimer et prédire le coût final avec la moindre erreur *MAPE*, ces attributs sont au nombre de 12 et sont les suivants : $2^{ème}$, $3^{ème}$, $4^{ème}$, $5^{ème}$, $6^{ème}$, $7^{ème}$, $9^{ème}$, $10^{ème}$, $11^{ème}$, $12^{ème}$, $14^{ème}$ et $19^{ème}$. Nous remarquons dans ce cas, que la $5^{ème}$ variable qui était classée comme avant dernière variable influençant sur le coût final par les *fuzzy-curves* fait partie de cette combinaison. Cela prouve encore une fois que l'interaction entre les attributs est aussi significative comme les attributs pris individuellement.

4.3 Analyses et discussions

À la lumière des résultats fournis pour les deux exemples précédemment présentés, nous avons pu faire ressortir quelques constatations et observations. Nous avons remarqué que les résultats fournis pour l'exemple de *pliages des pipes* n'étaient pas aussi satisfaisants que ceux du second exemple (*Spindle de DFMA*). Le résultat donné par l'approche *ANN* (Shtub et al., 1999) ne dépassait pas 61.1% comme *pourcentage de points bien prédits* et le meilleur pourcentage fourni par notre approche était de 77.7% . Nous pouvons expliquer cela par le fait que la base de données n'était peut être pas assez grande ou que les données mêmes n'étaient pas bien collectées. En effet, nous avons remarqué qu'une collecte de données bien réalisée est très importante pour la bonne qualité du résultat fourni.

Nous avons constaté également que la taille de la base de données et les variations dans les valeurs des attributs selon les échantillons présentés, joue un rôle important quant au choix de la taille de la population et du nombre de générations pour atteindre une convergence vers le meilleur résultat. En fait, plus la taille de la base est élevée, plus nous devons inclure des générations dans la recherche de l'optimum.

Nous avons aussi constaté que les valeurs des probabilités P_{cross} et P_{mut} peuvent entrer en considération concernant la vitesse de convergence de l'algorithme génétique, et d'où la vitesse de convergence vers notre meilleur résultat de prédiction.

La normalisation des données a aussi son impact en ce qui à trait à la finesse et la précisions des résultats. Nous avons en fait remarqué qu'un apprentissage avec les données brutes (sans normalisation) n'a pas d'aussi bons résultats qu'un apprentissage réalisé avec des données normalisées.

Le point fort des SVR (*Support Vector Regression*) est de pouvoir approximer et donc modéliser des fonctions ayant plusieurs paramètres en entrée avec simplicité et robustesse, chose que les méthodes classiques de régression ne font pas. En effet, la tâche devient très difficile pour ces méthodes lorsque le nombre de paramètres est élevé. Ces paramètres dans notre cas sont les attributs des produits et la fonction à modéliser est le coût final du produit fabriqué. Cependant, la fonction recherchée (coût final dans notre cas) n'est pas explicitement définie. En effet, les *SVR*, fonctionnent comme des *boites noires* et ne peuvent pas nous fournir la relation exacte existant entre ces différents attributs.

CONCLUSION

À travers de ce travail, nous avons pu répondre à la problématique posée en introduction, à savoir, pouvoir fournir aux concepteurs une approche d'estimation de coût efficace et rapide afin de pouvoir prédire les coûts de fabrication dès la phase de conception. La capacité de l'approche hybride basée sur les *SVR-AG*, proposée dans ce mémoire démontre que nous pouvons fournir pratiquement cet outil. L'avantage de cette approche est de fournir aux concepteurs un outil d'aide à la décision qui est flexible et capable d'estimer le coût de fabrication d'un produit lorsque ce dernier n'est pas encore complètement défini ainsi que ses conditions de fabrications ne sont pas connues.

Tel que démontré au chapitre 3 de ce mémoire, l'approche proposée donne des résultats très satisfaisants, particulièrement en le comparant à l'approche basée sur des réseaux de neurones et cette comparaison nous montre qu'une amélioration jusqu'à 26% a été obtenue avec l'approche proposée. Nous pouvons classer cette approche parmi les méthodes paramétriques d'estimation de coût et les résultats obtenus au chapitre 4 nous montrent que l'application de l'approche proposée est assez simple et flexible.

À part son originalité, du fait que les *SVM* n'ont jamais été appliquées dans le domaine d'estimation de coût, notre approche proposée est de nature universelle, en d'autres termes, nous pourrons l'appliquer à d'autres domaines qui sont autres que l'estimation des coûts des produits. L'implémentation de l'outil dérivé de cette approche proposée va certainement aider les concepteurs qui ont fait face constamment aux problèmes d'optimisation de conception des produits en satisfaisant les contraintes en termes de qualité (de la conception) et de coût (de la fabrication).

En terminant, nous pouvons affirmer en plus qu'à part son originalité, son potentiel et son universalité d'application, l'approche proposée dans le cadre de ce mémoire pourrait être

considérée comme un pas de plus en avant dans le domaine de recherche d'estimation de coûts, particulièrement des coûts de fabrication au stade de la conception d'un produit.

ANNEXE I

Tableau 5.1
Base de données de l'exemple « *pipe-bending*»
(Tiré de Shtub et al., 1999)

Attribut / *Échantillon*	*Diamètre intérieur de la pipe (1)*	*Diamètre extérieur de la pipe (2)*	*Dimension de l'espace de recourbement (3)*	*Nombre de points recourbement (4)*	*Nombre d'opérations requises (5)*
1	16	13	5	3	1
2	16	13	3	2	1
3	18	16	5	4	2
4	16	14	1	1	2
5	10	8	5	5	2
6	20	17	3	3	1
7	18	15	1	1	2
8	11	9	3	3	1
9	20	18	4	2	1
10	12.7	11.3	6	3	2
11	11	9	2	1	2
12	30	28	4	1	2
13	10	9	3	2	1
14	25	23	5	2	2
15	11	8	4	3	1
16	14	12	2	1	2
17	18	16	6	4	2
18	28	26	2	2	1
19	11	8	5	2	2
20	14	12	5	3	1
21	25.4	22.9	6	3	2
22	16	13	4	2	1
23	11	9	4	2	2
24	16	14	2	1	1
25	18	16	3	2	1
26	11	9	3	1	2
27	19.1	16.6	5	3	2
28	15	13	5	4	1
29	22	20	2	1	2
30	15	12	4	1	2
31	15	13	5	3	2
32	12.7	11.3	4	2	2
33	15	13	6	2	1
34	14	12	5	5	1
35	12.7	10.9	2	1	1
36	10	9	2	1	1

Source : Ce tableau est une partie d'un tableau tiré de l'article de journal de Shtub et al., *Estimating the cost of steel pipe bending, a comparison between neural networks and regression analysis* paru dans the International Journal of Production Economics Volume 62 (1999) p.201-207

ANNEXE II

Base de données de l'exemple «*Spindle de DFMA*»

Tableau 5.2

Tableau représentant la base de données complète de l'exemple «*Spindle de DFMA* ».

Attribut / Échantillon	Enveloppe :		Matériau	Cylindre 1 :		Cylindre 2 :		Cylindre 3 :	Cylindre 4 :		Trou1 :		Trou2 :	Rainure			Cannelure		Fini de surface
	Longueur (1)	Largeur (2)	(3)	Ød1 (4)	L1 (5)	Ød2 (6)	L2 (7)	Ød3 (8)	Ød4 (9)	L4 (10)	Øtrou1 (11)	Profondeur (12)	Øtrou2 (13)	Longueur (14)	Largeur (15)	Profondeur (16)	Largeur (17)	Profondeur (18)	(19)
1	120	100	Gen.All.	25	40	70	25	80	60	30	30	35	10	20	10	5	5	2	0.1 µm
2	100	100	Gen. All.	30	50	85	30	95	80	25	50	20	10	20	10	10	10	10	6.3 µm
3	100	100	Gen. Tit. All	30	50	85	30	95	80	25	50	20	10	20	10	10	10	10	6.3 µm
4	200	100	Gen. Tit. All.	30	50	85	30	95	80	25	50	20	10	20	10	10	10	10	6.3 µm
5	200	100	Gen.Stain.St	30	50	85	30	95	80	25	50	20	10	20	10	10	10	10	6.3 µm
6	200	200	Gen.Stain.St	30	80	90	40	180	150	50	120	50	20	10	20	5	10	5	1.6 µm
7	70	50	Gen. Stain.St.	10	20	40	20	50	35	10	30	10	4	10	5	5	5	7	6.3 µm
8	70	50	Gen.All.	10	20	40	20	50	35	10	30	10	4	10	5	5	5	7	6.3 µm
9	300	100	Gen.All.	10	20	40	20	50	35	10	30	10	4	10	5	5	5	7	6.3 µm
10	100	300	Gen. Tit. All	10	20	40	20	50	35	10	30	10	4	10	5	5	5	7	6.3 µm
11	100	300	Gen.Stain.St.	10	20	40	20	50	35	10	30	10	4	10	5	5	5	7	6.3 µm
12	500	500	Gen.Tit.All	100	100	450	100	500	400	200	300	150	20	80	20	10	50	10	6.3 µm
13	300	100	Gen.All.	20	100	70	75	100	85	100	75	80	5	65	30	10	10	20	1.6 µm
14	300	100	Gen.Stain.St.	20	100	70	75	90	75	100	75	80	5	30	30	10	10	20	1.6 µm
15	200	200	Gen.All.	20	100	70	75	90	75	100	75	80	5	30	30	10	10	20	1.6 µm
16	200	200	Gen.All.	30	80	90	20	180	150	80	130	80	0	0	0	0	0	0	6.3 µm
17	200	200	Gen.All.	30	80	90	20	180	150	80	130	80	0	70	20	10	10	10	6.3 µm
18	200	200	Gen.Stain.St	30	80	90	20	180	150	80	130	80	0	70	20	10	10	10	6.3 µm
19	30	30	Gen. All.	5	10	25	7	30	25	10	20	5	0	7	3	2	3	3	0.1 µm
20	30	30	Gen. All.	5	10	25	7	30	25	10	20	5	0	0	0	0	0	0	1.6 µm
21	30	30	Gen.Stain.St.	5	10	25	7	30	25	10	20	5	0	0	0	0	0	0	1.6 µm
22	100	50	Gen.Stain.St.	15	30	40	15	45	35	30	25	10	5	30	10	4	7	5	1.6 µm
23	50	50	Gen.All	10	10	45	10	50	40	20	30	15	2	18	2	1	5	1	1.6 µm
24	200	200	Gen.All	10	10	45	10	50	40	20	30	15	2	18	2	1	5	1	1.6 µm
25	200	200	Gen.Stain.St.	40	65	170	50	200	120	75	80	35	10	30	10	7	10	10	1.6 µm
26	100	150	Gen.Tit.All	30	35	140	15	150	120	40	100	20	8	30	5	5	7	4	6.3 µm
27	100	150	Gen.All.	30	35	140	15	150	120	40	100	20	8	30	5	5	7	4	6.3 µm
28	100	150	Gen.Stain.St.	30	35	140	15	150	120	40	100	20	8	30	5	5	7	4	6.3 µm
29	150	150	Gen.Stain.St.	20	60	135	30	145	130	65	100	50	8	50	10	5	10	5	0.1 µm

30	200	200	Gne.Stain.St.	20	60	135	30	145	130	65	100	50	8	50	10	5	10	5	1.6 μm
31	200	200	Gen.Tit.All	20	60	135	30	180	130	65	100	50	8	50	10	5	10	5	6.3 μm
32	200	200	Gen.All	20	60	135	30	180	130	65	100	50	8	50	10	5	10	5	6.3 μm
33	200	200	Gen.Stain.St.	20	60	135	30	180	130	65	100	50	8	50	10	5	10	5	6.3 μm
34	100	100	Gen. All.	20	30	70	20	100	80	35	70	20	5	30	10	10	15	3	1.6 μm
35	100	100	Gen.All.	10	15	35	10	50	40	17.5	35	10	2.5	15	5	5	7.5	1.5	1.6 μm
36	100	100	Gen.Stain.St.	10	15	35	10	50	40	17.5	35	10	2.5	15	5	5	7.5	1.5	1.6 μm
37	100	100	Gen.Tit.All	10	15	35	10	50	40	17.5	35	10	2.5	15	5	5	7.5	1.5	1.6 μm
38	50	50	Gen.Tit.All	10	15	35	10	50	40	17.5	35	10	2.5	15	5	5	7.5	1.5	1.6 μm
39	50	50	Gen.Stain.St.	10	15	35	10	50	40	17.5	35	10	2.5	15	5	5	7.5	1.5	1.6 μm
40	100	100	Gen. All.	10	15	40	10	95	75	25	60	25	3	10	10	5	8	3	0.1 μm
41	300	200	Gen.Tit.All	10	15	40	10	95	75	25	60	25	3	10	10	5	8	3	1.6 μm
42	300	50	Gen.Tit.All	10	15	30	30	45	30	25	25	25	4	10	5	5	7	5	6.3 μm
43	300	50	Gen. Stain.St.	10	15	30	30	45	30	25	25	25	4	10	5	5	7	5	6.3 μm
44	300	50	Gen.All.	10	15	30	30	45	30	25	25	25	4	10	5	5	7	5	6.3 μm
45	30	30	Gen.All.	8	7	25	5	30	25	15	22	10	0	0	0	0	0	0	0.1 μm
46	30	30	Gen.Stain.st.	8	7	25	5	30	25	15	22	10	0	0	0	0	0	0	0.1 μm
47	100	400	Gen. Tit. All	50	25	350	20	400	300	20	230	10	10	15	20	10	10	10	1.6 μm
48	100	400	Gen.Stain.St.	50	25	350	20	400	300	20	230	10	10	15	20	10	10	10	1.6 μm
49	40	40	Gen.Tit.All	15	15	32	8	38	25	10	20	8	5	8	8	3	6	2	0.1 μm
50	80	80	Gen.All.	30	30	64	16	76	50	20	40	16	10	16	16	6	12	4	6.3 μm
51	40	40	Gen A.ll.	15	15	32	8	38	25	10	20	8	5	8	8	3	6	2	6.3 μm
52	300	300	Gen.Stain.St.	50	75	200	75	280	150	75	120	65	10	50	20	10	15	10	6.3 μm

BIBLIOGRAPHIE

Apgar H.E et Daschbach J.M .1987. «Analysis of Design through Parametric Cost EstimationTechniques ». *Proceeding on International conference on engineering design , p* 759-766.

Arifovica J et Gencay R.2001. «Using genetic algorithms to select architecture of a feedforward artificial neural network . *Physica* , vol.289 , p.574-594.

Asiedu Y. et Gu P. 1998. «Product life cycle cost analysis : state of the art review ». *International Journal of Production Research*, vol. 36, n 4, p. 883-908

Ben-Arieh D. 2000. «*Cost* estimation system for machined parts ». *Intarnational Journal of Production Research*, 2000, Vol.38, No.17, p.4481-4494

Ben-Arieh.D. et Lavelle. J.P. 2000. «Manufacturing Cost Estimation: Application and Methods ». *Engineering Valuation and cot analysis.* Vol.3, p. 43-55

Ben-Arieh D et Qian L. 2002. « Activity-based costing Management for design and Development Stage ». *Intarnational Journal of Production Economics*, vol.83, p. 169-183.

Bode J.1998.« Neural Networks for cost estimation ».*Cost engineering*, vol.40, n 1, p.25-30.

Bode.J.2000. «Neural networks for cost estimation: simulations and pilot application ». *International Journal of production Research*, vol.38, n 6, p.1231-1254.

Bouaziz.Z, Ben Younes.J et Zghal.A .2006. « Cost estimation system of dies manufacturing based on the complex machining features ». *International Journal of Advanced Manufacturing Technology*, Volume 28 p.262–271.

Camargo M, Rabenasolo B, Jolly-desodt A.M et Castelain J.M. 2003. « Application of the cost estimation in the textile supply chain ». *Journal of textile and apparel, Technology and Management*, vol. 3, n° 1.

Canu S., Y. Grandvalet Y., Guigue V., et Rakotomamonjy A. 2005. *SVM and Kernel Methods Matlab Toolbox,* Perception Systèmes et Information, INSA de Rouen, Rouen, France. http://asi.insa-rouen.fr/enseignants/~arakotom/toolbox/index.html

Canu S. 2007. Les machines à noyaux pour l'apprentissage statistique. http://www.techniques-ingenieur.fr/dossier/machines_a_noyaux_pour_l_apprentissage_statistique/TE5255?jsessionid=ED45C616F089551FDAEB6BD952AA047E&resourceName=true

Cavalieri.S, Maccarrone.P et Pinto.R. 2004. «Parametric vs. neural network models for the Estimation of production costs: A case study in the automotive industry ». *Int. J. Production Economics*, vol. 9, p.1165–177.

Cawley G.C. 2006. « Leave-One-Out Cross-Validation Based Model Selection Criteria for Weighted LS-SVMs ». Proceedings of the International Joint Conference on Neural Networks. http://theoval.cmp.uea.ac.uk/~gcc/publications/pdf/ijcnn2006a.pdf

Chan D. S. K. et Lewis W.P. 2000. « The integration of manufacturing and cost information into the engineering design process ». *International Journal of Production research*, vol.38, n. 17, p. 4413-4427.

Chen Y.M et Liu J.J. 1998. «Cost effective design for injection molding ». *Robotics and Computer Integrated Manufacturing* , vol.15 , p.1-21.

Chen.M.Y et Chen.D.F. 2002. «Early cost estimation of strip-steel coiler using BP neural network ». *Procedding of the first international conference on machine learning and cybernetic*, p.1326-1331.

Cherkassky.V et Ma. Y. 2004. «Practical selection of SVM parameters and noise estimation for SVM regression ». Neural Networks, vol.17, p.113-126.

Chin K.S et Wong T.N .1995. « An expert System for Injection Mold Cost Estimation ». *Advances in Polymer Technology*, vol.14, n 4, p. 303-314.

Chougule R.G, et Ravi B. 2005. «Casting cost estimation in an integrated product and process design environment ». *International Journal of Computer Integrated Manufacturing*, vol. 19, n 7, p. 676-688.

Ciwei G., Bompard E., Napoli R et Cheng H , «Price forecast in the competitive electricity market by support vector machine » . *Physica*, vol. 382 , p.98–113.

Cornuéjols A.2002. Une nouvelle méthode d'apprentissage : Les SVM. Séparateurs à vaste marge » ..*Bulletin de L ' AFIA.*, n 51 , p.14-23.

Cornuéjols A .2002 . Apprentissage artificiel : Concepts et algorithmes.

De la Garza. JM et Rouhana. K.G . 1995 . « Neural networks versus Parameter-based applications in Cost estimation » . C*ost Engineering* , vo.37, n 2, p.14-17.

Dean E. 1995. « Parametric Cost deployment ». *Proceeeding of the seventh symposium on Quality function deployment*, p.27-34.

Duverlie. P.1996. *Étude et proposition d'une méthode d'estimation de coût de revient technique appliquée à la production mécanique et basée sur le raisonnement à partir de cas.* Thèse de doctorat, université valencienne.

Duverlie.P et Castelain.J.M .1999. « Cost Estimation During Design Step : Parametric method versus Case based reasoning Method ». *International Journal of Advanced Manufacturing Technology*, vol. 15, n 12, p. 895-906.

Duverlie P., Castelain J.M *et* Farineau T. 1999. « Estimation des coûts en production mécanique » , Techniques de l'ingénieur, traité Genie mécamique.

Emsley.M.W, Lowe.D.J, Duff .A.R Harding.A. et Hickson.A .2002. « Data modelling and the application of a neural network approach to the prediction of total construction costs » *Construction Management and Economics*, vol. 20, n 6, p. 465-472.

Eiben A.E, Hinterding R et Michalewicz Z.1999. Parameter Control in Evolutionary Algorithms. IEEE Transactions on Evolutionary Computation, vol.3, n.2, p.124-141.

Evans. D.K., Lanham. J.D et Marsh. R. 2006. « Cost estimation method selection: matching use requirements and knowledge availability to methods ». http://www.cems.uwe.ac.uk/amrc/seeds/DEvans/Microsoft%20Word%20-%20Paper%20for%20ICEC.pdf

Farineau T., Rabenasolo. B , Castelain J.M, Meyer Y. et Duverlie P. 2001. «Use of Parametric Models in an Economic Evaluation Step During the Design Phase ». *International Journal of Advanced Manufacturing Technology*, vol.17, n 2, p. 79-86.

Gopalakrishnan,B. et Pathak M.A. 1992. «Machine parameter selection and cost estimation techniques. Applications in concurrent engineering ». *Proceeding in Manufacturing Intarnational* , p 249-256.

Hasan M. 2006.«SVM : Machines à Vecteurs de Support ou Séparateurs à Vastes Marges ». http://georges.gardarin.free.fr/Surveys_DM/Survey_SVM.pdf.

Hegazy T. et Ayed A.1998. «Neural network model for parametric cost estimation of highway projects ».*Journal of construction engineering and management*, p.210-218.

Idri A, Mbarki.S et Abran.A .2002. « l'interpretation d'un réseau de neurones en estimation du coût des logiciels », *Actes du 6eme colloque Africain sur la recherche en Informaique (CAR'I 02), 14-17 octobre 2002*, pp.221-228 .

Jeremie M. 2006. Notes de cours en ligne .INRIA. Université de Lille. http://www.grappa.univ-lille3.fr/~mary/cours/SVM.pdf

Jung Jong-Yun. 2002. « Manufacturing cost estimation for machined parts based on manufacturing features ». *Journal of Intelligent manufacturing*, vol.13, p.227-238.

Keerthi S .2002. « Efficient Tuning of SVM Hyperparameters Using Radius/Margin Bound and Iterative Algorithms».*IEEE Trans. on Neural Networks,* vol.13, n 5.

Kim.G.H , Yoon.J.E, An.S.H, Cho .H.H et , Kang.K.I.2004. «Neural network model incorporating a genetic algorithm in estimating construction costs ». *Building and Environment*, vol. 39, p.1333 – 1340.

Kim K.J et Han.I.2000. «Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index ». *Expert Systems with Applications*, vol.19, p. 125–132.

Kim G. H., Seo D.Het Kang K.I . 2005. «Hybrid Models of Neural Networks and Genetic Algorithms for Predicting Preliminary Cost Estimates ». *Journal Of Computing In* Civil Engineering , p.208-211 .

Kim.K.J, Han. I. 2003. « Application of a hybrid genetic algorithm and neural network approach in activity-based costing ». *Expert Systems with Applications*, vol. 24, p. 73–77.

Kim K.J, Han.I. 2003. «Application of a hybrid genetic algorithm and neural Network approach in activity-based costing », *Expert Systems with Applications* , vol. 24 p.73-77

Kim G.H, An S.H, Kang K.I. «Comparison of construction cost estimating models based on regression analysis, neural network, an case-based reasoning ». *Journal of bulding and environment,* vol.39, p.1235-1242.

Layer A, Brinke E.T, Van Houten F, Kals H et Haasis S . (2002. « Recent and future trends In cost estimation » .*International Journal of Computer Integrated Manufacturing*, vol.15, n 6, p 499-510.

Lin.Y et Cunningham.G.A.1994. «A fuzzy Approach to Input Variable Identification». *Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference* , vol.3, p.2031-2036.

Lin.Y et Cunningham.G.A.1995. «A new approach to fuzzy-neural system modeling». *IEEE Transactions on fuzzy systems*, vol.3, n 2 p.190-198.

Lin.Y, Cunningham.G.A et Coggeshall.SV. 1995. «Input variable identification-Fuzzy curves and fuzzy surfaces ». *Fuzzy sets and Systems*, vol.82, p.65-71.

Lin.Y, Cunningham.G.A, Coggeshall.SV et Jones.R.D . 1998. « Nonlinear system input structure identification : Two stage fuzzy curves and surfaces » . *IEEE Transactions on systems, man and cybernetics – Part A : Systems and Humans* , vol.28, n 5, p. 678-684 .

Lin P.T , Su S.F. et Lee T.T . 2005. « Support Vector Regression Performance Analysis And Systematic Parameter Selection » . *Proceedings of International Joint Conference on Neural Networks*, Montreal, *Canada, July 31 - August 4, 2005.*

Martin M. 2002. « On-line Support Vector Machine Regression » . *European conference on machine* learning N°13, *Helsinki , FINLANDE* , vol. 2430, p. 282-294.
Version electronique : http://www.lsi.upc.edu/~mmartin/SVMr-slides.pdf

Mclhenny R, Sethumadhava, T.R, Kwan S.L et Keys L.K .1993. « Integrated system approach to injection molding to facilitate early cost estimation » . *American Society of Mechanical Engineers, Design Engineering Division* ,vol. 52, *Design for Manufacturability* p 105-109

McKim. R.A. 1993. « Neural network applications to cost engineering » . *Cost Engineering*, vol. 35, n 7, p.31-35 .

Moussafir J.O. 2005 . www.ceremade.dauphine.fr/~msfr/ens/kernel.pdf.

Niazi.A, Dai J.S, Balabani S. et Seneviratne L . 2006. «Product Cost Estimation:Technique Classification and Methodology Review » . *Journal of Manufacturing Science and Engineering* vol. 128, p.563-575.

Perry.N, Mauchand.M et Bernard.A. 2003 . «Modèles de coûts en fonderie sable : les limites d'une approche générique ». 3ème Colloque Int. en Conception et Production. http://www.supmeca.fr/cpi2005/FR/Articles/044.pdf.

Pai P.F et , Hong W.C . 2005. «Support vector machines with simulated annealing Algorithm in electricity load forecasting » . *Energy Conversion and Management* , vol.46 p. 2669–2688.

Patwardhan H. Ramani, K . 2004. « Manufacturing feature based dynamic cost estimation For design » . *Proceedings of the ASME Design Engineering Technical Conference*, vol. 3, *Proc. of the ASME Des. Eng. Tech. Conf. and Comput. and Inf. in Eng. Conf. 2004: vol. 3: 16th Int. Conf. on Des. Theory and Methodol.: 2nd Symp. on Int. Issues in Eng. Des.: Integr. of Materials Micro* p 945-953.

Pearce .1989. « A statistical / heuristic approach to estimating Mold costs ». *Annual Technical Conference - Society of Plastics Engineers*, p 364-366.

Roy R., Kelvesjo S, Forsberg S. et Rush C. 2001. «Quantitative and qualitative cost Estimating for Engineering Design ». *Journal of Engineering Design*, vol. 12, n 2, p.147-62

Rehman S et Guenov M.D. 1998. «A Methodology For Modelling Manufacturing Costs at Conceptual Design ». *Computers Ind. Engng, v*ol. 35, n 3-4, p 623-626.

Seo.K.K, Ahn.B.J. 2006, «A learning algorithm based estimation method for maintenance cost of product concepts ». *Computers & Industrial Engineering* , vol.50 p.66–75

Stockton. D et Wang.Q. 2004. «Developing cost models by advanced modeling technology» *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, v 218, n 2, February, 2004, p 213-224

Seo.K.K, ParkJ.H,, Jang.D.S et Wallace.D. 2002. «Approximate Estimation of the Product Life Cycle Cost Using Artificial Neural Networks in Conceptual Design ». *International Journal of Advanced Manufacturing Technology* , vol.19, p.461-471.

Seo.K.K et Park.J.H. 2004. «Incorporating life-cycle cost into early product development ». *Proceedings of the Institution of Mechanical Engineers, Part B (Journal of Engineering Manufacture)*, v 218, n B9, Sept. 2004, p 1059-66.

Seo K.K. et Ahn B.J 2006. « A learning algorithm based estimation method for maintenance cost of product concepts ».*Computers & Industrial Engineering*, vol. 50,p. 66–75.

Seo K.K, Park J.H , Jang D.S et Wallace D. 2002. «Approximate Estimation of the Product Life Cycle Cost Using Artificial Neural Networks in Conceptual Design ». *International Journal of Advanced Manufacturing Technology,* vol. 19 p. 461–471.

Seo.K . 2006. «A methodology for estimating the product life cycle cost using a hybrid GA and ANN model » . *Lecture Notes in Computer Science* v 4131 LNCS - I, *Artificial Neural Networks, ICANN 2006 - 16th International Conference, Proceedings*, 2006, p 386-395.

Schreve .1997. « *Cost Estimating Welded Assemblies Produced in Batches* » Thesis presented in partial fulfillment of the requirements for the degree Master of Engineering at the University of Stellenbosch.

Schreve K, Schuster et Basson A.H. 1999. «Manufacturing cost estimation during design of fabricated parts » . Proceedings of the Institution of Mechanical Engineers; Part B; Journal of engineering manufacture , vol:213 No:7 Page:731.

Shawe-taylor J. et Cristianini N. 2000. «*Support Vector Machines and other kernel-based Learning methods* ». Cambridge University Press

Shehab.E et Abdallah H.2002. «An Intelligent Knowledge-Based System for Product Cost Modelling ». *International Journal of Advanced Manufacturing Technology*, vol. 19, p.49-65.

Shtub.A et Zimerman.Y . 1993. « A neural-network-based approach for estimating the cost of Assembly systems ». *International Journal of Production Economics,* vol.32, p.189-207.

Shtub A. et Versano R. 1999 . « Estimating the cost of steel pipe bending, a comparison between neural networks and regression analysis». *International journal of production economics*, vol. 62, p.201-207

Smith. A. 1993. « Using artificial neural networks for estimation during cost analysis ». *Proceedings of the Industrial Engineering Research Conference* . p 102-106

Smith. A. et Mason A.K. 1997. «Cost Estimation Predictive Modeling: Regression versus Neural Network ». *Engineering Economist*, v 42, n 2, p 137-161 .

Smola A.J et Scholkopf.B.2004. « A tutorial on support vector regression ». *Statistics and Computing* , vol.14, p.199-222.

Tomovic M.M. 2002. «Methodology for engineering cost and lead-time estimation in metal casting applications ». *American Society of Mechanical Engineers, Innovations and Applied Research in Mechanical Engineering Technology,* p 27-33.

Vapnik V. 1999 . « Tree remarks on the support vector method of function estimation». *Advances in Kernel Methods support Vector learning*, p.25-42.

Vojislav.K. 2001. *Learning and soft computing- Support vector machines, Neural Networks and fuzzy Logic models* . The MIT Press.

Watson.P, Curranl R., Murphy A. et Cowan S. 2006. « Cost Estimation of Machined Parts within an Aerospace Supply Chain ». *Concurrent Engineering* , vol.14, n 1, p.14; 17

Weustink I.F, Brinke t.E, Streppel A.H et Kals H.J.J.2000. « A generic framework for cost estimation and cost control in product design » , *Journal of materiel Processing Technology*, vol.103, p.141-148.

Wang.Q et Stockson. D.J. 2002. «Artificial neural networks and their applications in cost model development process » . *Intelligent Engineering Systems Through Artificial Neural Networks*, vol. 12, p.1019-1024.

Wang.Q et Stockton.D.2001. « Cost model development using artificial neural networks ». *Aircraft Engineering and Aerospace Technology*, vol. 73,n 6 p.563-541.

Wierda L.S. 1988. « Product cost-estimation by the designer ». *Engineering Costs and Production Economics*, vol. 13, n 3,p. 189-198.

Welling M. « Support Vector Regression ». http://www.ics.uci.edu/~welling/classnotes/papers_class/SVregression.pdf

Yang C et Lin T.S 1997. « Developing an Integrated Framework for Feature-Based Early Manufacturing Cost Estimation ». *International Journal of Advanced Manufacturing Technology*, vol.13 p.618-629.

Zhang.Y.F et Fuh.Y.H. 1998. «A neural network approach for early cost estimation of packaging products ». Computers ind. Engineering, vol .34 , n 2 ,p.433-450.

Zhang. Y.F., Fuh. J.Y.H. et Chan. W.T .1996 . « Feature-based cost estimation for packaging products using neural networks » .*Computers in Industry*, vol. 32, p.95-113.