



HAL
open science

Surveillance des systèmes automatiques et systèmes embranchés

Mourad Benkaci

► **To cite this version:**

Mourad Benkaci. Surveillance des systèmes automatiques et systèmes embranchés. Automatique. Université Paul Sabatier - Toulouse III, 2011. Français. NNT: . tel-00647456

HAL Id: tel-00647456

<https://theses.hal.science/tel-00647456>

Submitted on 2 Dec 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Université
de Toulouse

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :
Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Discipline ou spécialité :
Systèmes Embarqués

Présentée et soutenue par :
Mourad BENKACI

le : jeudi 24 février 2011

Titre :

Surveillance des Systèmes Mécatronique d'Automobile par des Méthodes
d'Apprentissage

Ecole doctorale :
Systèmes (EDSYS)

Unité de recherche :
LAAS-CNRS (DISCO)

Directeur(s) de Thèse :
Andrei DONCESCU et Bruno JAMMES

Rapporteurs :
Houcine CHAFOUK
Pierre SIEGEL

Autre(s) membre(s) du jury

Gilles ROUX
Jacky DESACHY
Ghaleb HOBLOS

Remerciement

Mes travaux de thèse ont été réalisés au Laboratoire d'Analyse et d'Architecture des Systèmes du Centre National de la Recherche Scientifique (LAAS-CNRS) au sein du groupe Diagnostic, Supervision et Conduite qualitatifs (DISCO). Je tiens tout d'abord à remercier les directeurs successifs du LAAS, Messieurs Raja CHATILLA et Jean-Louis SANCHEZ de m'avoir accueilli au LAAS-CNRS.

Je tiens à remercier Madame Louise TRAVE-MASSUYES, responsable du groupe DISCO, de m'avoir accueilli dans son groupe de recherche pendant ces années de thèse.

J'exprime toute ma gratitude à mon Directeur de thèse, Monsieur Andrei DONCESCU, qui sur le plan personnel m'a permis d'établir une relation de confiance, partage et amitié et qui, du côté professionnel, m'a encadré avec patience, rigueur et compétence. Je le remercie pour sa disponibilité constante et efficace, ses critiques, son soutien et ses encouragements.

Ma reconnaissance va aussi à Monsieur Bruno JAMMES pour son encadrement tout au long de ce travail ainsi que pour ses corrections et ses remarques pertinentes.

Je suis très reconnaissant envers Monsieur Houcine CHAFOUK, Professeur à l'ESIGELEC de Rouen, et Monsieur Pierre SIEGEL, Professeur à l'Université d'Aix-Marseille, d'avoir accepté d'étudier mes travaux et d'en être les rapporteurs ainsi que pour l'intérêt et l'attention qu'ils ont accordés à cette étude.

J'adresse mes plus vifs remerciements aux membres du jury pour leurs conseils et leurs remarques dans l'évaluation de ce travail :

- Gilles ROUX, Professeur des universités (UPS), LAAS-CNRS
- Ghaleb HOBLOS, Enseignant-chercheur-HDR, ESIGELEC Rouen
- Jacky DESACHY, Professeur des universités, Université des Antilles et de la Guyane

Je remercie également Monsieur Charles-Henri GARIH, Directeur R&D, DELPHI FRANCE d'avoir accepté de juger mes travaux de thèse.

Je tiens également à remercier Madame Nathalie DONCESCU pour son aide précieuse dans la correction de ce manuscrit.

A présent, je remercie chaleureusement, le professeur Abd-El-Kader SAHRAOUI pour ses conseils et orientations, le professeur Mohamed CHERIET de m'avoir accueilli dans le laboratoire LIVIA-Montréal dans le cadre de la mobilité internationale ainsi que

Monsieur Hamid DEMMOU Maître de conférences à l'université Paul SABATIER pour ses conseils et ses encouragements.

Je ne saurais terminer sans remercier très profondément mes parents, mes sœurs et mon frère pour m'avoir continuellement soutenu, constamment encouragé dans mes démarches et indéfiniment témoigné affection et tendresse durant toutes ces années, bien plus encore chaque jour. Naturellement, mes respectueux remerciements s'adressent également à mes chers grands parents ainsi qu'à l'ensemble de ma famille.

Mourad BENKACI

Table des matières

Introduction générale.....	7
1 Surveillance des systèmes	12
1.1 Systèmes mécatroniques.....	13
1.1.1 Définition.....	14
1.1.2 La mécatronique et l'industrie automobile.....	15
1.2 Les approches du diagnostic.....	16
1.2.1 Les types de données enregistrées.....	16
1.2.2 Les différents types de pannes par composant	16
1.2.3 Les différentes pannes selon la dynamique.....	17
1.2.4 Approches à base de modèle	18
1.2.5 Approches à base de connaissances	21
1.2.6 Approches se basant sur la reconnaissance de forme.....	23
Phase d'analyse	25
Phase de choix du système de détection.....	25
Phase d'exploitation	25
1.3 La pertinence :	26
1.3.1 La pertinence et les systèmes cognitifs	26
1.3.2 La pertinence de variables en apprentissage de données.....	28
1.4 La sélection des variables pertinentes pour la classification	29
1.4.1 Introduction	29
1.4.2 Les différentes étapes de sélection des variables.....	30
1.4.3 Catégorisation des méthodes de sélection des variables	32
1.4.4 Détection et isolation des défauts.....	43
1.4.5 Les techniques FDI se basant sur les données en littérature.....	45
1.5 Apprentissage automatique	47
1.5.1 Introduction	47
1.5.2 Les types d'apprentissage.....	47
1.6 Conclusion.....	50
1.7 Références	51
2 Détection et isolation de défauts par projection : Notion du désaccord :	59
2.1 Introduction	59
2.2 Problème traité et objectifs de la thèse.....	60

2.3	Conflit/désaccord en fusion d'information.....	63
2.3.1	Liens entre fusion d'information et FDI.....	63
2.3.2	Théorie des fonctions de croyance	65
2.3.3	Conflit de Dempster-Shafer et fusion de sources d'information	67
2.4	Nouveau critère : Conflit/désaccord binaire entre variables	67
2.4.1	Introduction	67
2.4.2	Formalisation.....	68
2.4.3	Illustration	70
2.5	Exemple d'application en 4 dimensions : détection de fautes en automobile	71
2.5.1	Application.....	71
2.5.2	Partitionnement à pas fixe	72
2.5.3	Calcul du conflit.....	72
2.6	Techniques de partitionnement en hyper-rectangles	77
2.6.1	La famille R-Tree	77
2.6.2	Clustering Fuzzy Min-Max (FMMC).....	78
2.6.3	Nested Generalized Exemplar (NGE)	81
2.6.4	Comparaison.....	83
2.7	Conclusion et choix de Fuzzy-ARTMAP.....	84
2.8	Références	85
3	FDI à base de Fuzzy-ARTMAP	89
3.1	Introduction	89
3.1.1	Neurone	89
3.1.2	Les réseaux de neurones.....	91
3.2	La théorie de résonance adaptative.....	92
3.2.1	Théorie de la résonance adaptative.....	93
3.3	Les modèles ARTMAP existants	97
3.3.1	L'auto-organisation de la prédiction de cartes	97
3.3.2	Maximisation de la généralisation et minimisation de l'erreur de prédiction	97
3.3.3	L'auto-organisation par un système expert	98
3.4	Partitionnement par le réseau Fuzzy-ARTMAP.....	99
3.4.1	L'activité des couches fuzzy-ARTMAP.....	99
3.4.2	Algorithme Fuzzy-ARTMAP.....	100
3.4.3	Fuzzy-ARTMAP et ordre de passage de données	104
3.4.4	K-means avec fuzzy-ARTMAP	105

3.5	Application en détection et isolation de fautes : application automobile	107
3.5.1	Description de l'application	107
3.5.2	Sélection des variables pertinentes pour FDI	108
3.5.3	Résultats de l'approche FDI Fuzzy-ARTMAP sur l'application automobile	109
3.6	Conclusion.....	114
3.7	Références	115
4	Complexité de Kolmogorov pour la détection et l'isolation de défauts	118
4.1	Introduction	118
4.2	Mesure de la complexité.....	119
4.2.1	Représentation des séquences.....	119
4.3	La Complexité de Kolmogorov	120
4.3.1	Introduction	120
4.3.2	Complexité de Kolmogorov et Machines du Turing.....	122
4.3.3	Estimation de la Complexité de Kolmogorov par la compression	127
4.3.4	La Distance de Compression	137
4.4	Applications.....	139
4.4.1	Application 1 : Capacité discriminatoire de l'approche	139
4.4.2	Application 2 : Capacité de détection de l'approche.....	142
4.5	Conclusion.....	145
4.6	Références	146
	Conclusion générale	149
	Contributions de la thèse	151

Liste des figures

Fig. 1.1	Système mécatronique	14
Fig. 1.2	Mécatronique en automobile.....	16
Fig. 1.3	Evolution temporelle des défauts.....	18
Fig. 1.4	FDI par redondance.....	20
Fig. 1.5	Génération de résidus pour la détection des pannes.....	20
Fig. 1.6	Système expert	22
Fig. 1.7	Principe de la reconnaissance des formes	24
Fig. 1.8	Phase d'analyse : constitution du vecteur forme.....	25
Fig. 1.9	Stratégies de sélection de variables.....	32
Fig. 1.10	Catégorisation de Jain et Zongker.....	33
Fig. 1.11	La procédure filtre de sélection de variables.....	34
Fig. 1.12	La procédure Wrapper de sélection de variables	41
Fig. 1.13	Modèle d'un système de détection de défauts.....	44

Fig. 2.1 Système FDI proposé.....	61
Fig. 2.2 Exemple d'illustration.....	70
Fig. 2.3 Conflit du jeu de variable « position du papillon – valeur de couple transmise » : Conflit global = 0.0063.....	73
Fig. 2.4 Conflit du jeu de variable « position du papillon – durée d'injection ».....	73
Fig. 2.5 Conflit du jeu de variable « valeur de couple transmise – durée d'injection » : Conflit global = 0.2329.....	74
Fig. 2.6 Conflit du jeu de variable « durée d'injection – température de la sonde d'eau » : Conflit global = 0.2585.....	74
Fig. 2.7 Plan N°1 : « position de papillon – valeur de couple transmise ».....	74
Fig. 2.8 Plan N°2 : « position de papillon – durée d'injection ».....	75
Fig. 2.9 Plan N°3 : « valeur de couple transmise – durée d'injection ».....	75
Fig. 2.10 Plan N°4 : « durée d'injection – température de la sonde d'eau ».....	75
Fig. 2.11 Algorithme R-Tree.....	78
Fig. 2.12 Un hyper-rectangle Min-Max $R_j = \{V_j, W_j\} \in I^3$	79
Fig. 2.13 Architecture du réseau FMMC.....	80
Fig. 3.1 Un neurone réalisant une fonction f , possédant X_i entrée et une sortie Y	90
Fig. 3.2 Neurone de base de Perceptrons Multi-Couches.....	90
Fig. 3.3 Neurone RBF à non-linéarité gaussienne.....	91
Fig. 3.4 Architecture d'un réseau de neurones non bouclé complètement connecté à n entrées, N_s-1 neurones cachés et 1 neurone de sortie. W : connexions entrées – neurones, Z : connexions entre neurones.	92
Fig. 3.5 Architecture du Module ART, N : dimension de vecteur d'entrées ($i = 1:N$) et M : nombre de catégories disponibles ($j = 1:M$).....	94
Fig. 3.6 a- Entrées/sorties de la couche F_1 . b- Entrées/sorties de la couche F_2	95
Fig. 3.7 Le diagramme ARTMAP (Carpenter and Al, 1991).....	98
Fig. 3.8 L'architecture de réseau Fuzzy-ARTMAP (Carpenter et al. 1992).....	100
Fig. 3.9 Comparaison ART1 vs Fuzzy-ART (Carpenter et al. 1992).....	103
Fig. 3.10 Algorithme K-moyennes.....	105
Fig. 3.11 Combinaison fuzzy-ARTMAP K-moyennes.....	106
Fig. 3.12 Comparaison de l'erreur de classification (Fuzzy-ARTMAP vs Fuzzy-ARTMAP+K-means).....	108
Fig. 3.13 Méthode de sélection des variables.....	109
Fig. 3.14 Variables corrélées avec la panne 1.....	110
Fig. 3.15 Variables corrélées avec la panne 2.....	110
Fig. 3.16 Variables corrélées avec la panne 3.....	111
Fig. 3.17 Variables corrélées avec la panne 1 sans Adapt_richesse_fct.....	112
Fig. 3.18 Variables corrélées avec la panne 2 sans Adapt_richesse_fct.....	112
Fig. 3.19 Variables corrélées avec la panne 3 sans Adapt_richesse_fct.....	113
Fig. 4.1 Machine de Turing.....	123
Fig. 4.2 Compression de données sans perte d'information.....	129
Fig. 4.3 Compression de données avec perte d'information.....	130
Fig. 4.4 Distance des logfiles de la panne 1 aux prototypes des trois pannes.....	140
Fig. 4.5 Distance des logfiles de la panne 2 aux prototypes des trois pannes.....	141
Fig. 4.6 Distance des logfiles de la panne 3 aux prototypes des trois pannes.....	141
Fig. 4.7 Distance NCD pour la panne 1.....	143

Fig. 4.8 Distance NCD pour la panne 2.....	143
Fig. 4.9 Distance NCD pour la panne 3.....	144

Liste des tableaux

Tableau 2-1 Caractéristiques des variables utilisées	72
Tableau 4-1 Détermination des intervalles.....	136
Tableau 4-2 Calcul de la valeur du codage	136
Tableau 4-3 Reconstitution	137

INTRODUCTION GENERALE

Introduction générale

La surveillance des systèmes mécatroniques, en particulier ceux intégrés sur les véhicules d'aujourd'hui, est de plus en plus difficile. De tels systèmes, constitués de composants mécaniques, électroniques et logiciels, nécessitent la maîtrise de plusieurs technologies et des contraintes temps réel.

Les véhicules actuels intègrent de multiples systèmes mécatroniques, qui sont à l'origine de l'accroissement des performances et du confort, mais aux dépens d'une remarquable augmentation de leurs complexités. Les dysfonctionnements, liés à la dégradation des composants, à l'installation impropre et à l'implantation de commandes parfois inadaptées affectent significativement les avancées obtenues par l'introduction massive de la mécatronique.

Cet accroissement de la complexité des systèmes électroniques au sein du véhicule augmente la possibilité de défaillance de ses fonctionnalités et rend de plus en plus difficile la détection puis la localisation du composant ou du sous-système défaillant.

La détection et l'isolation des pannes (FDI pour **F**ault **D**etection and **I**solation) sont les deux étapes essentielles qui conditionnent les résultats et les performances d'un processus de diagnostic. Ce dernier se doit d'être à la fois performant, efficace, souple, sécurisant et applicable en temps réel. Toutes ces caractéristiques sont, en réalité, opposées les unes aux autres, et un compromis est primordial pour réussir la phase de détection et d'isolation, pour prendre ensuite les décisions de maintenance qui s'imposent.

Dans le milieu automobile, il existe deux types de diagnostic :

- Le diagnostic embarqué dans certains calculateurs qui surveillent le bon fonctionnement de la fonction, sur la base de l'appartenance ou non de certaines variables à des intervalles prédéfinis. Pour des considérations économiques, les calculateurs embarqués ne possèdent que quelques mégaoctets de mémoire et une fréquence de processeur de l'ordre de quelques MHz. Cette limitation de ressources induit l'incapacité d'embarquer aujourd'hui des algorithmes complexes de diagnostic.
- Le diagnostic d'après-vente, plus orienté vers la localisation des défauts. En effet, pour un véhicule amené en garage, son dysfonctionnement a déjà été détecté, soit

INTRODUCTION GENERALE

par le diagnostic embarqué qui signale le défaut au conducteur, soit par le conducteur lui même.

Les outils de diagnostic en garage s'appuient sur des méthodes à base de connaissances ou sur des modèles analytiques. Bien que ces approches soient efficaces, leurs utilisations nécessitent le développement de modèles mathématiques détaillés et/ou des bases de connaissances personnalisées. Les méthodes de diagnostic par reconnaissance de formes représentent une bonne alternative aux deux méthodes citées précédemment grâce à leur possibilité de suivi de la dynamique de système. Dans ces méthodes, le « diagnostiqueur » conçoit le modèle en se basant sur les données résumant le comportement du véhicule.

L'amélioration des outils de diagnostic dans le domaine automobile représente un des défis de ces dernières années. L'augmentation du nombre de calculateurs permet un suivi précis du mode opérationnel du véhicule, et doit donc en faciliter la surveillance. Cependant, cette augmentation génère aussi une quantité gigantesque d'informations, parfois redondantes. Une sélection des informations recueillies les plus pertinentes est donc nécessaire pour réaliser un processus FDI performant.

La problématique étudiée dans ces travaux se situe dans le domaine de la fusion d'informations, méthodologie initialement développée afin de gérer des quantités très importantes de données multi-sources dans le domaine militaire. Depuis quelques années ces méthodes ont été adaptées pour des applications en traitement du signal et plus particulièrement pour la classification. Plusieurs sens sont donnés à la fusion d'informations, nous reprenons ici la définition proposée par Bloch (Bloch et Maitre, 2002) (Bloch, 2003) (Bloch, 2005) et qui stipule que la fusion d'informations consiste à combiner des informations issues de plusieurs sources afin d'aider à la prise de décision. La qualité de la décision est directement liée à la pertinence des sources d'informations.

La pertinence d'une source d'information est une notion très vague. Plusieurs définitions ont été données, en particulier dans les problèmes de reconnaissance des formes, mais, en général, la pertinence d'une source d'information dépend exclusivement de l'objectif recherché. En adoptant cette définition, nous considérons qu'une variable est pertinente dans un processus FDI, si elle permet à elle seule, de mettre en évidence l'apparition d'un éventuel défaut dans le système surveillé.

INTRODUCTION GENERALE

Le conflit, en particulier celui introduit par Dempster et Shafer en fusion de sources d'informations, nous paraît adapté à la mesure de la pertinence des données. Cependant, ce conflit n'est exploitable que dans le cas de données aléatoires, ce qui ne correspond pas à notre application. Dans cette thèse nous allons exploiter cette notion, mais en proposant une évaluation de la pertinence basée sur l'appartenance ou non d'une donnée à la région de l'espace des paramètres associée au bon fonctionnement du véhicule.

Toujours, dans l'objectif de réaliser un système FDI performant, nous nous sommes intéressés à la théorie de l'information, et en particulier, la théorie de la complexité. Cette théorie, si elle est transposable à notre problématique, doit conduire à un outil de FDI performant ne nécessitant aucun prétraitement des données et peu consommateur de ressources calculatoires. Cette approche se présente comme une alternative intéressante aux méthodes classiques de classification dont la dynamique et les performances dépendent de plusieurs paramètres régulateurs. Il n'est pas étonnant de rencontrer des difficultés pour définir les classes si ces paramètres sont mal choisis. Idéalement, une méthode de classification doit reposer sur un nombre restreint de paramètres. La complexité de Kolmogorov est difficile à estimer, mais plusieurs travaux proposent d'utiliser un algorithme de compression pour estimer cette complexité. Des résultats très encourageants ont été obtenus avec une telle approche en bioinformatique, en apprentissage et en théorie de calculabilité.

La théorie de la complexité présente les avantages suivants:

- Une meilleure extraction des données sans aucun à priori sur celles-ci.
- Des performances bien supérieures à celles des algorithmes classiques.
- Un temps d'exécution et une complexité calculatoire réduits, en particulier si on utilise un algorithme de compression pour estimer la complexité.
- Aucune restriction sur le format des données.

Tous ces avantages ont motivé la mise en œuvre de cette technique pour élaborer un système *FDI* performant qui se basera sur la distance de compression, une distance qui semble bien adaptée pour estimer la similarité entre des ensembles de données.

Ce manuscrit est organisé ainsi :

- Dans le premier chapitre, sont présentées et comparées les méthodes de sélection de variables par apprentissage et les méthodes *FDI*.

INTRODUCTION GENERALE

- Dans le deuxième chapitre, le système FDI proposé est présenté ainsi qu'un bref rappel des méthodes d'apprentissage. Le critère d'évaluation pour sélectionner les variables pertinentes est détaillé, il est suivi par un exemple d'application.
- Le troisième chapitre présente la méthode d'apprentissage choisie ainsi que les résultats obtenus sur une application automobile.
- Dans le dernier chapitre, un autre système FDI, basé sur la complexité de Kolmogorov est proposé. Cette théorie est brièvement rappelée et est suivi d'un passage en revue des techniques de compression de données qui permettent d'estimer la complexité de Kolmogorov. Le compresseur de données PAQ utilisé dans le cadre de ces travaux sera ensuite détaillé, puis la distance de compression utilisée pour calculer les similarités entre ensembles de données sera définie. Cette approche sera testée sur deux applications automobiles.
- Ce manuscrit se termine par une conclusion et des perspectives.

“A good question

beats a good answer”

Ce premier chapitre commence par un état de l'art sur la notion de surveillance, en particulier, les méthodes existantes de sélection de variables en apprentissage. De plus, la notion de pertinence est abordée tout en éclairant son lien avec ces méthodes de sélection de variables. Ensuite, les méthodes existantes en détection et isolation de défauts sont présentées suivies d'une analogie entre la sélection de variables et le système de détection et d'isolation de fautes en systèmes mécatroniques.

1 Surveillance des systèmes

La surveillance des systèmes, en vue d'assurer un bon fonctionnement, consiste à suivre et à piloter un processus automatisé. Pour assurer un bon fonctionnement, nous nous intéressons à la localisation et à l'élimination de toutes défaillances (fautes) dans le système surveillé. Dans cette thèse, nous considérons la définition d'une défaillance comme un fonctionnement anormal (dysfonctionnement), continu ou par intermittence, du système. Un dysfonctionnement peut apparaître sur les capteurs, les actionneurs ou le système de contrôle. Les causes des anomalies ou des dysfonctionnements sont multiples, elles peuvent être provoquées par une panne d'un composant, par des coupures de l'alimentation électrique, par des perturbations extérieures, par des erreurs humaines, par une maintenance inappropriée ou par une panne de système de contrôle. Lors de la surveillance d'un système, nous devons assurer les tâches suivantes :

- détection et isolation des fautes ;
- diagnostic des causes et de l'influence de la faute sur le composant concerné et sur le système entier ;
- reconfiguration et retour au fonctionnement normal du système.

La surveillance d'un processus industriel vise à garantir sa fiabilité, sa maintenabilité et sa disponibilité.

CHAPITRE 1

- a. **Fiabilité** : elle représente l'aptitude d'un système à accomplir une tâche sous des conditions données pour une période quelconque.
- b. **Maintenabilité** : elle représente l'aptitude d'un système à être maintenu ou rétabli dans un état permettant d'accomplir une tâche donnée.
- c. **Disponibilité** : elle représente l'aptitude d'un système à accomplir une tâche donnée pour des conditions définies d'horaire tout en respectant les propriétés de fiabilité et de maintenabilité.

Les exigences requises pour la maintenabilité d'un système mécatronique exposé à des perturbations extérieures ou à l'apparition des fautes sont :

- détection du plus grand nombre de fautes avec un minimum de fausses alarmes ;
- minimisation du temps de la détection de la faute ;
- estimation exacte des caractéristiques de la faute ;
- robustesse de la méthode employée vis-à-vis de l'inexactitude du modèle, de la non-linéarité des composants et des phénomènes extérieurs.

En pratique, ces exigences sont rarement satisfaites toutes à la fois, ce qui contraint à chercher le meilleur compromis possible.

1.1 Systèmes mécatroniques

L'introduction des systèmes mécatroniques (Mihalache, 2007) a provoqué une énorme révolution dans le monde industriel, en particulier l'industrie des transports et plus particulièrement l'industrie automobile. L'utilisation de ces systèmes s'est répandue d'une manière précipitée et son influence demeure importante sur tous les secteurs industriels.

Avec son introduction, la mécatronique a bouleversé la conception et la fabrication des systèmes complexes. Comme cité précédemment, l'industrie automobile est très concernée par cette innovation et les processus de développement et de fabrication sont complètement modifiés.

Avec cette évolution, un véhicule, considéré jusqu'à lors comme un dispositif mécanique combiné avec quelques commandes électroniques, est devenu un véritable système mécatronique (Bertram, et al., 2003) intégrant différentes technologies (Desjardin, 1996).

1.1.1 Définition

Le terme mécatronique a été introduit pour la première fois en 1969 par un ingénieur japonais de la compagnie Yaskawa Electric Corporation. Il vise le contrôle des moteurs électriques par ordinateur (Yaskawa-Electric, 1969). Depuis, ce terme n'a pas cessé de se développer.

La mécatronique est, selon le Comité Consultatif de Recherche Industrielle et de Développement de la Communauté Européenne, une combinaison synergique de l'ingénierie mécanique de précision, de la commande électronique et des systèmes informatiques dans la conception des produits et les processus de fabrication (Comerford, 1994), (Grimheden, et al., 2001). Dans cette définition, nous constatons le caractère multidisciplinaire de la mécatronique. Le mot-clé synergique est utilisé par analogie à son utilisation en génie mécanique. Voici un exemple d'un système mécatronique proposé par Shetty et autres (Shetty, et al., 1997) :

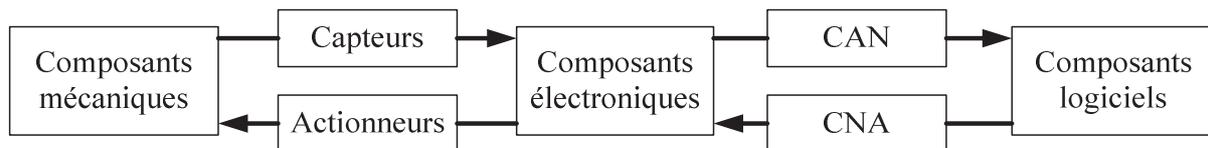


Fig. 1.1 Système mécatronique

Le système représenté sur cette figure rassemble, simultanément, de la mécanique, de l'électronique et du logiciel. En plus de ces trois composantes, nous retrouvons également des systèmes hydrauliques, pneumatiques et des systèmes thermiques. Dans ce type de systèmes, la synergie permet d'optimiser les performances en procédant à une combinaison intelligente et optimale des différentes technologies utilisées en mécatronique. Cette propriété permet d'apporter des solutions innovantes à celles obtenues par ces mêmes technologies prises séparément.

Dans un autre registre, l'arrivée de la mécatronique dans l'industrie automobile a provoqué de nouvelles contraintes qui sont :

- la maîtrise de plusieurs technologies ;
- les interactions entre les différentes entités fonctionnelles ;
- la considération de la contrainte temps réel ;
- la difficulté de réaliser des tests exhaustifs.

Mais, au-delà de ces contraintes, plusieurs avantages sont apportés par la mécatronique :

- la baisse des coûts ;
- la proposition de solutions innovantes ;
- la satisfaction d'exigences importantes comme la sécurité, le confort, ...etc.

Les travaux de (Thesame, 2003), (Saviuc, 2006), (Rzevski, 2003), (Hewit, 1996), (Isermann, 2007), (Millbank, 1993), (Kortum, et al., 1998) et (Ollero, et al., 2006) mettent en avant ces avantages.

1.1.2 La mécatronique et l'industrie automobile

Les premières voitures inventées à la fin de 18^{ième} siècle sont essentiellement composées de systèmes mécaniques. L'apparition de l'électronique d'améliorer les performances des véhicules. C'est ainsi qu'apparaît, en 1980, les systèmes de contrôle du groupe motopropulseur qui ont conduit à la réduction la consommation du carburant et des émissions de gaz polluants. L'incorporation de capteurs, actionneurs et calculateurs se multiplie de plus en plus dans les véhicules. Ce développement nécessite un dialogue permanent entre ces différents composants. Le fonctionnement de ces nouveaux systèmes, caractérisés par une complexité accrue, ne peut en aucun cas se ramener au fonctionnement de l'un de ses composants. Le système ABS, par exemple, regroupe plusieurs fonctions, systèmes ou organes.

Comme cité précédemment, un système mécatronique en automobile est composé de capteurs, d'actionneurs, d'un système de commande (des algorithmes pour contrôler les actionneurs) et d'électronique de puissance. Des exemples de systèmes mécatroniques dans un véhicule sont donnés sur la figure (Fig. 1.2). Ces systèmes peuvent être simples comme le lève-vitre électrique, ou sophistiqués comme l'ABS, l'anti-patinage... etc.

A noter que le futur de l'industrie automobile est lié au développement des systèmes mécatroniques qui devraient continuer à se multiplier dans les prochaines années. Par exemple, la voiture hybride qui combine une propulsion mécanique et électrique, nécessitera une gestion élaborée de la répartition entre les deux pour avoir une seule boîte de vitesses avec les mêmes pédales. Le véhicule à pile combustible (PAC) est une véritable centrale électrique dont il faut gérer les paramètres de production (alimentation, débit) en plus de la propulsion proprement dite.

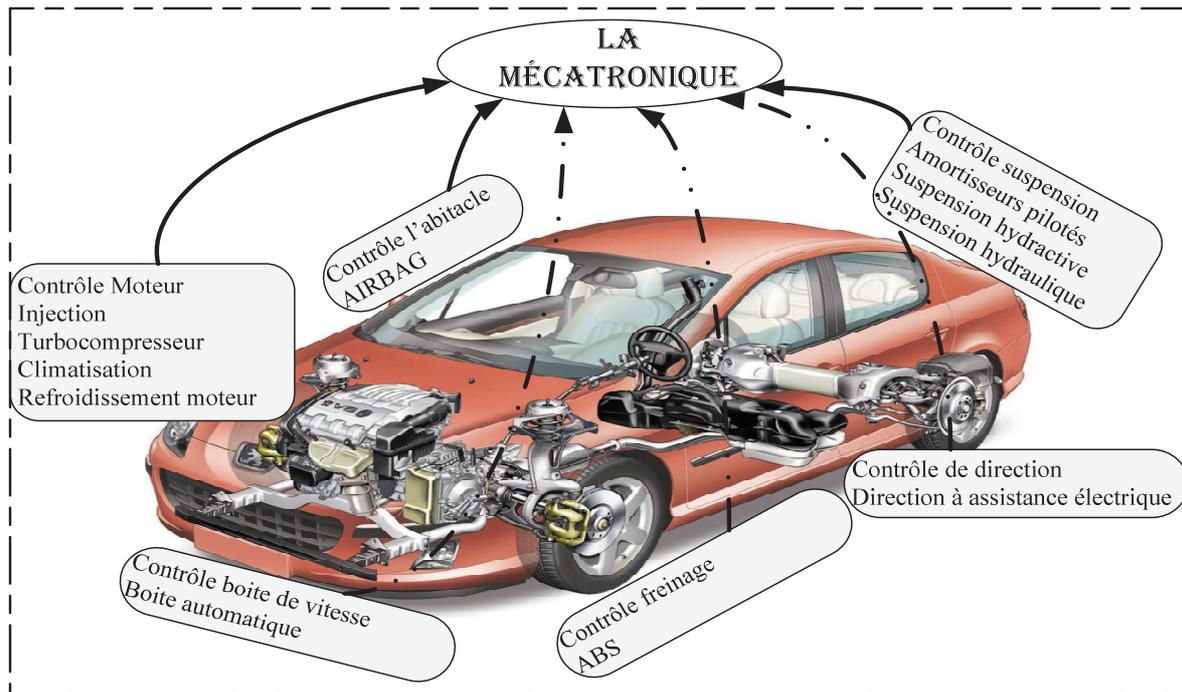


Fig. 1.2 Mécatronique en automobile

1.2 Les approches du diagnostic

Pour assurer la maintenabilité d'un système mécatronique, il nous faut des outils de diagnostic performants et sophistiqués. Le diagnostic automatique tente d'estimer les modes de fonctionnement d'un composant dans un système à surveiller (mode normal ou anormal). Cette estimation se base sur le modèle du système, qui peut être de différentes natures suivant les informations qu'il utilise. On peut ainsi distinguer trois types d'approches de diagnostic : à base de modèles analytiques, à base de connaissances et à base de reconnaissance de formes.

1.2.1 Les types de données enregistrées

Dans une partie de cette thèse, consacrée aux systèmes automobiles, nous allons utiliser des données acquises par un outil développé au sein de l'entreprise (ACTIA) présenté dans les travaux de Poullard (Poullard, 1996). Les données récoltées peuvent être des signaux analogiques, numériques, fréquentiels ...etc.

1.2.2 Les différents types de pannes par composant

Une classification pertinente des pannes peut être basée sur la nature du composant affecté. Nous distinguerons donc les pannes d'actionneurs, de celles liées aux capteurs et celles du système de commande.

1.2.2.1 Panne de Capteurs

Dans ce cas, la panne est provoquée par l'état physique du sous-système défaillant (ruptures, corrosions, déformations,...etc.). Si la panne est partielle, le signal généré par ce capteur sera erroné et l'écart entre la valeur réelle et la valeur mesurée est plus ou moins important. Par contre, une panne totale d'un capteur oblige ce dernier à renvoyer une valeur mesurée complètement différente de la valeur réelle de la grandeur physique surveillée. Une telle mesure va induire le système de commande à produire des signaux de commandes inadéquats et peut aller jusqu'à provoquer la détérioration du système surveillé. Pour palier ces pannes de capteurs, certains industriels utilisent la notion de redondance en réalisant plusieurs copies sur le capteur et le principe de relais est adopté.

1.2.2.2 Panne d'actionneurs

Ces pannes agissent sur les actionneurs du système, elles modifient le signal de commande approprié. Ces pannes correspondent soit à une perte partielle d'un actionneur agissant sur le système, soit à une perte totale. Une perte totale peut être indiquée par la stagnation du signal émis par l'actionneur à une valeur donnée, cette perte provoque la non commande du système.

1.2.2.3 Panne des composants du système

Elle concerne le système à surveiller, elle peut agir sur n'importe quel composant de ce dernier. En général, toute panne qui n'est pas liée à une défaillance d'un capteur ou d'un actionneur est rangée dans cette catégorie. Une panne de composant est provoquée par la rupture ou l'altération d'un composant élémentaire du système, réduisant la capacité de ce dernier à réaliser la tâche qui lui correspond.

1.2.3 Les différentes pannes selon la dynamique

Il existe plusieurs travaux liés à la catégorisation des défauts dont on peut citer ceux d'Isermann et autres (Isermann, et al., 2002). Ils proposent de diviser les défauts en trois catégories : selon leurs formes (systématique ou aléatoire), selon l'étendue de défaut (local ou global) et selon leur évolution dans le temps.

A. Défaut abrupt : il est caractérisé par une discontinuité brutale de l'évolution temporelle d'une variable du procédé. Il provoque un écart dans l'évolution de la variable en question par rapport à l'évolution normale.

B. Défaut intermittent : il représente une particularité des défauts précédents avec des départs et retours successifs aux valeurs normales. Il est souvent provoqué par des faux contacts.

C. Défaut graduel : il conduit à une divergence lente et continue des variables concernées. C'est un défaut difficile à détecter.

La figure suivante représente les trois types de défaut pouvant surgir dans un procédé :

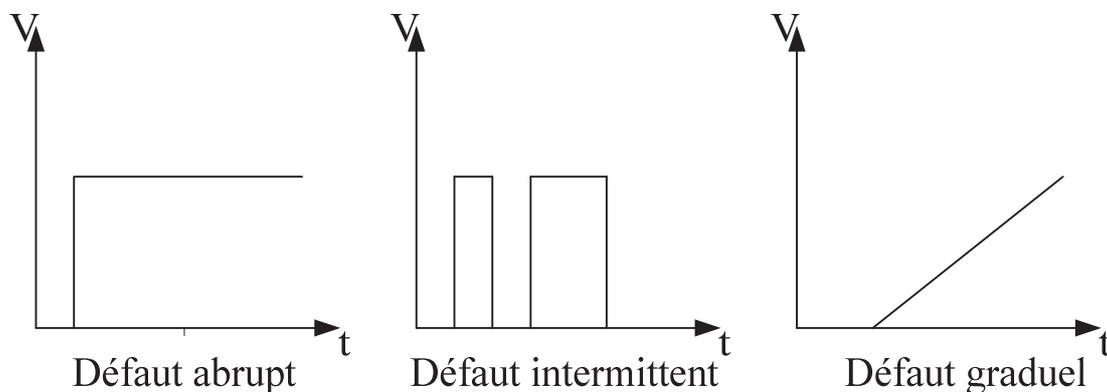


Fig. 1.3 Evolution temporelle des défauts

1.2.4 Approches à base de modèle

Ces méthodes se basent sur un modèle quantitatif de détection et d'isolation de fautes et comparent les mesures disponibles avec l'information à priori représentée par le modèle mathématique du système. Il existe trois principales variantes de cette approche : les méthodes dites à redondance analytique (méthodes à génération de résidu), les méthodes par observateur et les méthodes d'estimation de paramètres. Cette distinction n'est pas universellement adoptée, c'est l'hypothèse qui sera adoptée dans ce travail. Concernant les méthodes à redondance analytique, leur nom a été inspiré de l'utilisation des relations analytiques détaillant la dynamique d'interconnexion entre les différents composants du système. Cette méthode introduite par Chow et autres (Chow, et al., 1984), n'a cessé de progresser jusqu'à atteindre l'approche actuelle dite FDI (Fault Detection & Isolation). Nous pouvons distinguer deux axes de recherche traitant les méthodes à redondance analytique : d'une part les méthodes de l'espace de parité développées aux Etats-Unis et d'autre part les méthodes à base d'observateur traitées en Europe. Cependant, Gertler (Gertler, 1991) a

CHAPITRE 1

démontré que ces méthodes sont équivalentes, la seule différence est qu'elles utilisent des outils mathématiques différents dans la recherche de la robustesse.

Les méthodes par redondance analytique se basent sur le calcul de résidus. Ces résidus sont calculés par la soustraction de la sortie estimée de système \hat{Y} , obtenue en appliquant les entrées U sur le modèle du système, de la sortie réelle Y du système. Dans la situation idéale, ce résidu doit être statistiquement nul, ce qui n'est pas le cas dans le monde industriel. Les résidus différents de zéro sont provoqués soit par le bruit de mesure soit par les pannes. La difficulté majeure de ces méthodes est de distinguer les résidus provoqués par le bruit de ceux provoqués par des pannes. Dans le cas où le bruit est négligeable, les résidus sont directement analysés. Sinon, une analyse statistique s'impose. Afin de distinguer ces deux phénomènes (bruit ou faute), une forme logique appelée signature de faute est générée. Généralement, les résidus provenant du bruit ont une amplitude faible devant ceux provenant de fautes. Donc, l'application d'un seuil adéquat permet de faire la distinction. Selon la forme de modélisation du bruit, les méthodes par redondance analytique peuvent être classées en deux catégories : celles qui utilisent un modèle stochastique et celles qui utilisent un modèle déterministe.

Les résidus sont générés de différentes manières. Cependant, trois approches essentielles peuvent être distinguées (Chiang, et al., 2001) :

- **Equation de parité** : son principe est de vérifier l'exactitude des équations mathématiques du modèle en se basant sur les observations du système.
- **Estimation de paramètres** : les résidus sont calculés par la soustraction de paramètres nominaux de modèle aux paramètres estimés du modèle.
- **Observateurs** : ces observateurs permettent, en exploitant les mesures réelles du système, de reconstruire une estimation des sorties non accessibles. Cette estimation sera comparée aux sorties réelles du système pour calculer les résidus.

L'organigramme suivant représente une procédure de détection et d'isolation de faute en utilisant la méthode de redondance analytique :

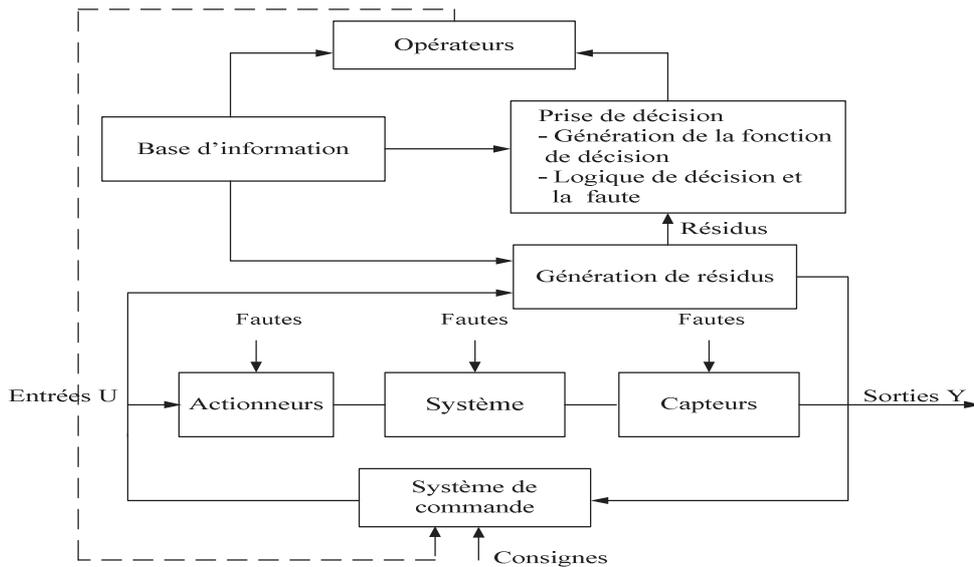


Fig. 1.4 FDI par redondance

Soit le système mécatronique représenté par les équations suivantes :

$$\begin{cases} \dot{X}(t) = A.X(t) + B.U(t) \\ S(t) = C.X(t) + D.U(t) \end{cases} \quad (1.1)$$

Où $X(t)$ est l'état interne du système, $U(t)$ la commande et $S(t)$ sa sortie. Le schéma bloc représentant la fonction génératrice des résidus est le suivant :

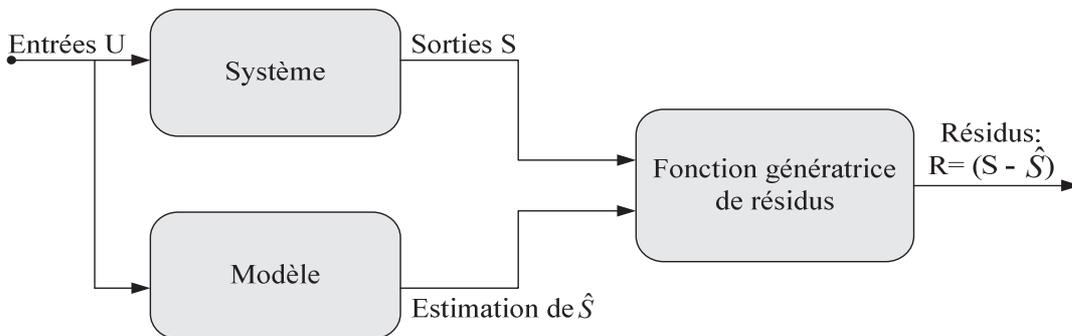


Fig. 1.5 Génération de résidus pour la détection des pannes

Les résidus générés en utilisant l'approche par redondance analytique se basent sur les relations nominales du système en exploitant l'entrée U et la sortie mesurée Y . Les relations de redondance à évaluer peuvent être considérées comme la dynamique de la relation Entrée-Sortie. Dans le cas de l'apparition d'une faute, cette relation n'est jamais satisfaite et des résidus non-nuls sont obtenus. Ces résidus sont utilisés pour prendre les décisions

appropriées. Ils seront évalués dans la composante de la logique de décision de faute pour déterminer le temps d'apparition de la faute et localiser l'origine de la faute.

1.2.5 Approches à base de connaissances

Avec le développement de systèmes industriels de plus en plus complexes, la surveillance, la commande tolérante aux fautes, la détection, l'isolation et le diagnostic de fautes deviennent extrêmement décisifs et fondamentaux. Dans ces cas, une simple baisse de vigilance peut entraîner des catastrophes dévastatrices (la catastrophe de Tchernobyl). De tels systèmes sont difficilement modélisables par des équations mathématiques, raison pour laquelle, nous faisons appel aux connaissances humaines. On peut distinguer plusieurs méthodes qui s'appuient sur les connaissances qualitatives que possèdent des experts concernant le fonctionnement et la surveillance du système exploité. Parmi ces méthodes, on peut mentionner la méthode d'Analyse des Modes de Défaillance et de leurs Effets (AMDE) (Faucher, 2004), la méthode d'Analyse des Modes de Défaillance, de leurs Effets et de leurs Criticités (AMDEC), les systèmes experts (Chatain, 1993) et les arbres de défaillances (Zwingelstein, 1995).

1.2.5.1 *Systèmes experts*

Les Systèmes Experts industriels représentent un outil pour codifier et organiser, sous forme informative, les expériences les opérations de réflexion et l'expertise d'un homme sur le déroulement d'un processus technologique, d'un plan industriel ou sur les caractéristiques du système. En d'autres termes, les systèmes experts imitent le raisonnement humain pour résoudre des tâches bien spécifiques. C'est une technique issue de l'intelligence artificielle. Elle est capable de réaliser plusieurs tâches qui sont : la représentation de l'expertise humaine disponible (donc, des connaissances), la prise en considération des bases de données existantes, l'acquisition de nouvelles connaissances, la réalisation d'inférence logique, la proposition de suggestions et, en dernier, la prise de décision basée sur un raisonnement. Un système expert est composé de quatre composantes principales :

- la base de connaissance ;
- le moteur d'inférence ;
- l'interface avec l'utilisateur ;
- l'interface avec l'expert.

Un système expert peut être résumé dans la figure suivante :

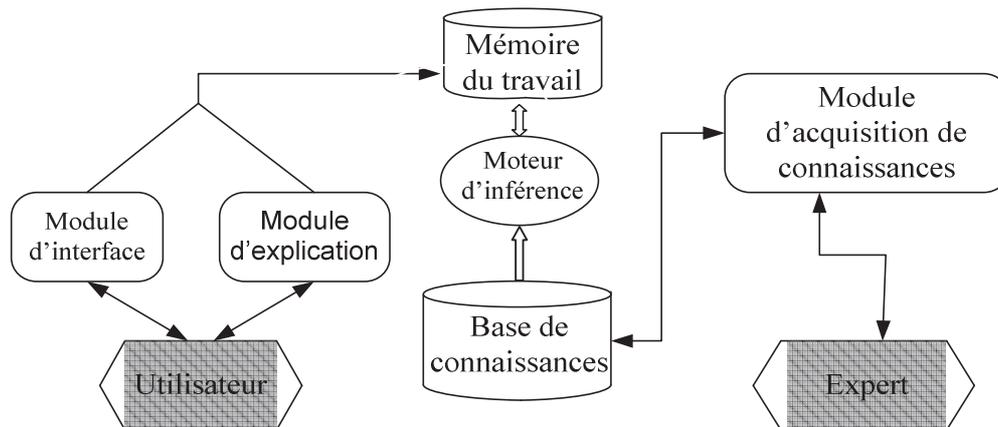


Fig. 1.6 Système expert

Dans cette approche, la tâche d'élaboration de la base de connaissances est ardue, surtout, quand le système à maîtriser est complexe. Ce désavantage limite l'application de cette technique à des systèmes simples.

1.2.5.2 Les méthodes AMDE et AMDEC

L'AMDE est une méthode d'analyse complète, qui explore tous les modes de défaillance possibles, en traitant composant par composant, et déduit leurs effets sur le système entier. Cette méthode est issue de la communauté de la sûreté de fonctionnement. Dans cette méthode, l'utilisateur génère un tableau récapitulatif de toutes les défaillances possibles de chaque composant et estime leurs effets sur le système global. L'ajout de la phase d'étude de la criticité à la méthode AMDE permet d'élaborer la méthode AMDEC. Cette étude permet d'évaluer et de distinguer les modes de défaillances les plus importantes en considérant la gravité des différents modes ainsi que les fréquences d'apparition.

Ces méthodes présentent les mêmes difficultés que les systèmes experts, à savoir qu'il est impossible de les utiliser dans un système complexe. Par exemple, une démarche AMDEC nécessite un effort considérable, voire irréalisable, afin de recenser tous les modes de défaillances possibles.

1.2.5.3 Les arbres de défaillance

L'analyse d'un système par arbre de défaillance permet d'exprimer, pour un événement redouté (dysfonctionnement ou accident), les scénarios conduisant à cet événement. Cette

étude se fait en s'appuyant sur la connaissance des éléments constitutifs du système étudié. Un arbre de défaillance décrit de manière synthétique tout ce qui peut conduire à l'événement redouté. Une analyse qualitative permet d'évaluer l'effet d'une modification du système et de comparer les conséquences des mesures envisagées. Une analyse quantitative consiste à évaluer la probabilité d'occurrence de l'événement redouté à partir de celles des événements élémentaires.

1.2.6 Approches se basant sur la reconnaissance de forme

Ces techniques qui se basent sur la reconnaissance de formes se divisent en deux catégories. La première catégorie est dédiée à la détection tandis que l'autre catégorie est réservée au diagnostic. Pour les méthodes dédiées à la détection nous pouvons citer les approches par carte de contrôle (Pillet, 2001), les approches par analyse en composantes principales (Lebart, et al., 2000) et les approches par projection dans les structures latentes. Concernant les méthodes dédiées au diagnostic, nous retrouvons toutes les méthodes de classification. Dans ce travail, on s'intéresse à la deuxième catégorie de méthodes.

Lors d'un processus de diagnostic, on tente d'identifier les différents modes de fonctionnement d'un système donné. Ces modes peuvent être nominaux ou défaillants. Chacun de ces modes occupe une partie de l'espace des paramètres du système. Les techniques de classification (Duda, et al., 2001) permettent de définir ces régions d'espace, et aussi, de discriminer les modes défaillants des modes nominaux.

La tâche principale d'un processus de diagnostic est d'attribuer à un nouvel élément du système une classe (nominale ou défaillante). Un système de diagnostic performant, se basant sur la classification (Denoeux, et al., 1997), doit réaliser avec succès ces différentes tâches :

- identifier les différentes classes du système ;
- classer correctement les nouveaux éléments ;
- détecter l'apparition de fautes.

Il existe plusieurs types de classifieurs, par exemple : les réseaux bayésiens (Friedman, et al., 1997), les arbres de décisions (Cornuéjols, et al., 2002), les machines à vecteurs supports (Vapnik, 1995), les K plus proches voisins (Cover, et al., 1967) et les réseaux de neurones (Dreyfus, et al., 2004). Ce type de diagnostic est connu sous le nom de Diagnostic par Reconnaissance de Forme (Diagnostic RDF).

La reconnaissance de forme est un des nombreux axes de l'intelligence artificielle. Elle permet, en se basant sur un ensemble de données apprises, d'interpréter toute nouvelle observation (forme). Les données apprises sont stockées sous forme de classes appelées prototypes. Une nouvelle observation sera comparée à ces prototypes et, ensuite, elle sera attribuée à l'un de prototypes. En reconnaissance de forme, on distingue deux approches : reconnaissance de forme structurelle (Friedman, et al., 1999) et reconnaissance de forme statistique (Fukunaga, 1972). Le premier type utilise les propriétés grammaticales pour caractériser les formes, tandis que le deuxième se base sur les formes numériques étudiées. Dans notre cas, on s'intéresse au deuxième type qui est la reconnaissance de formes statistiques.

En reconnaissance des formes, une observation est caractérisée par un vecteur X de dimension N (nombre de variables) : $X = (X_1, X_2 \dots X_N)$. Elle sera représentée par un point dans l'espace de dimension N

Le vecteur forme global représente l'état réel du système étudié. Il est composé des différentes acquisitions en provenance des capteurs embarqués sur le système. Soit : $C = (C_1, C_2, \dots, C_M)$ l'ensemble de prototypes (formes) regroupant des observations similaires. Le rôle de la reconnaissance des formes est d'attribuer une nouvelle observation à l'un des prototypes disponibles. La figure suivante explique ce principe :

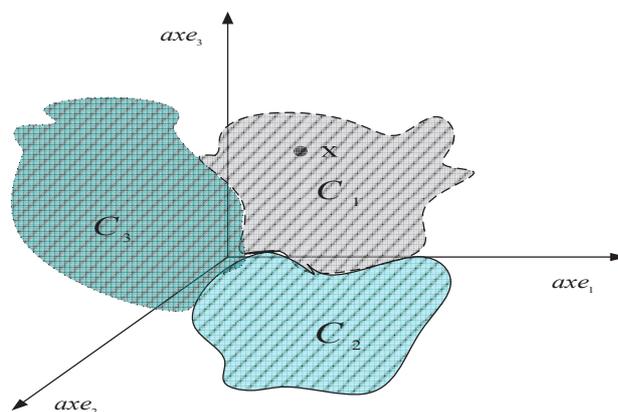


Fig. 1.7 Principe de la reconnaissance des formes

Sur cette figure, la nouvelle observation x est attribuée à la classe C_1 .

Le processus de reconnaissance des formes se déroule en trois phases principales :

Phase d'analyse

L'objectif principal de cette phase est d'étudier les observations (informations) en provenance de capteurs implantés sur le système. Généralement, ces informations sont fournies sous forme de signaux qui nécessitent une étape d'extraction de valeurs numériques. Le schéma suivant résume cette phase :

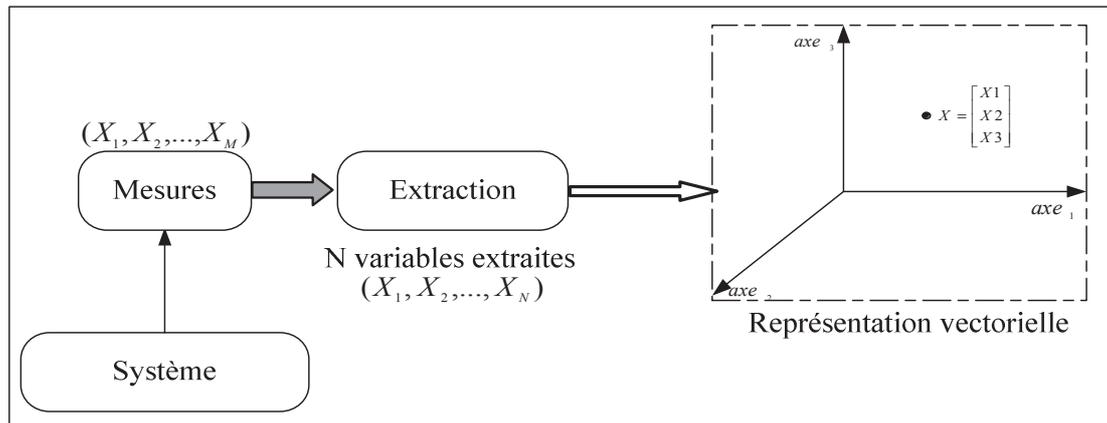


Fig. 1.8 Phase d'analyse : constitution du vecteur forme

Durant la phase d'analyse, on doit regrouper les N observations disponibles en M classes. Le résultat de ce regroupement constitue l'ensemble d'apprentissage. Cette étape est très lourde en termes de calculs.

Phase de choix du système de détection

Cette phase consiste à établir des frontières entre les différentes classes du vecteur forme. Ces frontières seront gérées par une règle de décision qui attribuera à chaque nouvelle observation une classe donnée.

Phase d'exploitation

La règle de décision sera implémentée durant cette phase et elle traitera toute nouvelle observation en lui attribuant la classe adéquate. La qualité de l'exploitation dépendra fortement de la pertinence du vecteur forme et des performances de la règle de décision.

Lors de la mise en place d'un système de diagnostic par reconnaissance de forme, nous devons répondre à différentes questions dont la plus essentielle est : quelles sont les variables (ou caractéristiques) les plus pertinentes représentatives du système à diagnostiquer ? En répondant à cette question, l'utilisateur va réduire la quantité d'information à traiter lors de processus du diagnostic.

1.3 La pertinence :

Le diagnostic à base d'une architecture cognitive tente de détecter un comportement anormal d'un système. Cependant, dans notre cas, le diagnostic n'est pas lié au comportement humain mais au bon fonctionnement des systèmes complexes « healthy behavior ».

La définition de la pertinence d'une variable est relative, elle dépend de l'objectif poursuivi. De ce fait, il est difficile de donner une définition exacte de la pertinence en science de l'ingénieur, notamment, en traitement de données. En apprentissage de données, il existe différentes définitions de la pertinence d'une variable, cette variété est due essentiellement au fait que la pertinence d'une variable dépend de l'objectif poursuivi.

1.3.1 La pertinence et les systèmes cognitifs

1.3.1.1 Introduction

La caractéristique principale de la communication humaine, qu'elle soit verbale ou non, est l'expression et la reconnaissance des intentions. La pertinence est une propriété potentielle non seulement de l'énoncé ou du phénomène observé, mais également de pensées, mémoires et conclusions d'inférences. Par convention, le récepteur considère que le message reçu est systématiquement pertinent. Dans leurs travaux, Dierdre et ses collègues (Deirdre, et al., 2004), ont démontré que la pertinence est liée à une double intentionnalité : un locuteur présente deux sentiments différents, le sentiment de transmettre un message et le sentiment de bien transmettre ce dernier tout en l'accompagnant de son intension. Ils considèrent qu'une information est pertinente pour un individu si, en l'associant aux connaissances dont il dispose, elle lui permet, de répondre aux questions qu'il a à l'esprit, d'améliorer ses connaissances dans certains domaines, d'éliminer des doutes, de confirmer des soupçons et de corriger des impressions fausses.

Dans la théorie de la pertinence, l'interprétation d'un message fait intervenir deux types de mécanismes, celui du décodage et celui d'inférence. Le mécanisme d'inférences combine l'opération de décodage du message avec le contexte disponible pour déduire de nouvelles propositions qui définissent l'objet de la communication. La difficulté majeure de cette opération est de déterminer l'étendue et le contenu du contexte. Sperber et Wilson (Sperber, et

CHAPITRE 1

al., 1989) proposent de se limiter à l'ensemble des informations contextuelles accessibles à travers le sens de l'énoncé. Si ces informations sont nulles ou pauvres en qualités et quantités, des informations moins accessibles seront sollicitées. Selon Sperber et Wilson, un contexte est un ensemble de propositions qui peuvent être vraies, probablement vraies, plutôt vraies, plutôt fausses, ou fausses. Ces étiquettes représentent la force des propositions. Cette définition de contexte permet d'évaluer la pertinence du message et par conséquent de se prononcer sur la réussite ou l'échec de la communication. Afin de mieux clarifier le mécanisme d'inférences, regardant ces deux définitions :

- La contextualisation : (de la proposition P dans le contexte C) est une déduction produite de l'union d'informations nouvelles P et d'informations anciennes C.
- L'implication contextuelle : (de la proposition P dans le contexte C) est une conclusion nouvelle produite à partir de P et C ensemble.

Selon Sperber & Wilson, une entrée est pertinente si son traitement apporte des effets cognitifs positifs. Dans différentes situations, le même stimulant peut être bon ou mauvais, les mêmes hypothèses sont plus au moins accessibles et le même effet cognitif est plus au moins facile à avoir. Donc, la pertinence peut être caractérisée par l'effort de traitement nécessaire et l'effet contextuel obtenu.

1.3.1.2 Effet contextuel :

L'opération de traitement du message s'achève généralement par la production de certains effets contextuels. Ces derniers sont le résultat d'un processus d'interprétation du message. Si dans un processus de contextualisation, nous ne faisons qu'ajouter au contexte de nouvelles informations, sans engendrer de modification du contexte, alors, cette contextualisation est sans effet contextuel. Sinon, l'un des effets suivants sera constaté :

- formulation d'implications contextuelles ;
- adaptation de la force de certaines hypothèses du contexte ;
- effacement de certaines hypothèses du contexte.

1.3.1.3 Effort de traitement :

C'est l'effort fourni par l'individu dans l'objectif d'interpréter le message reçu. Ces efforts de traitement correspondent à l'énergie dépensée par l'individu lors de son activité mentale. Il diffère d'un individu à un autre suivant la longueur de l'énoncé, la facilité d'accès aux informations ou encore du nombre de règles logiques impliquées par le mécanisme

déductif. Ainsi et en utilisant ces deux notions, Sperber & Wilson caractérisent la pertinence d'une proposition par l'importance de ces effets contextuels et la faiblesse de l'effort de traitement nécessaire pour l'interpréter.

1.3.1.4 Evaluation de la pertinence en systèmes cognitifs

La théorie de la pertinence tente de décrire comment l'esprit évalue ses propres résultats et efforts et comment il décide d'orienter ses efforts dans une direction quelconque plutôt qu'une autre. La quantification de la pertinence d'une proposition est fondamentale, elle permet alors de comparer les pertinences de plusieurs propositions ou de déterminer la forme la plus pertinente d'une intention communicative. Cette quantification révèle un problème psychologique général avec plusieurs facteurs.

En pratique, il est quasiment impossible de modéliser les opérations élémentaires constitutives des processus intellectuels complexes, ce qui entraîne l'impossibilité d'évaluer l'effort de traitement d'un énoncé. De plus, il n'existe pas d'indices fiables permettant par exemple de mesurer la durée d'un processus mental tout en ayant connaissance de la difficulté de distinguer une réflexion intense d'une réflexion détendue. Certains auteurs (Gazdar & Good) proposent de quantifier les effets contextuels en comptant le nombre d'implications contextuelles, et l'effort de traitement en comptant le nombre d'opérations de déduction. Par la suite, Sperber & Wilson montrent que ces opérations de comptage amplifient l'effort de traitement et que si la caractérisation de la pertinence est obtenue par des opérations de comptage, elle devient bornée. Cependant, quelques indices ont une influence forte sur la pertinence d'une proposition. Comme la pertinence d'une proposition (effets et efforts) est fortement dépendante de l'individu, de ses expériences et du contexte, il serait difficile de quantifier la pertinence. En analysant les arguments cités précédemment, la pertinence ne pourra être quantifiée d'une manière précise, mais il est possible de le réaliser en utilisant des jugements comparatifs.

1.3.2 La pertinence de variables en apprentissage de données

Avec l'évolution technologique récente de l'électronique embarquée et le développement de l'informatique et des capacités de calcul, les procédés industriels sont de plus en plus surveillés en augmentant le nombre de capteurs actionneurs embarqués à bord. Ces capteurs émettent à leurs sorties des quantités d'informations gigantesques représentant

CHAPITRE 1

l'évolution réelle du système à surveiller. Les algorithmes d'apprentissage, que soit supervisés ou non-supervisés, présentent des difficultés majeures lors du traitement de ce type de données. Généralement, la qualité des modèles d'apprentissage obtenus est jugée insuffisante pour assurer une surveillance efficace des systèmes. De ce fait, l'utilisation d'un sous-ensemble de variables, n'incluant que les variables pertinentes du système à surveiller, lors de processus d'apprentissage est largement recommandée. En allant dans ce sens, Porter et autres (Porter, et al., 1990) ont montré que moins de la moitié des variables disponibles sont réellement utilisées lors de la plupart des applications.

Comme indiqué précédemment, la pertinence d'une variable dépend principalement de l'objectif à atteindre. Le paragraphe qui suit concerne la pertinence dans un processus de classification. Dans le suivant, nous nous intéressons à la sélection de variables pertinentes pour un processus de diagnostic.

1.4 La sélection des variables pertinentes pour la classification

1.4.1 Introduction

La sélection de variables, ou la sélection d'attributs, est un domaine de recherche très ancien qui date des années 70 (Mocciardi, 1971). C'est une thématique qui en recouvre plusieurs : les statistiques (Narendra, et al., 1977), (Blum, et al., 1992) et (Miller, 1990), la reconnaissance de forme (Jain, et al., 1982), (Stearns, 1976) et (Kittler, 1978), le traitement de données (Chen, et al., 1996), les algorithmes d'apprentissage (Hall, 1999), les réseaux de neurones (Setiono, 1997), la programmation mathématique (Manabu, et al., 1984) et (Bradley, et al., 1999), ...etc.

Les méthodes de sélection des variables visent à réduire la dimension de l'espace des variables, en éliminant les variables non pertinentes, redondantes et bruitées. La sélection des variables influence positivement la tâche d'analyse des données, elle permet de réduire le temps d'exécution des algorithmes d'apprentissage, de mettre en avant la partie utile des données et d'augmenter la précision des modèles obtenus.

- **Formulation :**

Soit X l'espace des variables initial, d sa dimension. Le sous-ensemble de variables sélectionnées est Y de dimension d' tel que : $Y \subseteq X$. Si $J(Y)$ est le critère de sélection, alors, le sous-ensemble de variables maximisant ce critère est le plus pertinent. Le problème de la sélection des variables peut être résumé par l'équation suivante :

$$J(Y) = \max_{Z \subseteq X, |Z|=d'} J(Z) \quad (1.2)$$

Dans le cas d'une exploration exhaustive, l'utilisateur doit évaluer tous les sous-ensembles possibles. Cette tâche devient irréalisable avec l'augmentation de la dimension de l'espace de variables. La recherche des sous-ensembles de variables pertinents est souvent difficile (Kohavi, et al., 1997). Dans l'article de (Breiman, et al., 1984), les auteurs ont démontré que la plupart des problèmes de sélection sont NP-difficile. Nous pouvons classer les méthodes de sélection selon la stratégie employée :

- stratégie basée sur la connaissance du nombre de variables souhaité d' . Dans ce cas, l'objectif de la recherche est de déterminer les d' variables constituant le sous-ensemble optimal ;
- recherche du nombre minimal de variables permettant d'obtenir une discrimination supérieure à un seuil prédéfini ;
- détermination du sous-ensemble de variables tout en respectant un compromis entre la discrimination de classes (i.e. taux d'erreur de classification) et la dimension du sous-ensemble (i.e. nombre de variables sélectionnées).

La sélection des variables recherche un sous-ensemble des variables qui possède quelques propriétés désirées. Ces propriétés dépendent fortement des variables qui constituent le sous-ensemble défini, ce qui entraîne une difficulté majeure de trouver une fonction d'évaluation simple. De plus, le nombre de sous-ensembles candidats augmente exponentiellement avec l'augmentation de la dimension de l'espace initial. Langley (Langley, 1994) résume les méthodes de sélection de variables en quatre étapes qui sont présentées ci après.

1.4.2 Les différentes étapes de sélection des variables

1.4.2.1 La procédure de génération

Elle permet de générer les sous-ensembles à évaluer dont le nombre est égale à 2^d pour d variables. En pratique, il est crucial de générer les meilleurs sous-ensembles tout en essayant d'explorer tout l'espace de recherche. Les approches heuristiques permettent à chaque étape d'ajouter une nouvelle variable au sous-ensemble existant. Dans les approches aléatoires, l'utilisateur sélectionne aléatoirement les sous-ensembles de variables et choisit celui qui possède la meilleure évaluation. La génération commence généralement avec un ensemble vide en rajoutant une variable à chaque itération : stratégie en avant. Il existe aussi la stratégie en arrière qui commence par tout l'espace de variables qui sera réduit en éliminant

CHAPITRE 1

une variable à chaque itération. Enfin, quelques méthodes proposent d'appliquer l'une des deux stratégies que nous venons de citer à un sous-ensemble de variables choisies aléatoirement.

1.4.2.2 La fonction d'évaluation

Cette fonction permet de qualifier la capacité de classification d'une ou de plusieurs variables. Dash et autres (Dash, et al., 1997) proposent une classification intéressante des fonctions d'évaluations :

- Mesure de distance : Supposons deux classes C_1 C_2 , la variable X est préférée à la variable Y si :

$$P(C_1|X) - P(C_2|X) > P(C_1|Y) - P(C_2|Y) \quad (1.3)$$

Cette équation est vérifiée si la variable X provoque une augmentation de la probabilité conditionnelle par rapport à la variable Y .

- Mesure d'information : elle permet de mesurer la quantité d'information contenue dans chaque variable. Une variable X est préférée à une variable Y si le gain d'information apporté par X est plus élevé que celui apporté par Y .
- Mesure de dépendance ou de corrélation : elle indique la capacité d'un sous-ensemble de variables à prédire la valeur des autres. Dans cette logique, X est pertinente par rapport à Y si sa corrélation avec la classe à prédire est la plus importante. Cette mesure permet d'éliminer les redondances.
- Mesure de consistance : elle permet d'évaluer la capacité de la variable sélectionnée à séparer des objets de différentes classes. Si une variable possède deux échantillons identiques qui appartiennent à deux classes différentes, alors, elle ne sera pas sélectionnée.

1.4.2.3 Le critère d'arrêt

Il permet d'arrêter l'algorithme dans le cas où l'ajout de variables n'améliore pas la fonction d'évaluation. La recherche peut être stoppée si, le nombre d'attributs fixé par l'utilisateur est atteint, ou si l'amélioration apportée par l'ajout d'une nouvelle variable est faible.

1.4.2.4 La procédure d'évaluation

Elle mesure la qualité du sous-ensemble sélectionné. Selon la procédure d'évaluation, les méthodes de sélection de variables sont divisées en trois méthodes principales : les méthodes de type filtre, les méthodes de type Wrapper et les méthodes dites embarquées. Dans le premier type, la fonction d'évaluation est indépendante de l'algorithme de classification. Pour le second type, le résultat de la classification permet d'évaluer le sous-ensemble sélectionné (calcul de la qualité de la classification). Pour les méthodes embarquées, la sélection se fait en simultané avec la classification. La figure suivante représente l'architecture générale des modèles filtre et Wrapper :

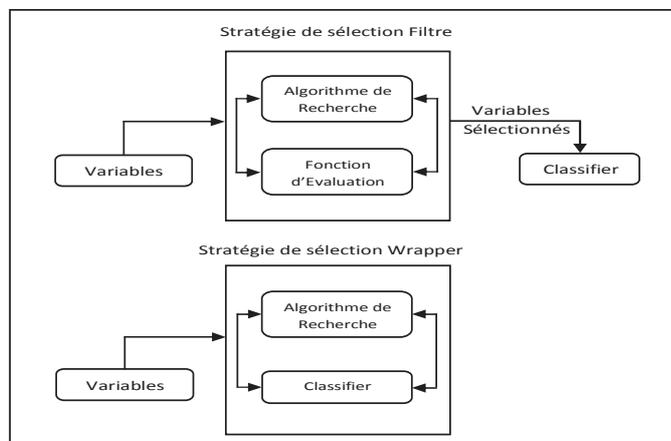


Fig. 1.9 Stratégies de sélection de variables

1.4.3 Catégorisation des méthodes de sélection des variables

Il existe plusieurs travaux dans la littérature dont l'objectif est de comparer les méthodes de sélection des variables (Mocciardi, 1971), (Kudo, et al., 2000), mais peu d'approches proposées ont permis de catégoriser ces méthodes. Seidlecki et autres (Siedlecki, et al., 1988) traitaient l'évolution de la sélection des variables avec le temps et proposent de les classer en méthodes passées, présentes et futures. Jain et Zongker (Jain, et al., 1997) proposent, dans un premier temps, de diviser les méthodes de sélection en deux catégories : celles inspirées des techniques de classification par reconnaissance de forme statistique et celles inspirées des réseaux de neurones artificiels. La première catégorie est à nouveau divisée en plusieurs sous-catégories selon la monotonie de critère de sélection ainsi que la complexité algorithmique de l'approche (Pudil, et al., 1994). Une autre catégorisation a été introduite selon la connaissance ou non de l'algorithme de classification à utiliser. Enfin, les méthodes de sélection de variables peuvent être classées selon l'optimalité ou non de la

CHAPITRE 1

procédure d'exploration (recherche partielle ou exhaustive). L'arbre de la figure (Fig. 1.12) représente la catégorisation proposée par Jain et Zongker : la catégorisation la plus pertinente est celle qui se base sur la procédure d'évaluation (modèle filtre, modèle Wrapper et modèle embarqué).

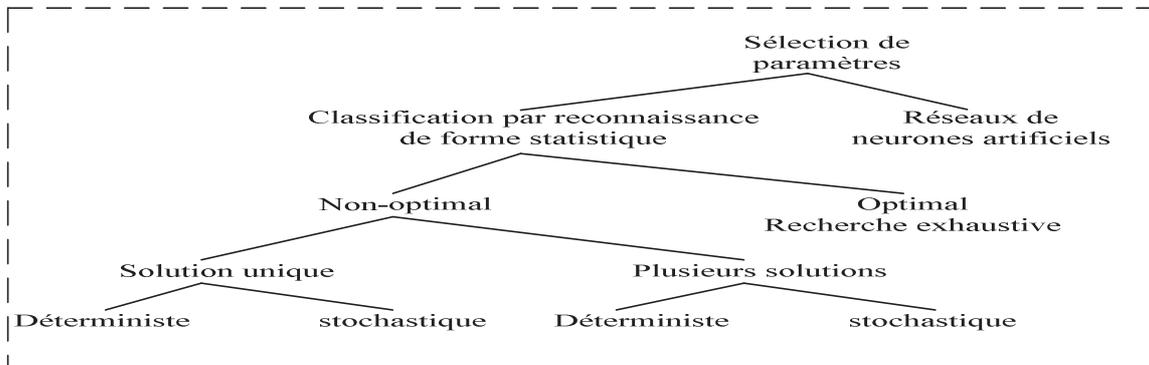


Fig. 1.10 Catégorisation de Jain et Zongker

1.4.3.1 *Modèle de sélection filtre*

C'est l'une des premières approches de sélection de variables. Elle utilise un critère de recherche indépendant pour extraire les sous-ensembles de variables pertinentes en amont de l'algorithme d'apprentissage. Son nom, filtre, est inspiré de la stratégie en amont (John, et al., 1994) qui permet d'éliminer les variables non pertinentes avant de lancer l'algorithme d'apprentissage. Ainsi, la sélection est faite indépendamment de l'algorithme d'apprentissage. Dans le modèle filtre, une fois que le sous-ensemble de variables est sélectionné, il sera possible d'utiliser différents algorithmes d'apprentissages. Donc, cette approche est efficace et permet de traiter des données de grande dimension. La figure (Fig. 1.13) résume la stratégie de sélection du type filtre.

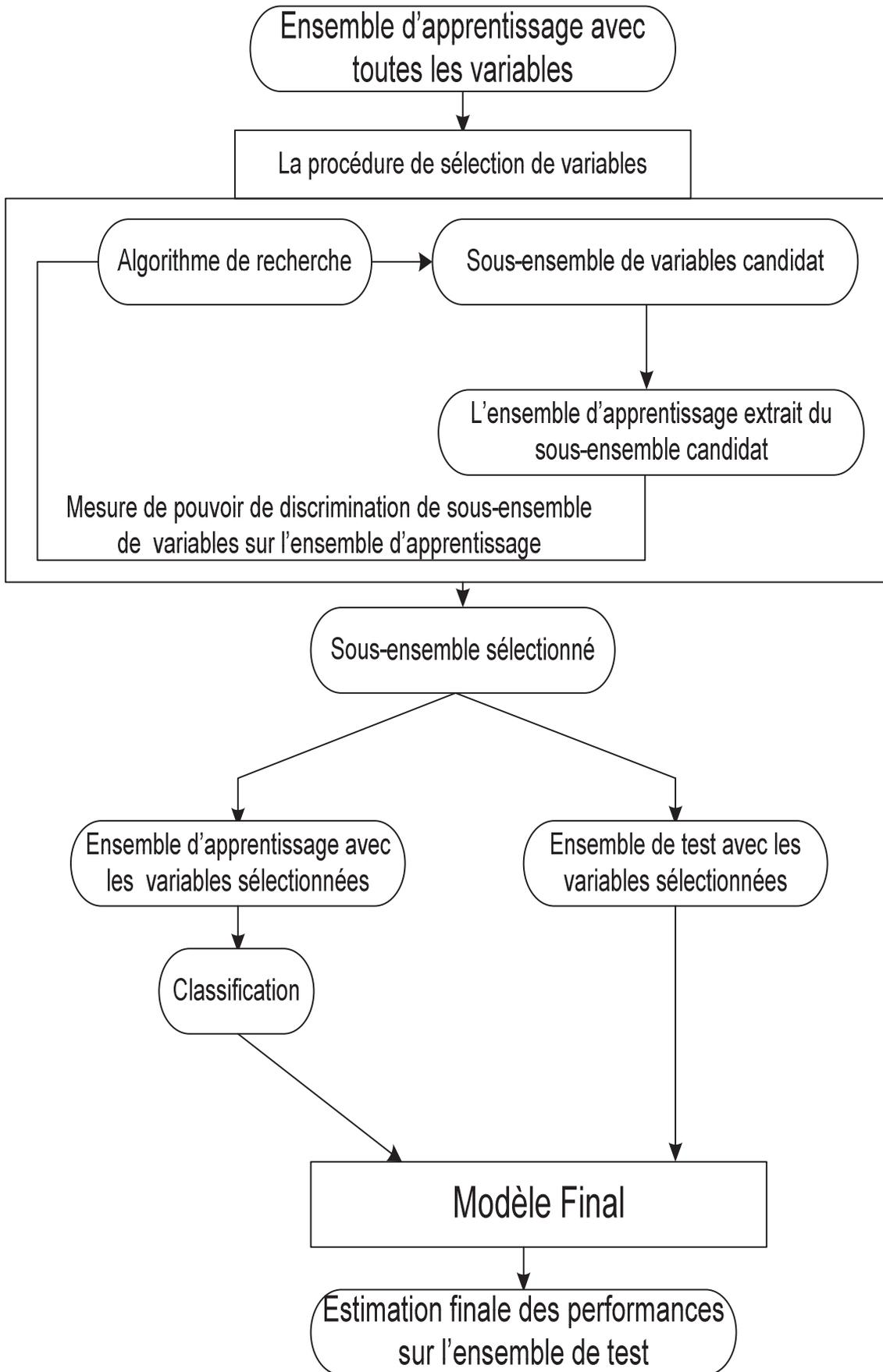


Fig. 1.11 La procédure filtre de sélection de variables

1.4.3.1.1 Méthodes basées sur la consistance

L'idée générale de cette classe de méthodes est de chercher le plus petit sous-ensemble de variables possible qui vérifie la propriété de consistance. L'algorithme de sélection FOCUS, proposé par Almuallim et autres (Almuallim, et al., 1991) pour traiter des données booléennes, recherche le sous-ensemble de variables pertinentes en évaluant chaque variable, des paires, des triplettes et ainsi de suite. Cet algorithme s'arrête quand la combinaison minimale de variables est établit. Le sous-ensemble résultant permet de construire un ensemble d'apprentissage caractérisé par des classes pures, c'est-à-dire, chaque échantillon doit appartenir à une seule classe (pas du chevauchement de classes). L'algorithme d'apprentissage utilisant les arbres de décision est alors lancé sur le sous-ensemble sélectionné. L'inconvénient principal de cette approche, révélé par Caruana et autres (Caruana, et al., 1994), est l'explosion de la dimension de l'espace de recherche avec l'augmentation de nombre de variables initiales. Les auteurs proposent quelques variantes de l'algorithme FOCUS pour booster la procédure de recherche. L'une des variantes se base sur la notion d'entropie. Son principe est le suivant : soit un sous-ensemble de variables S de dimension N , l'ensemble d'apprentissage est divisé en plusieurs groupes, chaque groupe possède les mêmes valeurs que les variables de S . Supposons que p_i et n_i représentent, respectivement, les nombres d'échantillons positifs et négatifs de $i^{\text{ième}}$ groupe (positifs : appartiennent sûrement à G_i , négatifs : ambiguïté d'appartenance avec un autre groupe G_j). La formule suivante est utilisée pour évaluer chaque variable susceptible de compléter S :

$$E(S) = - \sum_{i=0}^{2^{|S|}-1} \frac{p_i + n_i}{N} \left[\frac{p_i}{p_i + n_i} \log_2 \frac{p_i}{p_i + n_i} + \frac{n_i}{p_i + n_i} \log_2 \frac{n_i}{p_i + n_i} \right] \quad (1.4)$$

Le candidat avec la plus faible valeur de $E(S)$ sera sélectionné. Une approche similaire a été développée par Liu et autres (Liu, et al., 1996) en utilisant l'algorithme LVF (Las Vegas Filter). Ils proposent de mesurer l'inconsistance de la manière suivante :

$$I(S) = \sum_{g=1}^{G_S} \frac{n_g - f_g}{n} \quad (1.5)$$

G_S est le nombre de groupes d'objets définis par les variables S , n_g le nombre d'échantillons dans le groupe g , f_g est le nombre d'échantillons dans le groupe g qui appartiennent à la classe la plus fréquente et, n est le nombre total d'échantillons de l'ensemble d'apprentissage. L'algorithme LVF est le suivant :

- Le meilleur sous-ensemble A est initialisé avec toutes les variables initiales, et $I(A)$ est calculée.
- Un sous-ensemble B est choisi aléatoirement.
- Si la cardinalité de B est inférieure ou égale à la cardinalité de A , donc $I(B)$ est calculée.
- Si $I(B) \leq I(A)$, alors $A \leftarrow B$, et nous recommençons.

Cette approche est très performante dans le cas de données bruitées. Elle peut être erronée pour des paramètres possédant une grande variance car ces paramètres contribuent largement au calcul de la consistance, mais avec une diminution du pouvoir de généralisation. Des techniques similaires ont été étudiées également par Schlimmer (Schlimmer, 1993) et Oliveira et autres (Oliveira, et al., 1992).

1.4.3.1.2 Méthodes basées sur la théorie de l'information

Ce type de méthode de type filtre utilise la mesure de la quantité d'information transmise par le sous-ensemble de variables afin de guider la procédure de recherche. Les travaux de Sheinvald et autres (Sheinvald, et al., 1992) et de Koller et Sahami (Koller, et al., 1997) font partie de ce type de méthodes.

L'idée de base de Koller et Sahami est qu'un sous-ensemble de variables est pertinent si sa fonction de distribution de probabilité est proche de la distribution de l'ensemble initial. Formellement, soit C l'ensemble des classes, X l'ensemble de variables, S un sous-ensemble de X , $x = (x_1, \dots, x_n)$ un élément de X et x_S sont les projections de X dans S . Donc, cette approche tente de trouver un sous-ensemble A tel que : la probabilité $\Pr(C|S = x_S)$ soit proche de la probabilité $\Pr(C|X = x)$. Cet algorithme débute par la totalité des variables et applique la stratégie d'élimination en arrière. A chaque étape, il élimine la variable qui minimise la distance entre la distribution de probabilité initiale et la nouvelle. Cette distance est mesurée par l'équation suivante :

$$D \left(P_r(C|X_i = x_i, X_j = x_j), P_r(C|X_j = x_j) \right) = \sum_{c \in C} p(c|X_i = x_i, X_j = x_j) \log_2 \frac{p(c|X_i = x_i, X_j = x_j)}{p(c|X_j = x_j)} \quad (1.6)$$

Les variables sont éliminés successivement jusqu'à ce que le nombre de variables désiré soit atteint.

1.4.3.1.3 Méthodes basées sur la corrélation

La sélection de variables pour la classification est fortement liée, non seulement à la corrélation des variables entre elles, mais aussi à la corrélation entre variables et les classes attribuées, comme indiqué dans les travaux de Gennari et autres (Gennari, et al., 1989). Une variable est utile dans le processus de classification si elle est fortement corrélée avec la classe attribuée. Dans ce cas, elle possède une forte probabilité de prédire correctement les valeurs de la classe. Réciproquement, une variable est redondante si ses valeurs peuvent être prédites par d'autres variables, ceci est vérifié si cette variable est fortement corrélée avec les autres variables. De telles considérations permettent de conclure qu'un sous-ensemble est pertinent s'il est composé de variables fortement corrélées avec la classe attribuée et faiblement corrélées entre elles. Dans (Hall, 2000), l'auteur propose « The Correlation-based Feature Selector Method (CFS) » qui permet de sélectionner les variables en se basant sur la corrélation entre les attributs nominaux.

1.4.3.1.4 Méthodes basées sur les approches combinatoires

Dans (Charikar, et al., 2000), les auteurs analysent le problème combinatoire suivant : soit un ensemble de variables X et, $\prod_i i = 1, \dots, n$, ses propriétés, comment sélectionner un sous-ensemble S qui conserve les propriétés de l'ensemble initial. Selon le problème étudié, la dimension de S sera maximisée ou minimisée. Les auteurs considèrent un tel problème comme un modèle fondamental de sélection de variables et proposent les variantes suivantes :

1. Sélection du sous-ensemble : X ne vérifie pas certaines propriétés Π ; identifier le plus large sous-ensemble $S \in X$ tel que la projection de X sur S satisfasse toutes les propriétés.
2. Réduction de dimension : X satisfait toutes les propriétés Π ; identifier le plus petit sous-ensemble tel que la projection de X sur S satisfasse toutes les propriétés.

1.4.3.2 Modèle du type Wrapper

La stratégie de modèle Wrapper est d'exploiter le résultat de l'algorithme d'induction pour estimer l'utilité du sous-ensemble sélectionné sur les données d'apprentissage en utilisant l'estimation de la précision du classifieur obtenu comme métrique (critère d'évaluation). Grâce à l'interaction simultanée de la méthode d'induction et de données

d'apprentissage, les approches Wrapper renvoient de meilleurs résultats que les approches filtres. Dans les méthodes Wrapper, l'algorithme de classification est souvent sollicité, de ce fait, le temps du calcul nécessaire est généralement important. Plusieurs méthodes ont été proposées pour réduire la complexité des approches Wrapper. Nous allons les résumer dans les sections suivantes.

1.4.3.2.1 Méthodes basées sur les arbres de décision

La majorité des méthodes de cette classe est basée sur des algorithmes d'induction classiques, comme les arbres de décision ID3, C4.5 et ID4.5 introduits par Quinlan (Quinlan, 1993). Parmi les travaux les plus intéressants, on peut citer les travaux de : (John, et al., 1994), (Cherkauer, et al., 1996), (Langley, et al., 1994), (Vafaie, et al., 1995) et (Caruana, et al., 1994).

John et autres (John, et al., 1994) ont été les premiers à explorer la validité des méthodes Wrapper lors de la sélection de variables. Ces auteurs qualifient une variable de fortement pertinente si la distribution de probabilité de l'ensemble des variables change quand cette variable est éliminée. Elle est faiblement pertinente, si la distribution de probabilité d'un sous-ensemble de variables quelconque change faiblement quand nous éliminons cette dernière. Toutes les variables qui n'appartiennent pas à ces deux catégories peuvent être éliminées de la base de variables sans aucune influence.

Cherkauer et autres (Cherkauer, et al., 1996) suggèrent, lors de l'utilisation d'arbres de décision, d'exploiter les algorithmes génétiques pour déterminer le sous-ensemble optimal. Leurs algorithme SET-gen se base sur la fonction de fitness suivante :

$$Fitness(X) = \left(\frac{3}{4}\right)P + \left[1 - \left(\frac{s+f}{2}\right)\right] \quad 1.7)$$

Où X est le sous-ensemble de variables, p est la précision obtenue par la validation croisée sur l'algorithme C4.5, s est la dimension moyenne obtenue par C4.5 et f est la dimension de X . Cette formule permet de chercher les sous-ensembles avec un petit arbre mais avec une grande précision.

Langley et autres (Langley, et al., 1994) étudient l'apport de l'algorithme des plus proches voisins lors de la détermination des variables non-pertinentes. Les auteurs ont démontré que la dimension des données d'apprentissage nécessaires pour une bonne classification augmente exponentiellement avec la présence de variables non-pertinentes.

CHAPITRE 1

L'algorithme proposé par ces auteurs permet d'éliminer les variables non-pertinentes et redondantes et par conséquent, de booster le processus d'apprentissage de l'algorithme standard C4.5 utilisé sur des données de simulation.

1.4.3.2 Méthodes basées sur le contexte

Dans les problèmes de classification, il est fréquent de trouver des variables pertinentes pour une partie de l'ensemble d'apprentissage et non-pertinentes pour une autre partie. De plus, la qualité de la classification est mauvaise quand il existe des variables de l'ensemble d'apprentissage qualifiées de pertinentes si elles sont combinées avec une partie d'échantillons des autres variables et de non pertinentes si elles sont combinées avec l'autre partie d'échantillons. Toutes ces complications sont traitées quand les méthodes Wrapper sont utilisées dans la stratégie d'élimination en arrière. L'élimination d'une variable utile entraîne une perte de précision lors de la validation. La méthode RC, introduite par Domingos (Domingos, 1997), la méthode proposée par Moore et autres (Moore, et al., 1992) et celle proposée par Skalak (Skalak, 1994) tentent d'exploiter ces considérations. Pour illustrer le principe de ces méthodes, nous décrivons les principales étapes de la méthode RC :

1. L'algorithme est initialisé en lançant l'algorithme de classification sur l'ensemble des variables et la précision est estimée par la validation croisée.
2. Un élément (échantillon) de l'ensemble d'apprentissage est choisi et ses proches voisins de la même classe sont identifiés. Toutes les variables dont les éléments sont différents de l'élément initial sont éliminées.
3. L'algorithme de classification est lancé avec l'ensemble réduit des variables et sa précision est estimée par la validation croisée.
4. Si la précision ne se dégrade pas par rapport à l'ensemble de variables, l'élément sera accepté, sinon, il sera sauvegardé et signalé à ne pas utiliser lors de la prochaine itération.
5. Si tous les éléments sont testés, l'algorithme est arrêté, sinon, nous revenons en 1.

Les auteurs ont réalisé plusieurs expériences sur des données artificielles et ils ont montré la contribution insuffisante de ce type d'algorithme, en particulier, quand la majorité des variables est pertinente ou non-pertinentes. De plus, quand le nombre de variables est limité, ou les données sont bruitées, les méthodes Wrapper standards convergent d'une manière suffisante vers les variables non pertinentes.

1.4.3.2.3 Méthodes basées sur les classifieurs Bayesiens

L'importance des méthodes de sélection de variables n'a pas été négligée par les experts de la classification Bayésienne. Parmi les travaux intéressants utilisant cette technique de classification, nous pouvons distinguer les travaux de Pazzani (Pazzani, 1995), qui a introduit deux méthodes se basant sur la classification Bayésienne dans un schéma Wrapper : la première utilise la stratégie en avant et la deuxième utilise la stratégie en arrière. Dans le premier cas, l'algorithme incrémente le sous-ensemble actuel d'une variable en se basant sur les performances de l'algorithme de classification. Quand une variable utile est trouvée, une variable artificielle est créée en combinant la variable sélectionnée avec le sous-ensemble actuel. Le second algorithme fonctionne de manière inverse en éliminant les variables non-pertinentes et en combinant des paires d'éléments déjà sélectionnés en un seul paramètre virtuel rassemblant les deux. Les deux méthodes permettent d'améliorer les performances du classifieur Bayésien standard.

1.4.3.2.4 Autres méthodes de types Wrapper

Comme indiqué auparavant, l'inconvénient majeur des méthodes Wrapper est le coût calculatoire très élevé. Plusieurs méthodes ont été proposées pour faire face à cet inconvénient. Moore et autres (Moore, et al., 1994) proposent une stratégie dans laquelle ils mettent en compétition différents sous-ensembles de variables. Ils utilisent la méthode « *leave-one-out cross validation* ». Si, en se basant sur cette validation, un sous-ensemble de variables est considéré incapable de converger vers un nouveau sous-ensemble meilleur que le sous-ensemble courant, alors, l'évolution de ce sous-ensemble est stoppée. Une autre contribution intéressante est celle proposée par Kohavi et autres (Kohavi, et al., 1996) en introduisant la notion des opérateurs composés et dont l'objectif est d'améliorer l'efficacité calculatoire de la stratégie de sélection. D'abord, tout changement local possible (ajout ou élimination d'une variable) est considéré comme opérateur. Ensuite, chaque opérateur est appliqué à l'ensemble actuel de variables et le résultat est évalué. Les deux opérateurs gagnants (possédant les deux meilleures précisions) seront combinés dans un seul opérateur qui sera appliqué sur l'ensemble d'apprentissage. Le nouvel ensemble d'opérateurs (l'ensemble d'opérateurs de base plus l'opérateur issu de la combinaison) sera considéré lors de la nouvelle itération sera appliqué sur l'ensemble d'apprentissage ...etc. Les auteurs démontrent que l'utilisation de cette notion permet d'identifier rapidement les variables fortement pertinentes. La figure (Fig. 1.14) présente la procédure de sélection Wrapper.

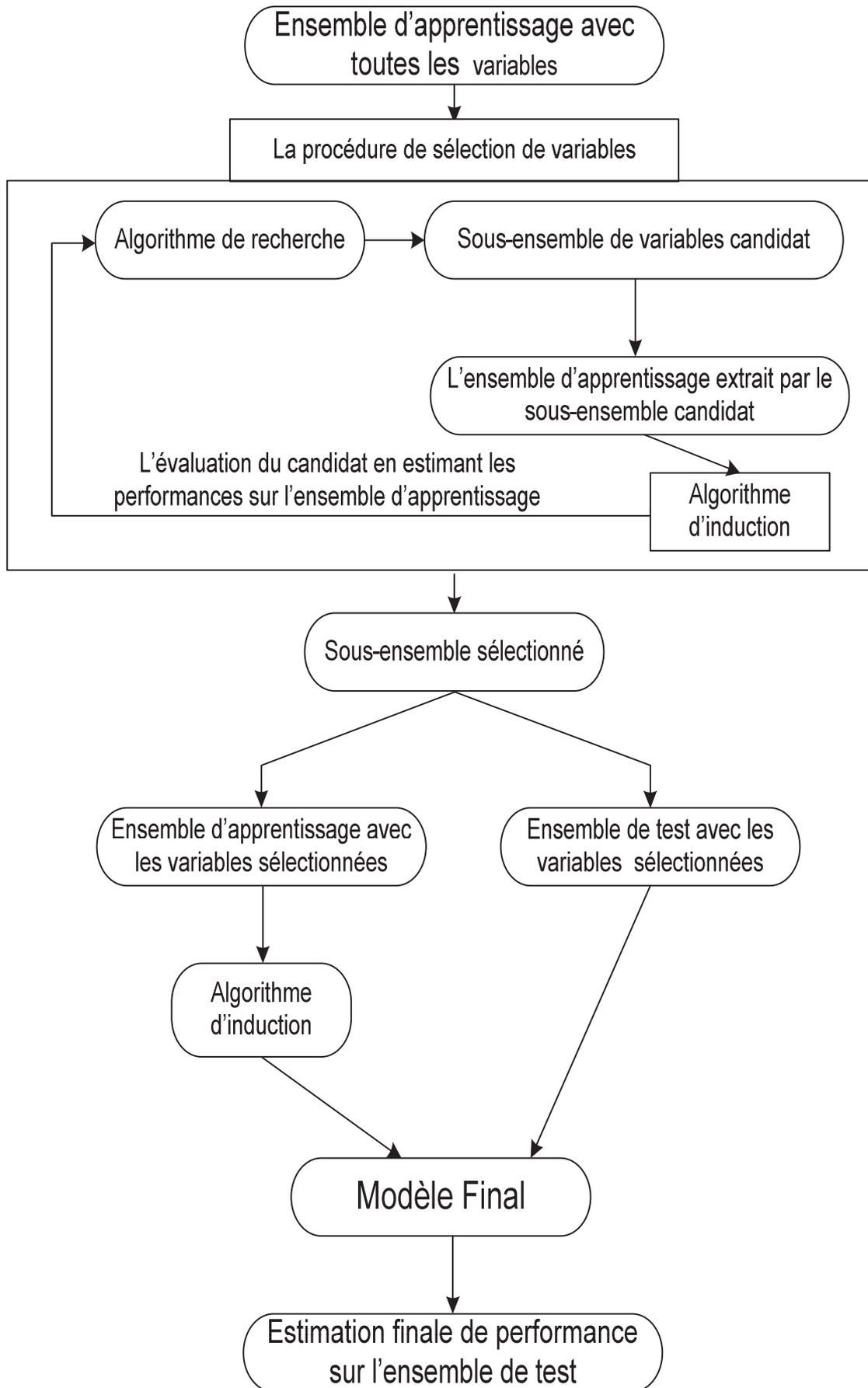


Fig. 1.12 La procédure Wrapper de sélection de variables

1.4.3.3 Modèle du type embarqué

Contrairement aux approches Wrapper, qui traitent la sélection de variables autour du processus d'induction, les approches embarquées embarquent la sélection avec l'algorithme basique d'induction. Nous pouvons citer plusieurs modèles de type embarqué comme les algorithmes ID3 et C4.5 proposés par, Quinlan (Quinlan, 1986) et (Quinlan, 1993), et l'algorithme CART proposé par Breiman et autres (Breiman, et al., 1984). Ces algorithmes d'arbre de décision utilisent les méthodes de partitionnement récursives pour l'induction et réalisent une recherche gourmande occasion par l'élargissement de l'arbre. A chaque étape, l'algorithme utilise une fonction d'évaluation pour sélectionner la variable qui possède la meilleure capacité de discrimination entre classes. Ensuite, les données de l'apprentissage sont partitionnées en se basant sur cette variable et réitère le processus pour chaque sous-ensemble en étendant l'arbre vers le bas jusqu'à ce qu'aucune discrimination ne soit possible. Outre ces trois algorithmes, il en existe un autre appelé modèle de pondération proposé par (Payne, 1999) et (Payne, et al., 1996) où une pondération est assignée à chaque variable.

Enfin, il faut signaler l'existence d'un autre modèle de sélection qui s'appuie à la fois sur les modèles Filtre et Wrapper. Cette catégorie est nommée modèle hybride (Das, 2001), (Xing, et al., 2001) et (Sebban, et al., 2002). La particularité du modèle hybride est de s'appuyer sur les méthodes de sélection citées précédemment en utilisant les différents critères d'évaluation, cités précédemment, dans plusieurs étapes de recherche du sous-ensemble optimal (Liu, et al., 2005). Ce modèle est très utilisé quand le nombre de variables devient considérable. Dans ce type de modèle nous pouvons, dans une première étape, faire une présélection en utilisant les méthodes du type filtre, afin de réduire le nombre de variables. Puis, la sélection sera affinée en appliquant l'approche du type Wrapper sur les variables synthétisées lors de la première étape.

Après avoir exposé les différents algorithmes de sélection de variables nous passons aux méthodes FDI et nous présentons un bref état de l'art de ces techniques.

1.4.4 Détection et isolation des défauts

1.4.4.1 Terminologie

Avant d'aborder cette problématique et vu la non-cohérence de la terminologie dans ce domaine, nous avons jugé utile de rappeler la terminologie qui sera utilisée dans la suite de ce manuscrit. La terminologie suivie est celle définie dans (Isermann, et al., 1997).

- **Défaut** : Un écart non acceptable d'au moins une propriété ou une variable, caractérisant le comportement, du système de son comportement nominal.
- **Défaillance** : Une interruption continue de la capacité du système à remplir une fonction requise dans des conditions d'opérations spécifiées.
- **Détection** : La détermination de la présence de défauts et de l'instant d'occurrence.
- **Isolation** : La détermination du type de défaut, la date d'occurrence et la localisation du défaut.
- **Identification** : La détermination de la taille et du comportement temporel du défaut
- **Diagnostic** : La détermination du type, de la taille, de la localisation et de l'instant d'occurrence d'un défaut (complète la détection par l'isolation et l'identification)
- **Surveillance** : Une tâche continue, réalisée en temps réel, qui consiste à déterminer l'état d'un système physique en enregistrant des informations et en reconnaissant et indiquant des anomalies de comportement.
- **Supervision** : La supervision d'un système physique et la prise de décisions adéquates pour maintenir son fonctionnement en cas d'apparition de défauts.

1.4.4.2 Positionnement du problème

L'apparition des défauts dans un procédé réel provoque la dégradation de ce dernier et la modification de ses caractéristiques dynamiques. Le défaut représente un changement anormal d'une, ou plusieurs propriétés d'un système ou de ses composants entraînant un écartement du procédé de son fonctionnement nominal. Comme un défaut ne provoque pas, dans l'immédiat, un arrêt total du système, il est essentiel de le détecter avant son amplification. La maîtrise d'un procédé passe par plusieurs étapes qui sont la détection, l'isolation, l'identification. La détection et l'isolation sont considérées comme primordiales. Elles sont regroupées, en littérature, dans une communauté de chercheurs qui s'appelle FDI (Fault Detection and Isolation).

Un défaut peut apparaître sur les actionneurs, sur les capteurs, sur le procédé ou sur un de ses composants. La figure suivante représente le schéma général d'un système de détection de défaut :

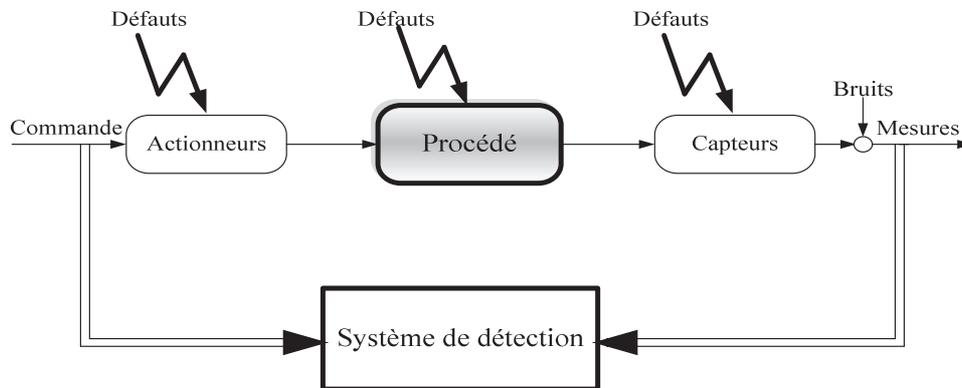


Fig. 1.13 Modèle d'un système de détection de défauts

Un système de détection de défauts se base sur les connaissances en provenance du procédé afin d'anticiper les différents défauts pouvant nuire à son fonctionnement (performances et sécurité). Ces connaissances peuvent être des signaux de commande ou des mesures ...etc. Un des défis majeurs de la détection de défauts est de distinguer les vrais défauts des autres entrées, et d'éviter, ou de limiter, les fausses alarmes.

Nous pouvons résumer les propriétés d'un système de détection performant en :

- **La robustesse** : Un système de détection performant doit être capable de détecter la présence d'un défaut en émettant le moins possible de fausses alarmes.
- **La vitesse de détection** : Un système de détection performant doit détecter des défauts dans des délais courts devant la dynamique du système.
- **La sensibilité** : Un système de détection performant doit être capable de détecter des défauts de très faibles amplitudes.

Il est essentiel de signaler l'existence d'un compromis entre les deux propriétés de robustesse et de sensibilité. Une grande sensibilité provoquera un nombre important de fausses alarmes.

1.4.4.3 Déteçtabilité

La déteçtabilité est le pouvoir du système de détection à déterminer la présence d'un défaut lors de son apparition. Elle est fonction de plusieurs paramètres comme les mesures, l'écart entre le défaut et l'état nominal... etc.

1.4.4.4 Isolabilité

L'isolabilité est le pouvoir du système de détection à discriminer un défaut d'un ensemble fini d'hypothèses. Elle est liée à la taille des défauts, au bruit des mesures, aux entrées externes, aux erreurs de modélisation, aux propriétés physiques du système à surveiller et à la l'étape de détection.

Dans cette thèse, nous nous intéressons aux méthodes FDI se basant sur les mesures. Nous proposons une revue de la littérature concernant ce domaine.

1.4.5 Les techniques FDI se basant sur les données en littérature

Il existe deux approches différentes : la première se base sur la technique de franchissement d'un seuil, la deuxième se consacre à l'étude et une analyse statistique ou fréquentielle des données disponibles fournies par les capteurs et actionneurs implémentés sur le procédé.

1.4.5.1 FDI par franchissement d'un seuil

Dans ces méthodes, on se base directement sur les signaux mesurés. Ces signaux seront comparés à des seuils prédéfinis par des experts. Si un ou plusieurs seuils sont franchis, nous concluons que le système est défaillant sans pouvoir définir les causes réelles de cette défaillance. Ces méthodes sont très sensibles aux bruits et, la plupart du temps, la détermination des seuils est délicate à réaliser. L'autre inconvénient de ces méthodes concerne les systèmes contrôlés en boucle fermée, où, les signaux sont maintenus à leurs valeurs de consigne par l'effet de la commande, d'où l'impossibilité de détecter un franchissement du seuil.

1.4.5.2 FDI par analyse des données disponibles

Il existe plusieurs techniques d'analyse des signaux par exemple, les fonctions de corrélation, les modèles ARMA, le spectre de fréquence ou, en utilisant quelques propriétés des signaux comme la variance, l'amplitude, la fréquence et la moyenne.

L'une des techniques la plus utilisée est l'analyse spectrale. Cette analyse se consacre d'abord à l'étude du fonctionnement normal du système où, les hautes fréquences sont liées à

la présence de bruit et les basses fréquences représentent l'évolution de l'état du système. La détection de défauts sera ensuite réalisée en scrutant la dérive des propriétés fréquentielles des signaux. Cette technique possède comme avantage la simplicité de mise en œuvre, et comme inconvénient la forte sensibilité aux bruits, en particulier quand ils apparaissent sur les fréquences propres du système. Il existe d'autres méthodes qui font appel aux fonctions d'auto-corrélation, à la décomposition en ondelettes, à la transformée de Fourier ou à la densité spectrale des signaux. Ces techniques conduisent à de meilleurs résultats dans le cas où les fréquences caractérisant les défauts possibles sont a priori connues.

Une autre approche très utilisée dans les méthodes FDI est l'analyse statistique (Basseville, et al., 1997). Elle considère que les changements des caractéristiques des signaux lors d'un défaut sont suffisamment rapides en comparaison aux dynamiques jugées lentes. L'intérêt de cette approche est la rapidité de détection des changements et non l'amplitude de ces changements. Cet avantage place cette approche en position idéale pour traiter la détection de changements graduels possédant des seuils de détection jugés faibles.

L'approche statistique s'est renforcée considérablement avec deux méthodes très utilisées en FDI. Il s'agit de la méthode d'Analyse en Composantes Principales (ACP) et la méthode des Moindres Carrés Partiels (De Veaux, et al., 1994). L'intérêt de ces méthodes réside dans la réduction de l'espace des variables en calculant des combinaisons linéaires ou non-linéaires de ces dernières. Les méthodes linéaires et non-linéaires ont été comparées dans (Genovesi, 1999). Le seul inconvénient de ces méthodes est leur complexité calculatoire, puisqu'elles proposent une combinaison linéaire ou non-linéaire des variables du système sans réduire l'espace de variables.

Enfin, il est intéressant de noter que ces méthodes d'analyse de signaux offrent une meilleure robustesse aux traitements, des bruits de mesure, à la calibration des capteurs et actionneurs et à l'incertitude en identification paramétrique.

Avant d'exposer l'approche proposée pour traiter la détection et l'isolation de défauts, il nous faudra faire un détour sur la thématique de fusion d'information, car c'est à partir de cette thématique, qui traite entre autres les conflits entre les sources d'information (capteurs) ; que nous proposons notre approche. Cette approche s'appuie en effet sur la notion de conflit/désaccord qui sera détaillée dans le deuxième chapitre.

1.5 Apprentissage automatique

1.5.1 Introduction

Un algorithme d'apprentissage peut être défini comme un algorithme qui détermine une fonction f à partir d'un ensemble de données E . L'ensemble E est appelé **ensemble d'apprentissage** (ensemble d'entraînement) et la fonction f est le **modèle**. En général, E contient sous forme de vecteurs l'information décrivant un problème quelconque à résoudre.

1.5.2 Les types d'apprentissage

Dans la littérature, il existe une multitude de techniques d'apprentissage. Nous pouvons diviser l'apprentissage en **apprentissage supervisé** et **non-supervisé**. Une autre catégorisation consiste à qualifier l'apprentissage d'**incrémental** ou **non-incrémental**. Dans les paragraphes suivants, nous donnons un bref aperçu de ces méthodes.

1.5.2.1 Apprentissage supervisé

L'apprentissage supervisé consiste à apprendre l'ensemble des données E en utilisant des cibles disponibles. Dans ce cas, E est composé de N paires d'entrées x_i et des cibles correspondantes y_i :

$$E = \{(x_i, y_i) | x_i \in X, y_i \in Y, i = 1:N\} \quad 1.8)$$

Typiquement, l'ensemble d'apprentissage est récolté en présentant l'ensemble des entrées à un groupe de personnes et en leur donnant comme tâche l'association de chacune de ces entrées à sa cible appropriée en respectant le contexte du problème à résoudre. La tâche principale d'un algorithme d'apprentissage est d'entraîner un modèle pour qu'il puisse imiter le processus d'étiquetage réalisé par un humain, c'est-à-dire, qu'il puisse prédire la cible y_i d'une entrée x_i quelconque. En plus de l'imitation de ce processus humain, les algorithmes d'apprentissage sont utilisés afin de modéliser la relation liant des paires d'entrées à des cibles issues d'un autre phénomène. L'ensemble de cibles Y dépend du type de problème à résoudre. Il existe deux types de problèmes qui sont les problèmes de classification et les problèmes de régression.

A. Classification : Dans un problème de classification, le vecteur cible Y est constitué d'un ensemble borné de classes susceptibles d'accueillir les différentes entrées. Dans le cas où x est une séquence, Y peut aussi être un ensemble de séquences de classes.

B. Régression : Dans ce cas, Y représente un ensemble de valeurs continues ou de vecteurs de valeurs continues.

1.5.2.2 Apprentissage non-supervisé

A l'inverse de l'apprentissage supervisé, l'apprentissage non-supervisé ne s'appuie sur aucune cible prédéfinie de la sortie. Par conséquent, l'ensemble d'apprentissage ne contient que les entrées X .

$$E = \{(x_i | x_i \in X, i = 1:N)\} \quad (1.9)$$

Cet ensemble ne définit pas explicitement la nature de la fonction f disponible à la sortie de l'algorithme d'apprentissage. Dans ce cas, c'est à l'utilisateur d'indiquer le problème à résoudre. A présent, voici quelques problèmes d'apprentissage non-supervisé.

A. Estimation de la densité : Dans ce type de problèmes, la fonction f doit fournir une estimation de la densité ou de la probabilité de la distribution avant de générer les éléments de E .

B. Extraction de caractéristiques : l'objectif de cette tâche est d'apprendre une nouvelle représentation $f(x)$ de l'entrée. Cette nouvelle représentation doit être plus utile et pertinente que la représentation originale. Cette représentation prend généralement la forme d'un vecteur dont les éléments, appelés '**caractéristiques**', sont calculés par le modèle f .

C. Réduction de dimensionnalité : dans ce cas, l'algorithme d'apprentissage associe au vecteur d'entrées X une représentation $f(X)$ possédant une dimension inférieure, mais, tout en conservant toute ou presque toute l'information disponible sur X .

D. Groupage (Clustering) : Dans cette tâche, l'algorithme d'apprentissage procède à un partitionnement de l'espace X en N sous-ensembles (ou groupes) G_1, \dots, G_N . Ces groupes peuvent être associés à un prototype \tilde{x}_i représentant les données groupées dans G_i . Il existe plusieurs algorithmes de groupage, néanmoins, nous nous contenterons de présenter l'algorithme le plus simple qui est l'algorithme K-means (Lloyd, 1982) (Steinhaus, 1956). Cet algorithme sera détaillé dans le chapitre 3.

CHAPITRE 1

L'algorithme des K-means est un algorithme itératif. Il est initialisé aléatoirement sur des vecteurs d'entrées X , considérés comme prototypes de départ. Il alterne entre les deux tâches suivantes :

- L'allocation de chaque vecteur d'entrées de E au prototype le plus proche suivant la distance euclidienne.
- La mise à jour des différents prototypes par la valeur moyenne des vecteurs d'entrées associés au même prototype.

1.5.2.3 Apprentissage incrémental

Dans un algorithme d'apprentissage incrémental, la base d'apprentissage n'est pas disponible a priori, puisque les exemples d'apprentissage arrivent au fil du temps. Ce type d'apprentissage est défini dans (Polikar, et al., 2001). Il doit avoir les propriétés suivantes :

- Il doit être capable d'apprendre des connaissances supplémentaires en utilisant de nouvelles données.
- Il ne doit pas exiger l'accès aux données d'origine (c'est-à-dire, les données déjà utilisées dans l'apprentissage).
- Il doit sauvegarder les connaissances déjà acquises.
- Il doit être capable d'apprendre de nouvelles classes candidates lors de l'arrivée de nouvelles données.

Il existe plusieurs algorithmes d'apprentissage incrémental, même si plusieurs d'entre eux ne satisfont pas tous les critères cités auparavant. Ces algorithmes sont divisés en apprentissage incrémental de paramètres et en apprentissage incrémental de structures et de paramètres.

A. Apprentissage incrémental de paramètres : Ce type d'apprentissage peut être assimilé à l'action d'**adaptation**. A l'initialisation de l'algorithme, l'utilisateur fournit la structure du modèle, tandis que les paramètres sont appris progressivement en fonction des données disponibles. Des algorithmes d'apprentissage incrémental sont présentés dans (Jr, et al., 2007), (Mouchere, et al.) et (Jang, 1993).

B. Apprentissage incrémental de structure et de paramètres : Ces algorithmes sont basés en général sur l'algorithme de clustering (groupage) se basant sur la théorie de résonance adaptative (ART : Adaptive Resonance Theory) qui sera détaillée au prochain chapitre. On peut citer comme exemples les modèles proposés par Carpenter et autre

(Carpenter, et al., 1992), Gary et autres (Gary, et al., 2001), Sadri et autres (Sadri, et al., 2006) et Lughofer (Lughofer, 2008).

A souligner, qu'il existe un autre type d'apprentissage qui est l'apprentissage non-incrémental. Ce type d'apprentissage, à l'inverse de l'apprentissage incrémental, ne possède pas la propriété d'adaptation.

1.6 Conclusion

Les méthodes de détection et d'isolation de pannes occupent une part importante dans les problématiques du diagnostic en général, et des systèmes mécatroniques en particulier. Dans ce chapitre, nous avons démontré l'intérêt que porte la communauté scientifique à la sélection de variables pertinentes. Parmi les méthodes les plus performantes nous retrouvons la méthode Wrapper. Leur caractère auto-évaluatif permet d'avoir une capacité d'adaptation importante et, surtout, d'en sortir des résultats très satisfaisants.

L'approche de surveillance dans laquelle nous nous situons est basée sur des méthodes quantitatives. Elle se résume en deux tâches, une tâche d'apprentissage des données nominales et un critère d'évaluation (détection et isolation) adaptatif.

CHAPITRE 1

1.7 Références

ACTIA [En ligne]. - <http://www.actia.com/fr/>.

Almuallim H et Dietterich T. G Learning with many irrelevant features [Conférence] // In Proceedings of the 9 th National Conference on Artificial Intelligence / éd. Press MIT. - Cambridge, Mass : [s.n.], 1991.

Basseville M et Nikiforov L Detection of abrupt changes - theorie and application [Revue] // Information and System Sciences. - Englewood Cliffs, N.J : [s.n.], 1997.

Bertram T [et al.] Modelling and simulation for mechatronic design in automotive systems [Revue] // Control Engineering Practice. - 2003. - Vol. 11. - pp. 179–190.

Blum A.L et Rivest R.L Training a 3-node neural network is NP-complete [Revue] // Neural Networks. - 1992. - Vol. 5. - pp. 117–127.

Bradley P. S, Fayyad U. M et Mangasarian O. L Mathematical programming for data mining: formulations and challenges [Revue] // INFORMS Journal on Computing. - 1999. - Vol. 11 (3). - pp. 217–238.

Breiman L [et al.] Classification and Regression Trees [Revue]. - [s.l.] : Belmont, CA: Wadsworth, 1984.

Carpenter G [et al.] Fuzzy ARTMAP : A neural network architecture for incremental supervised learning of analog multidimensional maps [Revue] // IEEE Transactions on Neural Networks. - 1992.

Caruana R et Freitag D Greedy attribute selection [Conférence] // In Machine Learning: Proceedings of the 11 th International Conference / éd. Kaufmann Morgan. - New Brunswick, New Jersey : [s.n.], 1994.

Charikar M [et al.] Combinatorial Feature Selection Problems [Conférence] // In Proceedings of FOCS 2000. - 2000.

Chatain Jean. N Diagnostic par système expert [Livre]. - [s.l.] : Hermes Sciences Publications, 1993. - Série Diagnostic et maintenance. - Traité des nouvelles technologies.

Chen M, Han J et Yu P Data mining: an overview from database perspective [Revue] // Knowledge and Data Engineering / éd. Trans IEEE. - 1996. - Vol. 8 (6). - pp. 866–883.

Cherkauer K. J et Shavlik J.W Growing simpler decision trees to facilitate knowledge discovery [Conférence] // In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining / éd. Press AAAI. - Portland, Oregon : [s.n.], 1996.

Chiang Leo H, Russell Evan L et Richard D Fault detection and diagnosis in industrial systems [Revue] // Springer-Verlag. - New York : [s.n.], 2001.

Chow E et Willsky A Analytical redundancy and the design of robust failure detection systems [Revue] // IEEE Transactions on Automatic Control . - 1984. - Vol. 29 (7). - pp. 603-614.

Comerford C Mecha...what ? [Revue] // IEEE Spectrum. - 1994. - pp. 46–49.

Cornuéjols A, Miclet L et Kodra Y Apprentissage artificiel : concepts et algorithmes [Revue]. - [s.l.] : Eyrolles, 2002.

Cover T.M et Hart P. E Nearest neighbor pattern classification [Revue] // IEEE Transactions on Information Theory. - 1967. - Vol. 13. - pp. 21–27.

Das S Filters, wrappers and a boosting based hybrid for feature selection [Conférence] // In Proceedings of International Conference on Machine Learning. - Williams College : [s.n.], 2001. - pp. 74-81.

Dash M et Liu H Feature Selection for Classification [Revue] // Intelligent Data Analysis. - 1997. - Vol. 1 (3).

De Veaux R. D, Ungar L. H et Vinson J. M Statistical approaches to fault analysis in multivariate process control [Conférence] // American Control Conference proceedings . - 1994.

Deirdre W et Sperber D Relevance Theory [Livre]. - Oxford , Blackwell : L. Horn & G. Ward/ The Handbook of Pragmatics, 2004. - pp. 607-632.

Denoeux T, Masson M et Dubuisson B Advanced pattern recognition techniques or system monitoring and diagnosis : A survey [Revue] // Journal European des Systemes Automatisés. - 1997. - Vol. 31 (9-10) :. - pp. 1509–1539.

Desjardin L A day in the life of mechatronic engineers 10 years from now [Conférence] // In SAE International Congress and Exposition. - Detroit/Michigan : [s.n.], 1996.

Domingos P Context-sensitive feature selection for lazy learners [Revue] // Artificial Intelligence Review. - 1997. - Vol. (11). - pp. 227-253.

Dreyfus G [et al.] Réseaux de neurones : Méthodologie et applications [Revue]. - [s.l.] : Eyrolles, 2ème édition, 2004.

Duda R. O, Hart P. E et Stork D. G Pattern Classification 2nd edition [Livre]. - [s.l.] : Wiley, 2001.

Faucher J Pratique de l'AMDEC [Livre]. - [s.l.] : Dunod, 2004.

Friedman M et Kandel A Introduction to pattern recognition: Statistical, structural, neural, and fuzzy logic approaches [Livre]. - River Edge, NJ : World scientific, 1999.

Friedman N, Geiger D et Goldszmidt M Bayesian network classifiers. Machine Learning [Revue]. - 1997. - Vol. 29 (2-3). - pp. 131–163.

Fukunaga K Introduction to Statistical Pattern Recognition [Livre]. - New York, NY : Academic Press, 1972.

Gary G et Yen P. M An effective neuro-fuzzy paradigm for machinery condition health monitoring [Revue] // IEEE Transactions on Systems, Man, and Cybernetics. - 2001. - Vol. 31 (4). - pp. 523 - 536.

Gennari J. H, Langley P et Fisher D Models of incremental concept formation [Revue] // Artificial Intelligence. - 1989. - Vol. 40. - pp. 11-61.

CHAPITRE 1

Genovesi A. M Détection des défauts et diagnostic en ligne de procédés biologiques de dépollution [Rapport] : Thèse de doctorat / Université de Perpignan. - 1999.

Gertler J. J Analytical redundancy methods in fault detection and isolation : survey and synthesis [Conférence] // IFAC symposium SAFEPROCESS'91. - Baden-Baden : [s.n.], 1991. - pp. 9-21.

Grimheden M et Hanson M What is mechatronics ? proposing a didactical approach to mechatronics [Conférence] // 1st Baltic Sea Workshop on Education in Mechatronics. - Kiel, Germany : [s.n.], 2001.

Hall M. A Correlation-based Feature Selection for Machine Learning [Conférence] // In Proceedings of the 17 th International Conference on Machine Learning / éd. Pubshers Morgan Kaufmann. - Stanford University, C.A : [s.n.], 2000.

Hall Mark A. Correlation-Based Feature Selection for Machine Learning [Rapport] / University of Waikato ; Department of Computer Science. - Hamilton, New Zealand : [s.n.], 1999.

Hewit J. Mechatronics design - the key to performance enhancement [Revue] // Robotics and Autonomous Systems. - 1996. - Vol. 19:. - pp. 135–142.

Isermann R et Ballé P Trends in the application of model-based fault detection and diagnosis of technical processes [Revue] // Control Engineering Practice. - 1997. - Vol. 5 (5). - pp. 709-719.

Isermann R, Schwarz R et Stolzl S Fault-tolerant drive-by-wire systems [Revue] // Control Systems Magazine. - 2002. - Vol. 22 (5). - pp. 64-81.

Isermann R. Mechatronic systems - innovative products with embedded control [Revue] // Control Engineering Practice. - 2007. - p. 10 :16.

Jain A et Zongker D Feature selection: Evaluation, application, and small sample performance [Revue] // Pattern Analysis and Machine Intelligence / éd. Trans IEEE. - 1997. - Vol. 19(2). - pp. 153–158.

Jain A.K et Chandrasekaran R Dimensionality and sample size consideration in pattern recognition practice [Revue] // In P.R. Krishaniah and L.N. Kanal. - North-Holland, Amsterdam, The Netherlands : Handbook of Statistics, 1982. - Vol. 2. - pp. 835–855.

Jang J. S ANFIS : adaptive-network-based fuzzy inference system [Revue] // Systems, Man and Cybernetics / éd. on IEEE Transactions. - 1993. - Vol. , vol. 23, p. 665-685.

John G.H, Kohavi R et Pflieger K Irrelevant features and the subset selection problem [Conférence] // In Proceedings of the Eleventh International Conference on Machine learning / éd. Kaufmann Morgan. - New Brunswick, NJ : [s.n.], 1994. - pp. 121–129.

Jr J. J. L et Zeleznik R. C A Practical Approach for Writer-Dependent Symbol Recognition Using a Writer-Independent Symbol Recognizer [Revue] // Transactions on Pattern Analysis and Machine Intelligence. - 2007. - Vol. 29 (11). - pp. 1917-1926.

Kittler J Feature set search algorithm., editor, Pattern Recognition and Signal [Revue] // In C.H. Chen / éd. Processing Pattern Recognition and Signal. - Sithof and Noordhoff, Alphen aan den Rjin, The Netherlands : [s.n.], 1978. - pp. 41–60.

Kohavi R et John G Wrappers for feature subset selection [Revue] // Artificial Intelligence. - 1997. - Vol. 97 (1-2). - pp. 273–324.

Kohavi R et John G Wrappers for feature subset selection [Revue] // Artificial Intelligence, special issue on relevance. - 1996. - Vol. 97 (1-2). - pp. 273-324.

Koller D et Sahami M Hierachically classifying documents using very few words. [Conférence] // In Machine learning: Proceedings of the 14 International Conference. - 1997.

Kortum W., Goodall R. M et Hedrick J. K Mechatronics in ground transportation-current trends and future possibilities [Revue] // Annual Reviews in Control. - 1998. - Vol. 22 . - pp. 133–144.

Kudo M et Sklansky J Comparison of algorithms that select features for pattern classifiers [Revue] // Pattern Recognition. - 2000. - Vol. 33 (1). - pp. 25-41.

Langley P et Sage S Scaling to domains with irrelevant features [Revue] // Computational Learning Theory an Natural Learning Systems / éd. Greiner In R.. - 1994. - Vol. volume 4, . MIT Press.

Langley P Selection of relevant features in machine learning [Conférence] // In Proceedings of the AAAI Fall Symposium on Relevance. AAAI Press. - 1994.

Lebart L, Morineau A et Piron M Statistique exploratoire multidimensionnelle [Livre]. - [s.l.] : DUNOD, 2000.

Liu H et Setiono R A probabilistic approach to feature selection: A filter solution [Conférence] // In Machine learning: Proceedings of the 13 th International Conference on Machine Learning / éd. Kaufmann Morgan. - 1996.

Liu H et Yu L Toward integrating feature selection algorithms for classification and clustering [Revue] // IEEE Transactions on Knowledge and Data Engineering. - 2005. - Vol. 17 (4). - pp. 491-502.

Lloyd S. P Least squares quantization in pcm. Information Theory [Revue] // IEEE Transactions on. - 1982. - Vol. 28(2). - pp. 129–137.

Lughofer E FLEXFIS : A Robust Incremental Learning Approach for Evolving TakagiSugeno Fuzzy Models [Conférence] // IEEE Transactions on Fuzzy Systems. - 2008. - Vol. 16 (6). - pp. 1393-1410.

Manabu I et Sklansky J Optimum feature selection by zero-one integer programming [Revue] // Systems, Man, and Cybernetics / éd. Trans. IEEE. - 1984. - Vol. SMC-14 (5).

Mihalache A. G MODELISATION ET EVALUATION DE LA FIABILITE DES SYSTEMES MECATRONIQUES : APPLICATION SUR SYSTEME EMBARQUE [Rapport] : Thèse / ÉCOLE DOCTORALE D'ANGERS. - Angers : [s.n.], 2007.

Millbank J Mecha-what ! [Revue] // Mechatronics Forum Newsletter. - 1993. - Vol. 6.

Miller A.J Subset Selection in Regression [Livre]. - Washtington D.C. : Chapman and Hall, 1990.

Mocciardi M A comparison of seven techniques of choosing subsets of pattern recognition [Revue] // Computers / éd. Trans IEEE. - 1971. - Vol. C-20. - pp. 1023–1031.

CHAPITRE 1

- Mocciardi M** A comparison of seven techniques of choosing subsets of pattern recognition. [Revue] // IEEE Trans. Computers. - 1971. - Vol. 20. - pp. 1023–1031.
- Moore A. W et Lee M. S** Efficient algorithms for minimizing cross validation error [Conférence] // In Machine learning: Proceedings of the 11 th International Conference / éd. Kaufmann Morgan. - 1994.
- Moore A. W, Hill D. J et Johnson M. P** Computational Learning Theory and Natural Learning Systems [Revue]. - [s.l.] : MIT Press, 1992. - Vol. 3.
- Mouchere H, Anquetil E et Ragot N** On-line writer adaptation for handwriting recognition using fuzzy inference systems [Revue] // International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI). - 2007 : [s.n.]. - Vol. 21 (1). - pp. 99-116.
- Narendra P et Fukunaga K** A branch and bound algorithm for feature subset selection. [Revue] // IEEE Trans. Computer. - 1977. - Vol. 26(9). - pp. 917–922.
- Oliveira A. L et Vincetelh A. S** Constructive induction using a non-greedy strategy for feature selection [Conférence] // In Proceedings of the 9 th International Conference on Machine Learning / éd. Kaufmann Morgan. - Aberdeen, Scotland : [s.n.], 1992. - pp. 355-360.
- Ollero A. [et al.]** Mechatronics, robotics and components for automation and control : Ifac milestone report [Revue] // Annual Reviews in Control. - 2006. - Vol. 30(1). - pp. 41–54.
- Payne Terry R** Dimension Reduction and Representation for Nearest Neighbor Learning [Rapport] : PhD Thesis / University of Aberdeen ; Department of Computing Science. - 1999.
- Payne Terry R et Edwards P** Survey of work on feature selection [Revue] // Draft Copy Only. - 1996.
- Pazzani M** Searching for dependencies in Bayesian classifiers [Conférence] // In Proceedings of the 5 th International Workshop on AI and Statistics. - 1995.
- Pillet Maurice** Appliquer la maîtrise statistique des procédés MSP/SPC 3ème édition [Livre]. - [s.l.] : Les Editions d'Organisation, 2001.
- Polikar R [et al.]** Learn++ : An Incremental Learning Algorithm for Supervised Neural Networks [Revue] // IEEE Transactions on Systems, Man, and Cybernetics. - 2001. - Vol. 31. - pp. 497-508.
- Porter B, Bareiss R et Holte R** Concept learning and heuristic classification in weak-theory domains [Revue] // Artificial Intelligence. - 1990. - Vol. 45 (1-2). - pp. 229-263.
- Poullard H** Statistique et Réseaux de Neurones pour un Système de Diagnostic : Application au Diagnostic de Pannes Automobiles [Rapport] : Thèse / Université Paul Sabatier. - Toulouse : [s.n.], 1996.
- Pudil P, Novovicova J et Kittler J** Floating search methods in feature selection [Revue] // Pattern Recognition Letters. - 1994. - Vol. 15. - pp. 1119–1125.
- Quinlan J. R** C4'5: Programs for Machine Learning [Revue]. - San Mateo, CA : Morgan Kaufmann, 1993.
- Quinlan J. R** Induction of decision trees [Revue] // Machine Learning. - 1986. - Vol. 1. - pp. 81–106.

- Quinlan J. R** Programs for Machine Learning [Revue]. - San Marteo, CA : Morgan Kaufmann, 1993.
- Rzevski G.** On conceptual design of intelligent mechatronic systems [Revue] // Mechatronics. - 2003. - Vol. 13(10). - pp. 1029–1044.
- Sadri J, Suen C. Y et Bui T. D** A New Clustering Method for Improving Plasticity and Stability in Handwritten Character Recognition Systems [Conférence] // International Conference on Pattern Recognition. - 2006. - Vol. 2. - pp. 1130-1133.
- Saviuc V** La mécatronique intelligente. J [Revue] // LE JOURNAL D'INFORMATION TECHNOLOGIQUE (Jitec). - 2006.
- Schlimmer J. C** Efficiently inducing determinations: A complete and systematic search algorithm that uses optimal pruning [Conférence] // In Proceedings of the 10 th International Conference on Machine Learning / éd. Kaufmann Morgan. - Amherst, MA : [s.n.], 1993. - pp. 284-290.
- Sebban M et Nock R** A hybrid filter / wrapper approach of feature selection using information theory [Revue] // Journal of Pattern Recognition. - 2002. - Vol. 35 (4). - pp. 835-846.
- Setiono R** Neural network feature selector [Revue] // Neural Networks / éd. Trans IEEE. - 1997. - Vol. 8 (3). - pp. 654–662.
- Sheinvald J, Dom B et Niblack W** Unsupervised image segmentation using the minimum description length principle [Conférence] // In Proceedings of the 10 th International Conference on Pattern Recognition. - 1992.
- Shetty D et Kolk R** Mechatronics System Design [Revue] // PWS Publishing Company. - 1997.
- Siedlecki W et Skansky J** On automatic feature selection [Revue] // International Journal of Pattern Recognition and Artificial Intelligence. - 1988. - Vol. 2. - pp. 197–220.
- Skalak D. B** Prototype and feature selection by sampling and random mutation hill climbing algorithms [Conférence] // In Machine Learning: Proceedings of the 11 th International Conference / éd. Kaufmann Morgan. - 1994.
- Sperber D et Wilson D** La pertinence : communication et cognition (Relevance : Communication and cognition, 1985) [Livre]. - [s.l.] : Editions de Minuit, 1989.
- Stearns S.D** On selecting features for pattern classifiers [Conférence] // In Proceedings of the 3rd International Conference on Pattern Recognition. - Coronado, CA : [s.n.], 1976. - pp. 71–75.
- Steinhaus H** Sur la division des corp matériels en parties [Revue] // . Bulletin L'Académie Polonaise des Sciences. - 1956. - Vol. 4. - pp. 801–804.
- Thesame** La mécatronique à l'épreuve du marché [Revue] // LE JOURNAL D'INFORMATION TECHNOLOGIQUE (Jitec). - 2003.
- Vafaie H et De Jong K** Genetic algorithms as a tool for restructuring feature space representations [Conférence] // In Proceedings of the International Conference on Tools with A.I / éd. Press IEEE Computer Society. - 1995.

CHAPITRE 1

Vapnik V. N The Nature of Statistical Learning Theory [Revue] / éd. Springe. - 1995.

Venkatasubramanian V, Rengaswamy R et Kavuri S. N A review of process fault detection and diagnosis part iii : Process history based methods [Revue] // Computers and Chemical Engineering. - 2003. - Vol. 27 (3). - pp. 327–346.

Xing E. P, Jordan M. I et Karp R. M Feature selection for highdimensional genomic microarray data [Conférence] // In International Conference on Machine Learning. - 2001. - pp. 601-608.

Yaskawa-Electric C [En ligne]. - 1969. - [http ://www.yaskawa.co.jp/encompany/rekisi.htm](http://www.yaskawa.co.jp/encompany/rekisi.htm).

Zwingelstein G Diagnostic des défaillances, théorie et pratique pour les systèmes industriels [Livre]. - [s.l.] : HERMES, 1995.

“Don’t implement solutions.

Prevent Problems”

Dans ce chapitre, une nouvelle approche de détection et d’isolation de fautes est introduite. Après avoir précisé la problématique traitée dans cette thèse, nous allons proposer notre système FDI qui nécessite quelques propriétés fondamentales et se divise en deux étapes : apprentissage et évaluation. Concernant le critère d’évaluation, nous nous sommes inspirés de la théorie des fonctions de croyance et, en particulier, du conflit de Dempster. Enfin, un exemple d’application est donné et le choix de l’algorithme d’apprentissage est fixé.

2 Détection et isolation de défauts par projection : Notion du désaccord :

2.1 Introduction

L’augmentation du nombre de capteurs permettant de communiquer l’état des systèmes mécatroniques en temps réel a engendré des quantités d’informations colossales. Ces informations nécessitent plusieurs étapes de traitement avant de pouvoir être utilisées dans la maîtrise des procédés, notamment, en détection et isolation de défauts.

Après avoir présenté, dans le chapitre précédent, la terminologie à suivre dans ce manuscrit, maintenant, nous présentons les différentes approches FDI existantes. La détection et l’isolation des défauts est un problème crucial traité avec différentes approches résumées dans les articles (Willsky, 1976) et (Frank, 1990) et les deux livres (Patton, et al., 1989) et (Basseville, et al., 1993). Ces deux étapes sont considérées parmi les étapes les plus importantes conduisant à une surveillance performante du système. Dans ses premières applications, la FDI a été réservée à la commande et la surveillance des systèmes complexes dont nous pouvons citer, les centrales nucléaires, les applications aérospatiales ...etc. Comme indiqué dans le chapitre précédent, un système de surveillance doit impérativement caractériser les différents modes de fonctionnement du système mécatronique, en particulier, indiquer la présence d’une anomalie. Ainsi, les fonctions de surveillance renseignent l’utilisateur du procédé sur l’apparition d’états indésirables sur ce dernier. Cette tâche est de

plus en plus délicate à cause de la problématique soulevée au paragraphe précédent sur les quantités d'informations importantes.

Dans la plupart des cas, les variables représentant l'évolution du système ne sont pas toutes sensibles et/ou impliquées dans l'apparition des défauts de fonctionnement. Comme indiqué précédemment, quelques travaux ont montré que, dans plusieurs applications, plus de la moitié des variables disponibles ne sont pas importantes (voire inutiles) à l'exploitation du procédé et en particulier à la détection et l'isolation des défauts. Par conséquent, une sélection de ces variables, en tenant compte de leurs pertinences, est plus que bénéfique à l'opération de détection et d'isolation des défauts.

Il existe plusieurs méthodes de réduction de la dimension de l'espace des variables (citées au premier chapitre). Ces méthodes permettent de projeter l'espace des variables entières dans des sous-espaces très réduits, mais, plus performants et utiles à la surveillance du système. Dans la détection et l'isolation des défauts, cette tâche permettra d'accélérer le processus d'identification et, par conséquent, d'anticiper les actions nécessaires et adéquates pour remédier à ces défauts.

2.2 Problème traité et objectifs de la thèse

Le chemin parcouru par les méthodes de détection et d'isolation de défauts est déjà important, en particulier, dans le cas des méthodes basées sur les données. Les questions qui se posent sont :

- Comment proposer un système FDI à base de données génériques, c'est-à-dire, applicable à toutes sortes de défauts et d'applications ?
- Comment proposer un système FDI robuste et facilement interprétable ?
- Comment proposer un système FDI se basant sur un modèle appris valable pour n'importe quelle dimension du système ?
- Comment proposer un critère de choix simple (complexité de calcul réduite) et efficace (performant en détection et isolation) ?

Dans cette thèse, nous proposons un système FDI réalisé en deux étapes essentielles : **l'apprentissage des données nominales** du système et la détection et l'isolation en utilisant un **critère d'évaluation**. Le système FDI proposé est résumé dans la figure (Fig. 2.1).

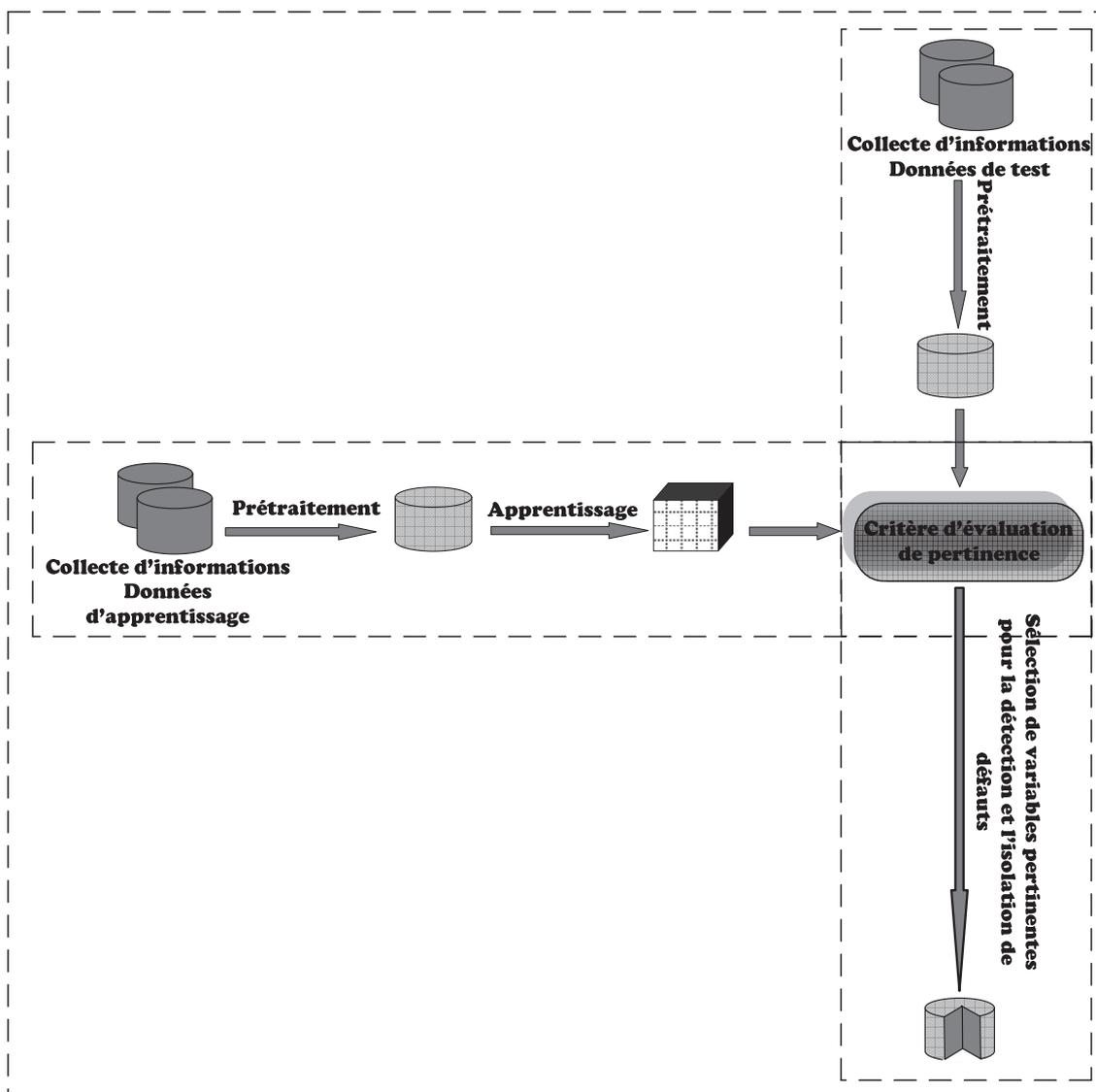


Fig. 2.1 Système FDI proposé

Concernant l'étape d'apprentissage, nous avons cherché dans la littérature afin d'utiliser un algorithme d'apprentissage robuste, adaptatif et facilement interprétable.

Dans le développement de ces propriétés que doit vérifier notre système FDI, nous commençons par « **facilement interprétable** » : il s'agit de trouver un algorithme d'apprentissage qui fournit des formes interprétables par l'utilisateur. L'auteur (Salzberg, 1989), a proposé un algorithme dans lequel il traite le problème de la représentation des connaissances acquises pendant et à la fin d'un processus. Salzberg rapporte que plusieurs systèmes d'apprentissage renvoient des lois, souvent, sous forme logique. Ces systèmes d'apprentissage sont confrontés à la difficulté de généraliser ces lois pour une utilisation globale sur des données quelconques. Selon Salzberg encore, il existe également un autre

moyen de représenter l'information en utilisant les arbres de décision. Mais ces arbres possèdent l'inconvénient de ne pas bien représenter les connaissances, ce qui rend les sorties de ces algorithmes trompeuses, notamment quand l'effet négatif d'une variable est caché par les autres variables. Une autre représentation de connaissances consiste à créer des schémas et à les généraliser (Mooney, et al., 1985).

Selon Salzberg, en plus des formes présentées précédemment, il existe d'autres formes de présenter des connaissances. Dans l'algorithme qu'il propose, EACH (Exemplar-Aided Constructor of Hyper-rectangles), il introduit une architecture qui crée un espace mémoire dans lequel les exemplaires appris sont stockés. Une partie de ces exemplaires représente des généralisations alors que d'autres représentent des exemples individuels.

Cette structure de mémoire d'exemplaires est la principale sortie de l'algorithme d'apprentissage qu'il propose. Les exemplaires généralisés possèdent une forme d'hyper-rectangles. En plus de ça, EACH modifie ses métriques, qu'elle utilise pour calculer les distances entre les exemplaires, en utilisant le feedback retourné de la prédiction. La caractéristique importante, selon Salzberg, des hyper-rectangles appris par EACH est qu'un hyper-rectangle peut être encastré à l'intérieur d'un autre. L'hyper rectangle le plus profond représente une exception, qui a été jusqu'à présent, une des difficultés de l'apprentissage automatique.

Un autre travail soulignant l'importance de l'utilisation des hyper-rectangles dans la représentation des connaissances est le travail de Byron Ju Gao (Byron, 2007), dans sa thèse il rapporte que l'objectif de la fouille des données est d'extraire des connaissances utiles. Ces connaissances doivent posséder, idéalement, des formes compréhensibles par l'utilisateur qui, au travers de ces connaissances, peut avoir des idées et des intuitions concernant le problème à résoudre et prendre les décisions judicieuses. Dans sa thèse, consacrée à l'étude de la généralisation discriminative de données à base de rectangles, Byron rapporte que les hyper-rectangles possèdent cette propriété. Il a, notamment, mis l'accent sur la description des groupes (Clusters), les règles d'apprentissage ainsi que la classification se basant sur les plus proches rectangles.

La deuxième propriété que nous cherchons à avoir dans l'algorithme d'apprentissage est l'**adaptation**. Cette propriété est liée à la stratégie utilisée dans l'apprentissage. Comme nous l'avons souligné précédemment, il existe deux types d'apprentissage, incrémental et non-incrémental. Dans son article cité précédemment, Salzberg fait une description de ces

CHAPITRE 2

deux stratégies. Il rapporte qu'un apprentissage incrémental doit posséder des exemplaires qui se généralisent facilement. Il rapporte, en particulier que les algorithmes qui fournissent des ellipsoïdes n'ont pas la possibilité d'inclure d'autres formes ou de modifier leurs distances. Les algorithmes d'apprentissage non-incrémentaux possèdent une stratégie qui permet un accès aléatoire aux exemples de l'ensemble d'apprentissage, ce qui permet de rechercher dans l'ensemble d'apprentissage des formes et des règles pour formuler des lois et des arbres de décisions. De ce fait, ces algorithmes examinent plusieurs fois l'ensemble d'apprentissage avant d'élaborer leurs sorties. Leur avantage est de ne pas être sensible à l'ordre de présentation des données, mais ils ne possèdent pas un critère d'arrêt et ils souffrent du manque d'efficacité : pour ajouter un exemple au modèle d'exemplaires nous devront refaire le traitement de tous les exemples appris précédemment.

Un algorithme d'apprentissage doit explorer les données une seule fois afin de réduire sa complexité calculatoire. Il doit également être capable d'apprendre à la fois des données binaires et des données analogiques. Cette dernière propriété, soulignée par Salzberg, nécessite un algorithme d'apprentissage avec des concepts disjonctifs : à chaque exemple qui arrive, la catégorie qui lui correspond lui est associé.

Pour conclure, nous devons utiliser un algorithme qui fournit des catégories facilement interprétables qui possède la capacité de généralisation (incrémental), et doit traiter, dans son exploration, les données une et une seule fois. C'est pour ça que nous avons opté pour une représentation par hyper-rectangle. De plus, ce type de représentation va nous permettre d'introduire un critère d'évaluation performant en détection et isolation de défauts. Ce critère sera détaillé dans les paragraphes suivants.

Dans le paragraphe suivant, nous allons présenter la thématique de fusion d'information, dans le but de faire un lien avec les techniques FDI. Plus précisément, nous allons nous intéresser à la notion de conflit entre sources d'information. C'est une notion très forte qui permet de sélectionner les sources pertinentes pour la surveillance du système.

2.3 Conflit/désaccord en fusion d'information

2.3.1 Liens entre fusion d'information et FDI

La fusion d'information (Bloch, et al., 2001) (Bloch, 2001) (Bloch, 2003) a été initialement développée afin de gérer des quantités très importantes de données multi-sources

dans le domaine militaire et en particulier en fusion d'images satellitaires. Cette discipline a beaucoup évolué ces dernières années dans différents domaines, en particulier en vision et en robotique, et les sources d'informations se sont multipliées, qu'il s'agisse de signaux analogiques délivrés par les capteurs, d'informations à priori, de connaissances génériques, etc... Chaque source d'information étant en général imparfaite, il est important d'en combiner plusieurs afin d'avoir une meilleure connaissance du système. La fusion d'informations peut alors se définir comme la combinaison d'informations (souvent imparfaites et hétérogènes) afin d'obtenir une information globale plus complète, de meilleure qualité, et permettant de mieux décider et agir. Parallèlement, les méthodes pour modéliser les connaissances et informations imparfaites et pour les combiner ont connu des développements théoriques importants, et leurs champs d'applications se sont étendus. Ces méthodes, souvent issues des théories de la décision, de l'incertain et de l'intelligence artificielle, peuvent être numériques ou symboliques.

L'objectif de la théorie de l'information est l'étude de l'information (en particulier de la quantité d'information) apportée par un message, un signal ou une source d'information donnée.

La théorie de l'information s'est développée en s'appuyant essentiellement sur la notion d'entropie. Issue de la thermodynamique, cette notion s'est étendue à d'autres domaines de la physique et surtout de l'informatique pour devenir une notion fondamentale de la théorie de l'information. Shannon (Shannon, 1948), Fisher (Fisher, 1922) et Hartley (Hartley, 1928) sont certainement ceux qui ont le plus contribué à fonder les bases modernes de la théorie de l'information.

En physique, l'entropie désigne le degré de désordre dans la matière. On parle également de la dispersion des particules dans un volume donné. Dans la théorie de l'information, l'entropie (nommée aussi entropie de Shannon : après l'accord des autres scientifiques de son époque) est liée à la dispersion de l'information dans un message allant d'un émetteur à un récepteur. On parle aussi du degré d'incertitude concernant la prévision d'un résultat. Cette incertitude est une incertitude probabiliste, c'est-à-dire qu'elle exprime la probabilité d'apparition d'événements dans le message transmis.

A partir de la notion d'entropie relative, la théorie de l'information permet d'évaluer l'indépendance générale entre deux variables. Cette indépendance ne dépend plus des propriétés des signaux eux-mêmes, mais est basée sur les probabilités calculées de ces signaux.

CHAPITRE 2

La théorie de l'information peut donc être utilisée pour évaluer l'indépendance entre les signaux.

L'ampleur que prend la fusion d'information suit celle que prennent les technologies et le traitement de l'information en général. La difficulté majeure de cette discipline est la sélection des sources pertinentes pour une application donnée. La même problématique se retrouve dans un processus de diagnostic, et en particulier, en détection et isolation de défauts.

2.3.2 Théorie des fonctions de croyance

2.3.2.1 Introduction

La théorie des fonctions de croyance a été introduite par Dempster en 1968 (Dempster, 1968), puis développée sous un formalisme mathématique plus abouti par Shafer (Shafer, 1976). Elle est également connue dans la littérature sous le nom de « théorie de l'évidence » « théorie des plausibilités » et surtout « théorie de Dempster-Shafer »

Cette théorie est souvent décrite comme une généralisation de l'inférence Bayésienne au traitement de l'incertain, dans le sens où elle permet de manipuler des événements non nécessairement exclusifs. Cette capacité lui réserve l'avantage de représenter explicitement l'incertitude sur des événements. Elle prend en compte ce qui reste inconnu et représente parfaitement ce qui déjà connu. La littérature dans ce domaine est très abondante, l'ouvrage de référence écrit par Shafer reste à l'heure actuelle le document le plus clair et le plus complet en ce qui concerne les concepts mathématiques.

De plus, la théorie des fonctions de croyance permet de combiner des sources d'informations plus ou moins disparates pour produire un résultat informatif sur le système étudié et éventuellement prendre une décision qui s'impose.

2.3.2.2 Nature du problème à traiter

On peut considérer le problème général qui consiste à identifier un objet ou une situation, désigné par la variable générique x , par un ensemble d'hypothèses H_i dont l'une d'elle est susceptible d'être la solution. Par exemple, il peut s'agir d'identifier la maladie d'un patient (diagnostic médical), de reconnaître une cible aérienne, d'estimer un paramètre, etc...

La théorie de l'évidence permet alors d'évaluer la véracité des propositions. Pour ce faire, elle se fonde sur le degré de croyance pour représenter l'incertitude des différentes propositions. Notons que l'évaluation réalisée est de nature subjective, ce qui signifie qu'elle s'établit selon notre état de connaissance. En effet, elle est essentiellement conditionnée par l'ensemble des informations issues d'une ou plusieurs sources.

Par la suite, nous présentons la théorie de l'évidence, et notamment les différents concepts manipulés par cette théorie, ainsi que la règle de fusion qui est utilisée.

Définitions (Dempster, 1968), (Shafer, 1976)

A. Cadre de discernement : On appelle cadre de discernement tout ensemble regroupant N hypothèses exhaustives susceptibles d'être la solution du problème étudié :

$$\Theta = \{H_1, \dots, H_N\} \quad (2.1)$$

B. Référentiel de définition : Il est défini à partir du cadre de discernement et représente l'ensemble des combinaisons possibles de ce dernier :

$$2^\Theta = \{A/A \subseteq \Theta\} = \{\emptyset, H_1, \dots, H_N, H_1 \cup H_N, \dots, \Theta\} = \Omega \quad (2.2)$$

C. Exhaustivité et exclusivité du cadre de discernement :

- **Exhaustivité :** Les hypothèses doivent être exhaustives, ce qui signifie que la solution du problème est forcément l'une des hypothèses du cadre de discernement. Le cadre de discernement est la proposition certaine, et l'ensemble vide est la proposition impossible (nous considérons le cas « Closed-World ¹ »).
- **Exclusivité :** Les hypothèses du cadre de discernement doivent être exclusives, ce qui signifie que la solution est unique.

D. Masse d'évidence : Une masse d'évidence m représente le degré de croyance (degré de crédibilité) associé à un sous-ensemble du référentiel de définition. Les sous-ensembles ayant des masses d'évidence non nulles sont appelés **éléments focaux**.

$$\sum_{A \in \Omega} m(A) = 1 \quad \text{tel que : } 0 \leq m(A) \leq 1 \quad \forall A \quad (2.3)$$

$$m(\emptyset) = 0 \quad (2.4)$$

¹ A noter que certains auteurs comme Smets (Smets, 1988) contestent la notion d'exhaustivité du cadre de discernement. En effet, il est quasiment impossible de recenser toutes les hypothèses de façon exhaustive pour certains problèmes. Dans ce cas, le cadre de discernement est considéré « Open-World », ce qui fait appel à la notion de classe de rejet. Dans le cas d'une exhaustivité, le cadre de discernement est appelé « Closed-World ».

E. Combinaison de sources distinctes : La masse d'évidence de la combinaison de plusieurs sources d'information est donnée par :

$$(\oplus m_i)_{i=1,\dots,M}(A) = \frac{1}{K} \cdot \sum_{A_1 \cap A_2 \cap \dots \cap A_M = A} (m_1(A_1) \cdot m_2(A_2) \dots m_M(A_M)) \quad (2.5)$$

Avec : $A, A_i \subseteq \Omega$ et $A \neq \emptyset$

F. Conflit de Dempster-Shafer : le conflit entre les sources d'information est défini par :

$$K = 1 - \sum_{A_1 \cap A_2 \cap \dots \cap A_M = \emptyset} (m_1(A_1) \cdot m_2(A_2) \dots m_M(A_M)) \quad (2.6)$$

2.3.3 Conflit de Dempster-Shafer et fusion de sources d'information

Dans la littérature, il existe plusieurs travaux (Schubert, 1993) (Schubert, 1995) (Schubert, 1996) et de Régis et autres (Régis, et al., 2004 a) (Régis, et al., 2004 b), utilisant la théorie des fonctions de croyance dans les applications de fusion d'information, notamment dans la caractérisation de la pertinence des sources d'information. Le conflit défini dans le cadre de cette théorie, appelé conflit de Dempster-Shafer, est utilisé comme une mesure de la séparabilité entre sources d'information. Les travaux de Schubert sont consacrés à la détermination des sous-ensembles de variables traitant les mêmes éléments focaux. Les travaux de Régis et autres sont fortement liés à ceux de Schubert, puisqu'ils considèrent que le groupe le plus large sélectionné par la technique de Schubert contient les éléments les plus pertinents. Dans leurs travaux sur les bioprocédés, Régis et autres mettent l'accent sur la notion de la pertinence locale et globale, la pertinence locale étant évaluée pour chaque échantillon de leur bioprocédé. Les auteurs ont souligné l'importance de cette théorie pour caractériser la pertinence des signaux. Toutefois, ils ont signalé quelques problèmes numériques liés aux valeurs des masses d'évidence, et ont proposé une nouvelle formule simplifiée de ce conflit.

2.4 Nouveau critère : Conflit/désaccord binaire entre variables

2.4.1 Introduction

Les résultats présentés dans la littérature nous ont amené à utiliser la notion de conflit dans une procédure FDI. Dans notre recherche d'un critère d'évaluation de la procédure FDI,

nous avons suivi l'hypothèse proposée par Régis (une variable est pertinente si elle est en désaccord avec la plupart des variables disponibles du procédé.) et proposé un nouveau conflit/désaccord. Il faut souligner que le choix de l'algorithme d'apprentissage, en particulier le choix de la forme des classes (hyper-rectangles), a joué un rôle déterminant dans la définition de notre critère qui se base sur l'intersection de l'information disponible sur chacune des variables, en se référant au modèle de bon fonctionnement du système.

L'approche proposée se divise en deux principales étapes : l'apprentissage des données de bon fonctionnement du système et le calcul du conflit sur des données de test susceptibles de contenir des modes défectueux. Afin d'apprendre le modèle de bon fonctionnement une et une seule fois et de l'appliquer sur toutes les pannes possibles, nous avons choisi une forme des classes hyper-rectangles qui nous permet de projeter le domaine modélisé en dimensions inférieures avec une perte d'information minimale et donc d'extraire le contenu informationnel de chaque variable en se basant sur les données de bon fonctionnement. Les contenus informationnels des différentes variables seront intersectés pour juger l'appartenance ou non d'un nouvel élément (échantillon ou acquisition) à la zone de bon fonctionnement. Cette appartenance sera modélisée par une variable booléenne qui représente le conflit entre deux ou plusieurs variables. Dans le paragraphe suivant, nous présentons la formalisation mathématique de cette approche.

2.4.2 Formalisation

Avant de formaliser le critère d'évaluation que nous avons proposé, nous présentons quelques concepts :

- **M** : Nombre de classes (hyper-rectangles) de l'ensemble d'apprentissage (représente les données de bon fonctionnement du système)
- **N** : Nombre d'échantillons de la base de test (susceptibles de contenir des modes défectueux)
- **d** : Nombre de variables
- **Modèle de bon fonctionnement** : $C = \{C_1, C_2, \dots, C_M\}$
- **E_j** : Echantillon de la base de test : $j = 1 : N$
- La classe j est composée de $2*d$ coordonnées :

$$C_k = \{(X_{1,inf}C_k, X_{1,sup}C_k), \dots, (X_{d,inf}C_k, X_{d,sup}C_k)\} \quad (2.7)$$

Avec : $k = 1 : M$

CHAPITRE 2

La connaissance disponible sur chaque variable X_i concernant l'appartenance de l'échantillon E_j au modèle de bon fonctionnement est donnée par:

$$C_{i,j} = \{C_k | X_{i,inf}C_k \leq E_j(X_i) < X_{i,sup}C_k\} \quad (2.8)$$

$C_{i,j}$: ensemble des classes sélectionnées par la variable X_i indiquant l'appartenance de l'échantillon E_j à l'ensemble d'apprentissage. Le conflit, ou désaccord, instantané entre deux variables X_l et X_m relatif à l'appartenance de l'échantillon E_j au modèle de bon fonctionnement est calculé de la manière suivante :

$$Conflit_j(X_l, X_m) = \begin{cases} 1 & \text{Si } (C_{l,j} \cap C_{m,j}) = \emptyset \\ 0 & \text{Si } (C_{l,j} \cap C_{m,j}) \neq \emptyset \end{cases} \quad (2.9)$$

La généralisation de ce conflit à plusieurs variables est assurée par l'équation suivante :

$$Conflit_j(X_1, X_2, \dots, X_d) = \begin{cases} 1 & \text{Si } (C_{1,j} \cap C_{2,j} \cap \dots \cap C_{d,j}) = \emptyset \\ 0 & \text{Si } (C_{1,j} \cap C_{2,j} \cap \dots \cap C_{d,j}) \neq \emptyset \end{cases} \quad (2.10)$$

Avec cette approche, la détection et l'isolation des défauts se ramènent à un problème d'optimisation dans lequel, nous cherchons le sous-ensemble de variables qui maximise le conflit global sur les données défectueuses. Il faut souligner que l'existence d'un conflit pour un échantillon E_j signale la non-appartenance de ce dernier au modèle de bon fonctionnement, ce qui correspond donc à l'apparition d'un défaut (si la non-appartenance persiste bien sûr). Le conflit que nous avons proposé peut être vu comme la distance séparant un échantillon du modèle de bon fonctionnement. Le conflit global permet de chercher le plus petit sous-ensemble E^* de variables qui sépare au mieux les données de test du modèle d'apprentissage, c'est-à-dire, le sous-ensemble des variables pertinentes. Le problème à résoudre est :

$$\max \sum_{j=1}^N Conflit_j(E^*) \quad (2.11)$$

Tel que : $E^* = \{X_I | I \subset \{1, 2, \dots, d\}\}$ est taille minimale

Le problème peut être présenté autrement :

$$\begin{cases} \max \sum_{j=1}^N Conflit_j(E^*) \\ \text{tel que :} \\ \left(\bigcap_{I \in E^*} C_{I,j} \right) = \emptyset \end{cases} \quad (2.12)$$

Le paragraphe suivant va permettre d'illustrer notre méthode.

2.4.3 Illustration

Dans cet exemple, nous considérons un problème en deux dimensions. La figure (Fig. 2.2) représente le modèle d'apprentissage avec trois classes rectangulaires.

Dans cet exemple nous avons traité les deux cas possibles, présence et absence de défaut. Pour l'échantillon E_1 : absence de défaut, pour l'échantillon E_2 : présence de défaut.

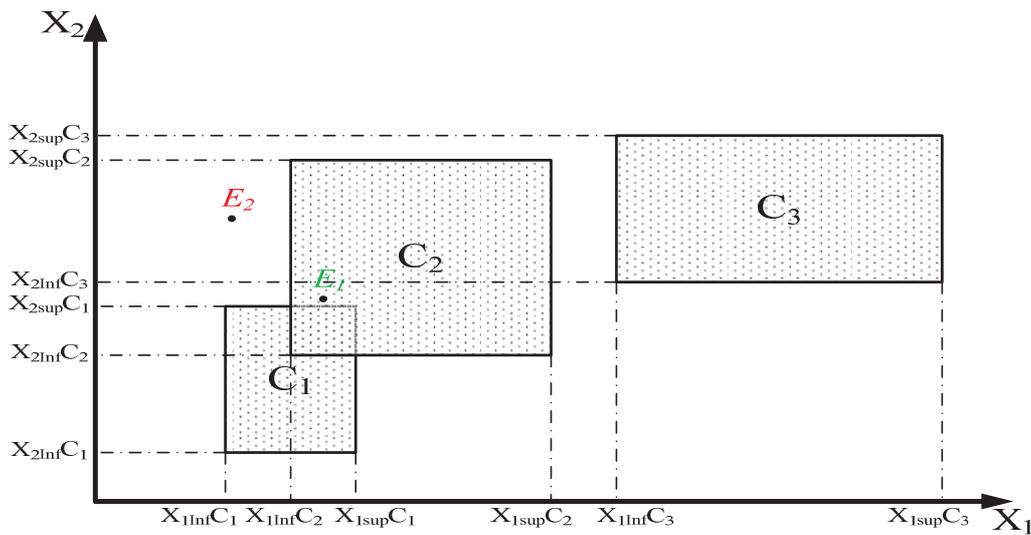


Fig. 2.2 Exemple d'illustration

C_{ij} : représente l'ensemble des classes de l'échantillon i en se référant à la variable j

Pour l'échantillon E_1 :

$$C_{1,1} = \{C_1, C_2\}$$

$$C_{1,2} = \{C_2\}$$

$$C_{1,1} \cap C_{1,2} = \{C_2\} \Rightarrow \text{Conflit}_1 = 0$$

Pour l'échantillon E_2 :

$$C_{2,1} = \{C_1\}$$

$$C_{2,2} = \{C_2, C_3\}$$

$$C_{2,1} \cap C_{2,2} = \{\emptyset\} \Rightarrow \text{Conflit}_2 = 1$$

En analysant cet exemple, nous confirmons que la présence de nombreux échantillons similaires à E_2 permet de détecter la présence d'un défaut du système.

CHAPITRE 2

Dans le paragraphe suivant, nous présentons un exemple d'application dans le domaine automobile.

2.5 Exemple d'application en 4 dimensions : détection de fautes en automobile

2.5.1 Application

Parmi les fonctions d'une automobile intégrant de nombreux capteurs, nous pouvons signaler la gestion du moteur (injection/allumage) qui permet, entre autre, d'optimiser son fonctionnement en termes de consommation. Face à une telle augmentation de la complexité de ce système de gestion, le développement d'un outil FDI performant est primordial.

Dans cette application, notre rôle est d'utiliser les données de bon fonctionnement fournies par les capteurs pour concevoir un système de détection et d'isolation automatique de pannes. Nous disposons de données qui correspondent au fonctionnement nominal du véhicule et au mode défaillant. Ces données ont été enregistrées par la société ACTIA (ACTIA), spécialisée dans le diagnostic automobile, au cours de campagnes d'essais sur véhicules dans le cadre de la thèse de Khayati (Khayati, 2003). Une base de données a été enregistrée lors d'essais routiers avec un véhicule en parfait état de fonctionnement, l'autre a été enregistrée sur le même véhicule après que son fonctionnement ait été volontairement dégradé.

Les bases de données sont constituées de la manière suivante : pour débiter la campagne d'essais, des données de fonctionnement du véhicule à l'arrêt ont été enregistrées, ensuite, en roulant sur autoroute, en ville et en campagne dans le but de couvrir les différents modes de fonctionnement.

Ces pannes sur le véhicule sont les suivantes : la première consiste à débrancher une bougie, la deuxième à obstruer le conduit d'admission et la troisième à débrancher la commande d'un injecteur.

La base de données de bon fonctionnement est composée de 22037 acquisitions pour 10 variables qui sont : la position du papillon, la valeur du couple transmise, durée d'injection, la tension de la sonde d'eau, le débit d'air, le Rapport Cyclique d'Ouverture (RCO ralenti), la correction de la richesse, la consommation du carburant, le régime moteur et l'adaptation de

la richesse. Le tableau suivant présente les caractéristiques des variables utilisées dans cette application :

Variables	Moyennes	Ecart-types	Acronyme
Position du papillon	15,97	15,05	Pos_papillon
Valeur du couple transmise	96,64	79,98	Val_couple_transmise
Durée d'injection	5,57	3,61	Durée_inj
Tension de la sonde d'oxygène	0,34	0,32	Tension_sonde_O2
Débit d'air	82,33	67,49	Débit_d_air
RCO au ralenti	47,04	9,72	RCO_ralenti
Correction de la richesse	127,72	4,03	Correction_richesse
Consommation du carburant	1923,31	1894,61	Conso_carburant
Régime du moteur	2603,27	821,03	Regime
Adaptation richesse en fonctionnement	118,32	0,91	Adapt_richesse_fct

Tableau 2-1 Caractéristiques des variables utilisées

2.5.2 Partitionnement à pas fixe

Dans cette première application, nous avons utilisé un partitionnement à pas fixe de l'espace occupé par les données de bon fonctionnement du véhicule. Cet espace est donc discrétisé en hyper-rectangles, il pourra donc être projeté sur chaque axe sans perte d'information. Le pas de discrétisation sur chaque variable a été choisi manuellement en fonction du pas d'échantillonnage des grandeurs traitées. Une fois le maillage effectué, tous les hyper rectangles qui contiennent des données sont labellisés « bon fonctionnement ».

2.5.3 Calcul du conflit

Nous utilisons, dans cette étape, les données enregistrées sur le véhicule présentant un dysfonctionnement. Pour chaque échantillon, on détermine l'ensemble des pavés labellisés « bon fonctionnement ». Nous calculons, ensuite, le conflit instantané entre les variables de la manière définie précédemment. La panne est détectée en utilisant la valeur moyenne du conflit. Si un jeu de variables possède un conflit relativement élevé, alors, ces variables sont fortement liées à la panne. Dans le cas contraire le jeu de variables ne permet pas de mettre en évidence la panne.

CHAPITRE 2

Nous commençons par présenter les valeurs du conflit global pour différentes paires de variables (figures 2.3-2.6). Ensuite, nous présentons les projections des deux bases de données (bon fonctionnement et panne) dans les plans correspondants (figures 2.7-2.10), afin de vérifier que les variables fortement en conflit sont celles qui permettent de détecter au mieux le défaut. A noter que ces résultats ont été publiés dans (Benkaci, et al., 2008) et (Benkaci, et al., 2009).

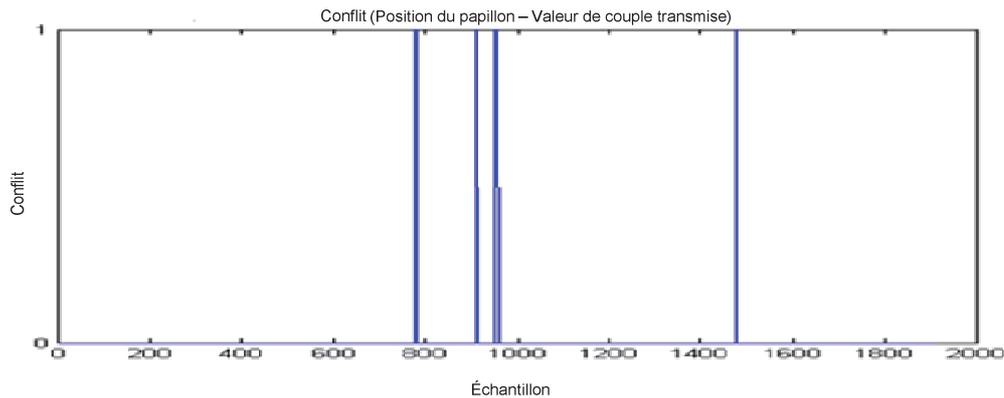


Fig.

2.3 Conflit du jeu de variable « position du papillon – valeur de couple transmise » :

Conflit global = 0.0063

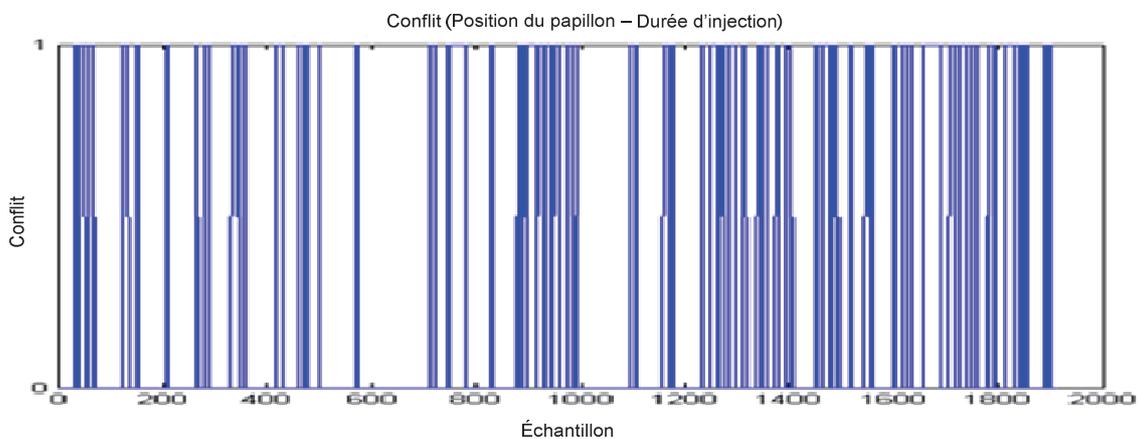


Fig. 2.4 Conflit du jeu de variable « position du papillon – durée d'injection »

Conflit global = 0.2679

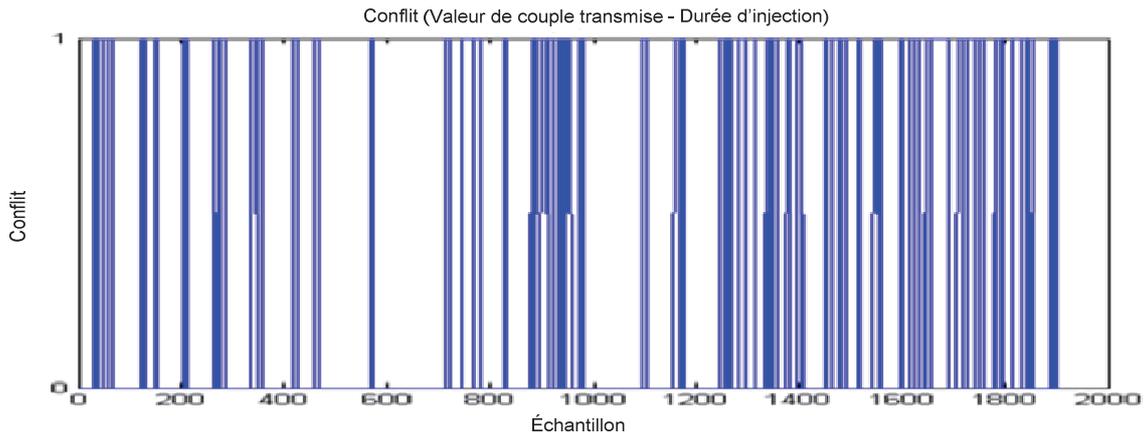


Fig. 2.5 Conflit du jeu de variable « valeur de couple transmise – durée d’injection » :
Conflit global = 0.2329

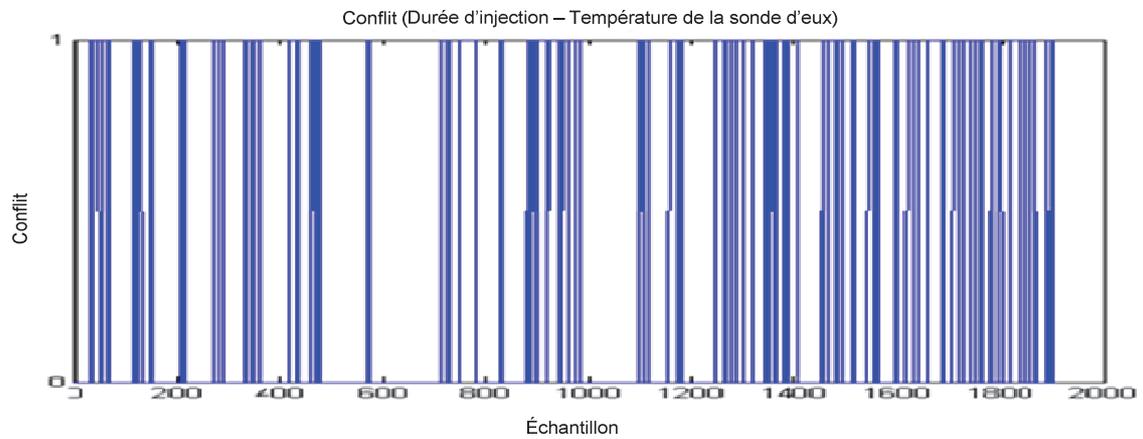


Fig. 2.6 Conflit du jeu de variable « duré d’injection – température de la sonde d’eux » :
Conflit global = 0.2585

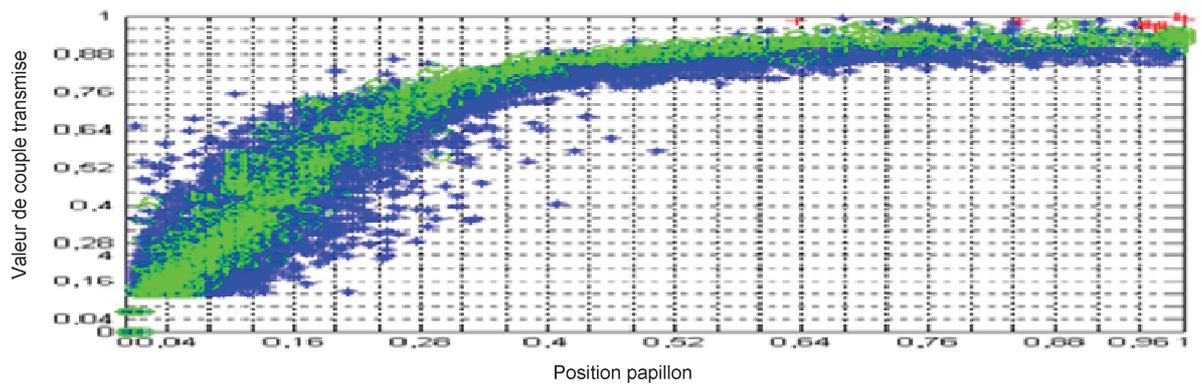


Fig. 2.7 Plan N°1 : « position de papillon – valeur de couple transmise »

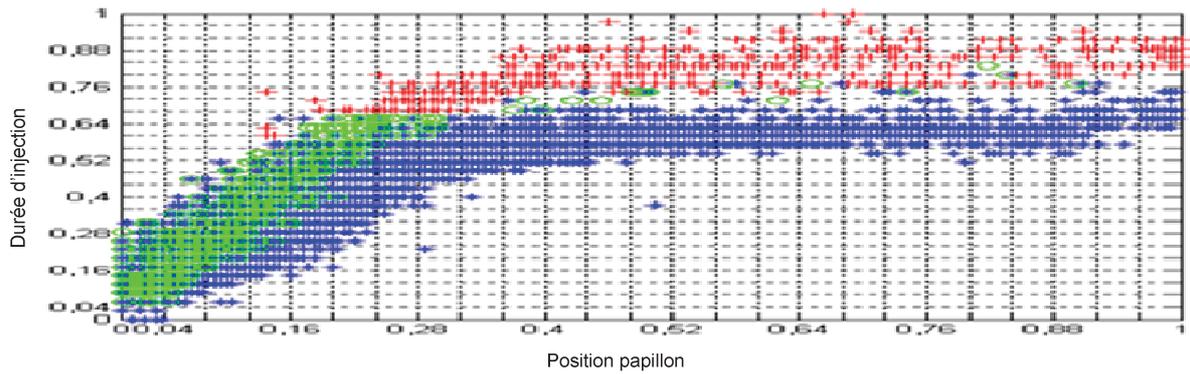


Fig. 2.8 Plan N° 2 : « position de papillon – durée d'injection »

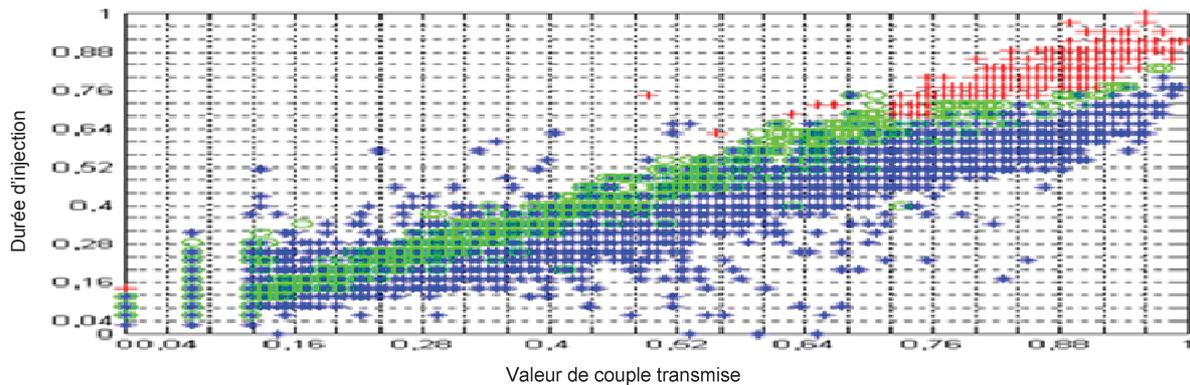


Fig. 2.9 Plan N° 3 : « valeur de couple transmise – durée d'injection »

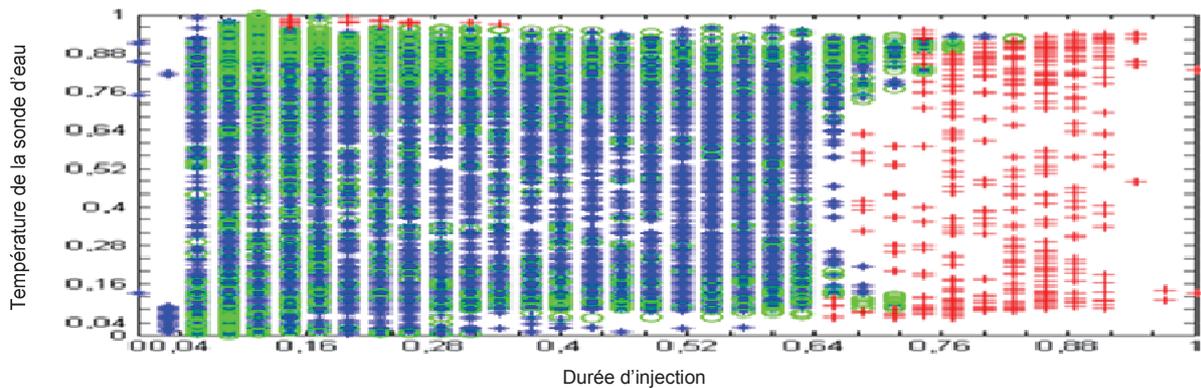


Fig. 2.10 Plan N° 4 : « durée d'injection – température de la sonde d'eau »

Sur les figures (Fig. 2.3 à Fig. 2.6) nous pouvons voir que le conflit entre les variables « position de papillon – valeur de couple transmise » est très faible (0.0063) alors que le conflit entre les trois autres jeux de variables (« position de papillon – durée d'injection », « valeur de couple transmise – durée d'injection » et « durée d'injection – température de la sonde ») est beaucoup plus important. Ce deuxième ensemble de variables permet de mettre en évidence l'existence de la panne générée sur le système d'injection du moteur.

Pour valider cette conclusion nous avons comparé (Fig. 2.7 à Fig. 2.10) les projections des 2 bases de données (véhicule avec et sans dysfonctionnement) sur les plans définis par les différents jeux de variables étudiés. Sur chacune de ces figures, les «*» représentent les données acquises sur le véhicule sain, les «o» représentent les données acquises sur le véhicule présentant un dysfonctionnement, mais qui appartiennent au modèle de bon fonctionnement, alors que les «+» représentent les données acquises dans les mêmes conditions, mais situées en dehors de la classe de bon fonctionnement. Dans chacun des plans, ces mesures sont les seules qui permettent de détecter la panne.

A la lecture des figures (Fig. 2.7 à Fig. 2.10), nous constatons après projection de la base de données dans le plan défini par le jeu de variables de $N^{\circ} 1$, variables très faiblement en conflit, que peu de mesures acquises sur le véhicule présentant un dysfonctionnement différent de bon fonctionnement. Ce jeu de variables ne permet donc pas de détecter la panne considérée. Il n'est donc pas pertinent vis-à-vis de la panne considérée. En revanche, les intersections des projections des deux bases de données sur les plans définis par les jeux de variables qui possèdent des conflits élevés sont beaucoup plus faibles (nombreux «+»). Ce sont donc les variables les plus pertinentes pour détecter la panne considérée. La faible différence entre les conflits s'explique par la présence d'un grand nombre de points pour les faibles sollicitations du moteur, zone de fonctionnement dans laquelle la panne considérée n'est pas détectable. Le problème de non homogénéité de la répartition des données doit donc être considéré dans nos travaux futurs. Nous remarquons sur les jeux de variables pertinents la présence d'une variable qui est fortement liée à la panne considérée (dysfonctionnement du système d'injection), c'est la variable « durée d'injection ». Malgré cela, la mise en œuvre de notre méthode montre tout son intérêt puisque nous retrouvons ici les résultats obtenus manuellement dans des travaux antérieurs.

Bien que le partitionnement en utilisant un maillage à pas fixe de l'espace des données nous ait permis de valider notre approche, sa généralisation à une grande dimension se confronte au problème de l'explosion combinatoire de nombre d'hyper-rectangles modélisant la zone de bon fonctionnement. Ce problème est lié à la malédiction de la dimension, un terme très répandu en recherche. Le premier à avoir introduit cette notion est Bellman (Bellman, 1961), la malédiction de la dimension stipule que la représentation vectorielle nécessaire pour l'estimation d'une fonction donnée en assurant une certaine performance, augmente exponentiellement avec l'augmentation de la dimension du problème traité. Nous allons présenter les différents algorithmes existants dans la littérature qui permettent de faire

un partitionnement en hyper-rectangles de l'espace de données. Ensuite, nous allons faire une synthèse de ces méthodes pour choisir un algorithme adapté et performant.

2.6 Techniques de partitionnement en hyper-rectangles

Il existe plusieurs algorithmes qui permettent de partitionner un domaine en hyper-rectangles. Nous présenterons l'algorithme R-Tree (Guttman, 1984) et ses dérivées, l'algorithme de classification Fuzzy Min-Max (Simpson, 1991), l'algorithme NGE proposé par Salzberg et enfin nous introduirons l'algorithme Fuzzy-ARTMAP de Carpenter.

2.6.1 La famille R-Tree

Il existe plusieurs heuristiques qui appartiennent à cette catégorie de partitionnement. Parmi elles, nous pouvons citer les **R-Tree** (Cha, et al., 2002), **R^{*}-Tree** (Beckmann, et al., 1990) et les **R⁺-Tree** (Sellis, et al., 1987). Nous nous limitons à la description de l'algorithme R-Tree, puisque les autres algorithmes représentent des évolutions de ce dernier. R-Tree est un arbre qui regroupe les vecteurs d'entrée selon leurs proximités. Ces vecteurs sont stockés dans les feuilles de l'arbre tandis que les nœuds de l'arbre représentent la hiérarchie des hyper rectangles englobant. Dans cette partition, le plus petit hyper rectangle pouvant englober l'ensemble des données du niveau inférieur et mémoriser les caractéristiques de l'hyper rectangle englobant tous les objets (REMs ou vecteurs) stockés dans le sous arbre pointé est appelé rectangle englobant minimal (REM). Dans cette approche, la taille des nœuds est fixée à priori (limitée) et la construction de l'arbre se fait d'une manière dynamique par insertions successives. Ces insertions provoquent, en plus de l'augmentation de la profondeur de l'arbre, des fractionnements des feuilles et des nœuds. Pour remédier à la saturation des nœuds, les REMs du nœud saturé sont divisés en deux sous-ensembles de même dimension en minimisant le volume de l'espace couvert par les deux derniers rectangles englobant. Cette division équitable permet de garantir l'équilibre de l'arbre. La figure (Fig. 2.11) représente le principe de l'arbre R-Tree.

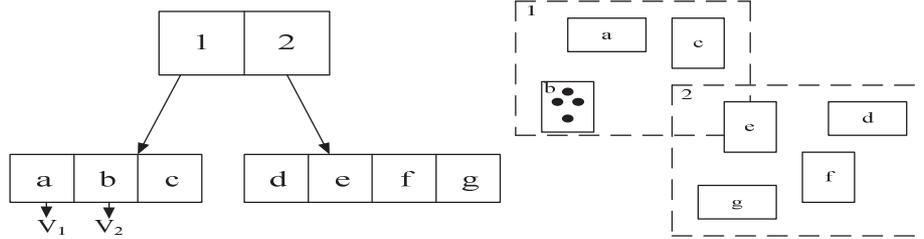


Fig. 2.11 Algorithme R-Tree

À noter que l'inconvénient majeur de ce type d'arborescence est la dégradation de ses performances avec l'augmentation de la dimension du problème traité. Ces arborescences sont, généralement, adaptées à des données uniformément réparties dans l'espace. Elles utilisent généralement la distance euclidienne.

2.6.2 Clustering Fuzzy Min-Max (FMMC)

L'algorithme de clustering Fuzzy Min-Max a été introduit par Simpson en 1992 (Simpson, 1992). Il appartient à la famille des réseaux de neurones, et est doté de trois couches : une couche d'entrée, une couche cachée et une couche de sortie. La couche d'entrée est de taille fixe à la différence de celles des deux autres couches qui évoluent avec le nombre de prototypes donc de classes. Le processus d'apprentissage se déroule en trois phases : expansion, test de recouvrement entre prototypes et contraction.

Principe du FMMC

Le FMMC génère des prototypes sous forme d'hyper-rectangles flous :

$$R_j = \{X, V_j, W_j, f(X, V_j, W_j)\} \quad \forall X \in I^N \quad (2.13)$$

Le vecteur de paramètres $[V_j, W_j]$ correspond aux points min et max de la zone d'influence du prototype. La fonction d'appartenance délivre le degré de correspondance entre un point quelconque et l'hyper-rectangle. Pour une entrée de dimension N , un hyper-rectangle I^N est formé. Les valeurs d'appartenance sont comprises entre 0 et 1, une entrée qui se trouve à l'intérieur de l'hyper-rectangle se voit attribuer un degré d'appartenance de 1. La figure suivante illustre les points min et max en trois dimensions :

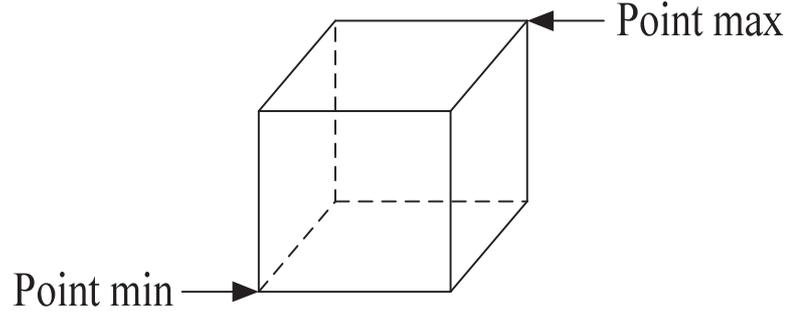


Fig. 2.12 Un hyper-rectangle Min-Max $R_j = \{V_j, W_{j\}} \in I^3$

D'après la définition de l'ensemble des l'hyper-rectangles flous, la classe C_K est définie par :

$$C_K = \bigcup_{j \in K} R_j \quad (2.14)$$

Où K est l'ensemble des indices des hyper-rectangles associés à la classe k .

L'une des plus importantes propriétés de cette approche est de concentrer son traitement sur la détermination des frontières des clusters.

L'apprentissage FMM autorise le recouvrement des hyper-rectangles de la même classe et l'interdit pour les hyper-rectangles de classes différentes. La fonction d'appartenance du $j^{\text{ème}}$ hyper-rectangle $R_j(H_h)$, $0 \leq R_j(H_h) \leq 1$, mesure le degré avec lequel l'entrée E_h tombe à l'extérieure de l'hyper-rectangle R_j . En d'autres termes, elle quantifie la possibilité qu'un échantillon soit supérieur (ou inférieur) à la valeur de point Max (ou Min) dans chaque dimension de l'hyper-rectangle. Si la valeur de la fonction d'appartenance est proche de 1, alors, le point considéré est contenu dans l'hyper-rectangle. La fonction d'appartenance qui vérifie toutes ces conditions est la suivante :

$$r_j(E_h) = \frac{1}{2 \cdot N} \sum_{i=1}^N [\max(0, 1 - \max(0, \gamma \cdot \min(1, a_{hi} - w_{ji}))) + \max(0, 1 - \max(0, \gamma \cdot \min(1, v_{ji} - a_{hi})))] \quad (2.15)$$

Où $E_h = (e_{h1}, e_{h2}, \dots, e_{hN}) \in I^N$ est la $h^{\text{ème}}$ entrée, $V_j = (v_{j1}, v_{j2}, \dots, v_{jN})$ est le point minimum de R_j , $W_j = (w_{j1}, w_{j2}, \dots, w_{jN})$ est le point maximum de R_j et γ est le paramètre de sensibilité qui permet de régler la rapidité de diminution de la valeur d'appartenance quand la distance entre E_h et R_j croît.

L'architecture des réseaux neuronaux utilisés par la technique FMCC est représentée sur la figure (Fig. 2.13). La couche d'entrée, F_A , est constituée d'un nombre de nœuds égal à la dimension du vecteur d'entrée. La couche F_C contient un nombre de nœuds égal au nombre de classes. La couche cachée, F_R , est appelée la couche d'hyper-rectangles, chaque nœud représente un hyper-rectangle flou. Les connexions de F_A à F_R sont les points Min et Max, et la fonction de transfert de F_R est la fonction d'appartenance définie précédemment.

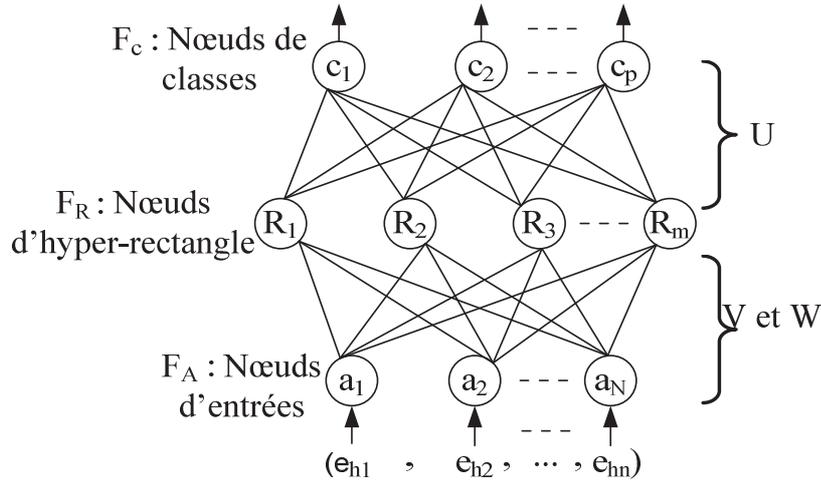


Fig. 2.13 Architecture du réseau FMCC

Les points Min et Max sont stockés respectivement dans les matrices V et W . Les poids des connexions liant la couche cachée à la couche de sortie sont binaires et sont stockés dans la matrice U .

$$u_{jk} = \begin{cases} 1 & \text{Si } R_j \in C_k \\ 0 & \text{Si } R_j \notin C_k \end{cases} \quad (2.16)$$

La sortie d'un nœud de la couche F_C représente le degré d'appartenance de l'élément A_h à la classe k . La fonction de transfert de chaque nœud de la couche F_C effectue l'union floue de l'ensemble des valeurs floues des hyper-rectangles appropriés. Cette opération est définie par :

$$c_k = \max_{j=1}^m (r_j \cdot u_{jk}) \quad (2.17)$$

La couche de sortie peut être utilisée de deux façons. Si une décision continue est exigée, la sortie est utilisée directement. Si une décision binaire est exigée, le nœud de F_C possédant la valeur la plus élevée est sélectionné et sa valeur est basculée à 1 pour indiquer

CHAPITRE 2

que c'est la sortie qui correspond le mieux à l'entrée présentée, tandis que les sorties des autres nœuds sont forcées à 0.

Dans la technique FMMC, la méthodologie de l'apprentissage flou Min-Max est un processus d'expansion/contraction. L'ensemble d'apprentissage D est constitué d'un ensemble de M paires ordonnées $\{X_h, d_h\}$, où $X_h = (x_{h1}, x_{h2}, \dots, x_{hN}) \in I^N$ est le vecteur d'entrée, et $d_h \in \{1, 2, \dots, m\}$ est l'indice de l'une des classes. Le processus d'apprentissage débute par la sélection aléatoire de l'une des paires ordonnées de l'ensemble D , puis recherche l'hyper-rectangle de la même classe à étendre pour y inclure l'entrée. Le critère d'expansion est :

$$N\theta \geq \sum_{i=1}^N (\max(w_{ji}, x_{hi}) - \min(v_{ji}, x_{hi})) \quad (2.18)$$

Où : $0 \leq \theta \leq 1$ est la taille de l'hyper-rectangle.

Si aucun hyper-rectangle vérifiant le critère d'extension n'est trouvé, un nouvel hyper-rectangle est créé dans le réseau. Le processus d'extension permet de former des classes non-linéairement séparables. Il existe une possibilité de chevauchement quand les hyper-rectangles sont élargis, ce chevauchement est contrôlé par un test de chevauchement qui n'autorise que le chevauchement de la même classe. Par contre, si le chevauchement apparaît sur des classes différentes, le processus de contraction est activé et le chevauchement est éliminé. Finalement, le processus d'apprentissage du FMMC possède une série d'activités qui créent et élargissent des hyper-rectangles, puis affine ces hyper-rectangles par le test du chevauchement et le processus de contraction.

2.6.3 Nested Generalized Exemplar (NGE)

Ce type d'apprentissage se base sur une classe d'exemplaires, où les prototypes sont un ensemble d'hyper-rectangles dans l'espace euclidien de dimension N . Les classes d'exemplaires sont soit hyper-rectangles, soit, des échantillons uniques de l'ensemble d'apprentissage. Les points uniques sont considérés comme des hyper-rectangles insignifiants.

L'entrée de l'algorithme NGE est un ensemble d'exemples, présentés de façon incrémentale. Chaque exemple est défini par un vecteur des variables associées à une classe prédéfinie. Les N attributs sont utilisés pour décrire les exemples définis dans l'espace

euclidien de dimension N . Les attributs ont des valeurs réelles, tandis que les classes peuvent être binaires, discrètes ou continues.

D'une façon simple, le système NGE généralise un ensemble de points initiaux fixé par l'utilisateur, et ensuite les élargit suivant une ou plusieurs dimensions chaque fois qu'un nouvel exemple est présenté. Le choix de l'hyper-rectangle à élargir dépend de la métrique choisie. Dans le cas de données précises, la distance euclidienne pondérée est utilisée à la fois pour comparer point-à-point ou point-à-hyper-rectangle. L'ajustement des pondérations a été adopté par NGE comme un moyen de renforcer la pertinence des variables dans un processus de classification. Ce renforcement peut être positif ou négatif suivant la contribution de chaque variable à la classification correcte des exemples. Durant l'apprentissage, l'incrément de la pertinence d'une variable est indiquée par la diminution de la valeur de poids associé W_{vi} . Une stratégie similaire est adoptée pour associer les poids aux exemplaires.

Initialement, W_R est mis à 1. Après, W_R est mis à jour de façon incrémentale. A chaque étape :

$$W_R = U/C \quad (2.19)$$

Où : U est le nombre d'utilisation de R et C est le nombre de prédictions correctes réalisées par R . D'après cette équation, W_R varie inversement avec la fiabilité de R . Tous les poids des variables W_{Vi} sont initialisés à 1. Ces poids sont mis à jour, façon incrémentale, selon la contribution de chaque variable V_i à la classification de l'exemple E_{new} . Si E_{new} appartient à une classe exemplaire R , la correspondance entre la valeur E_{Vi} de chaque variable V_i de E_{new} et la valeur de R_{Vi} de la classe d'exemplaire R est examinée. Si E_{Vi} superpose (E_{Vi} se trouve sur la même gamme R_{Vi}) R_{Vi} , W_{Vi} est incrémenté ; sinon, W_{Vi} est décrémenté. Si E_{new} n'est pas correctement classée, l'ajustement des poids se fait de manière opposée.

La distance entre un hyper-rectangle R et un échantillon est calculée de deux façons :

A. Si R est un point :

$$Dist = W_R \cdot \sqrt{\sum_{i=1}^m \left(W_{Vi} \cdot \frac{E_{Vi} - R_{Vi}}{max_i - min_i} \right)^2} \quad (2.20)$$

Où :

W_R : poids de l'exemplaire R

CHAPITRE 2

W_i : poids de la composante i

E_{Vi} : la valeur de la $i^{ème}$ composante de E

R_{Vi} : la valeur de la $i^{ème}$ composante de R

max_i et min_i : la valeur maximale et minimale de la variable Vi

m : nombre de variables

B. Si R est un hyper-rectangle :

$$Dist = W_R \cdot \sqrt{\sum_{i=1}^m \left(W_{Vi} \cdot \frac{dif_i}{max_i - min_i} \right)^2} \quad (2.21)$$

Où :

$$dif_i = \begin{cases} E_{Vi} - R_{supérieur} & \text{Si } E_{Vi} > R_{supérieur} \\ R_{inférieur} - E_{Vi} & \text{Si } E_{Vi} < R_{inférieur} \\ 0 & \text{Autrement} \end{cases} \quad (2.22)$$

La distance entre R et E est équivalente à la distance entre le point E_{Vi} et la surface ou l'arrête ou un coin de R (Salzberg, 1991).

Algorithme NGE

Le processus d'apprentissage NGE est résumé ainsi : les paramètres sont initialisés aléatoirement par l'utilisateur qui choisit le nombre de points qui seront transformés en exemplaires ; ces points deviennent des hyper-rectangles virtuels qui seront les conditions initiales de l'algorithme. A chaque nouvel échantillon E_{new} , le processus d'apprentissage cherche parmi tous les hyper-rectangles déjà créées celui qui en est le plus proche $H_{proche1}$ et le plus proche en second $H_{proche2}$. Ces deux hyper-rectangles sont les candidats potentiels pour une éventuelle généralisation. Si E_{new} et $H_{proche1}$ appartiennent à la même classe, l'hyper-rectangle $H_{proche1}$ est élargi pour inclure E_{new} , ce processus est appelé généralisation. Sinon, si E_{new} et $H_{proche2}$ appartiennent à la même classe, NGE spécialisera $H_{proche1}$ en réduisant sa taille, en éliminant ses arrêtes provenant de E_{new} , et $H_{proche2}$ devient alors l'hyper-rectangle le plus proche $H_{proche2}$ est modifié pour englober E_{new} . Par contre, si la classe de E_{new} diffère des deux hyper-rectangles, alors E_{new} devient un nouvel exemplaire.

2.6.4 Comparaison

Avant de faire un choix sur l'algorithme d'apprentissage que nous allons utiliser dans notre algorithme FDI que nous proposons, nous avons repris une comparaison faite par Carpenter entre les algorithmes FMCC et NGE et le fuzzy-ARTMAP. Dans cette

comparaison, les auteurs ont montré que les trois algorithmes utilisent des formes hyper-rectangles pour représenter leurs catégories dans un apprentissage supervisé et qu'ils utilisent la même équation d'apprentissage pour mettre à jour les catégories. Ces trois algorithmes diffèrent significativement dans leur réponse à une prédiction incorrecte. Les auteurs accordent cette différence au paramètre de vigilance de fuzzy-ART non disponible sur les deux autres algorithmes. FMMC et NGE ne possèdent pas de système de contrôle interne pour gérer la taille des hyper-rectangles. Par exemple, NGE permet d'élargir ou de réduire les hyper-rectangles, ce qui ne lui permet pas d'atteindre de stabilité caractérisant fuzzy-ART. Avec la propriété de stabilité et l'utilisation de match tracking, le fuzzy-ARTMAP construit automatiquement le nombre de catégories nécessaire pour apprendre n'importe quel ensemble de données. Dans FMMC, chaque classe de sortie est associée à un seul hyper-rectangle, ce qui limite l'ensemble de catégories qu'il peut apprendre. Par contre, fuzzy-ARTMAP peut associer plusieurs catégories à une seule entrée.

Cette comparaison faite par Carpenter et ses collègues, nous a poussé à opter pour l'algorithme fuzzy-ARTMAP. Une description détaillée de cet algorithme sera présentée dans le troisième chapitre, chapitre dans lequel nous allons combiner fuzzy-ARTMAP avec le conflit proposé pour détecter et isoler des défauts dans une application automobile.

2.7 Conclusion et choix de Fuzzy-ARTMAP

Dans ce chapitre, nous avons passé en revue les méthodes de détection et d'isolation en commençant par positionner le problème de la FDI.

Puisque la plupart des méthodes de détection et d'isolation de défauts à base de données s'articulent autour d'un processus d'apprentissage, nous avons donné une définition de l'apprentissage automatique, suivi d'une revue des méthodes existantes, en particulier, l'apprentissage incrémental que nous allons utiliser dans notre algorithme de détection et d'isolation. Dans cette thèse nous avons choisi, en nous inspirons des travaux de Salzberg, un algorithme d'apprentissage qui renvoie des classes facilement interprétables (hyper-rectangle) et possèdent une bonne propriété de généralisation.

Dans la troisième partie de ce chapitre, nous avons décrit le problème traité dans le cadre de cette thèse et proposé un système FDI performant se basant sur un apprentissage performant basé sur un conflit/désaccord entre les variables du système à surveiller, inspiré du conflit proposé par Dempster-Shafer en théorie des fonctions de croyance. Ce conflit se base

CHAPITRE 2

sur l'intersection des connaissances disponibles sur chaque variable par rapport au modèle appris, constitué d'hyper-rectangle. Le sous ensemble possédant le conflit le plus élevé est le plus concerné par le défaut. Ce conflit nous permet d'abord de déterminer la présence d'un défaut, ensuite, il nous permet de mettre en évidence les éléments responsables de ce défaut.

Un exemple tiré de l'industrie automobile a permis d'illustrer ce concept, sur la base d'un modèle des données de bon fonctionnement réalisé par des hyper-rectangles résultant d'un partitionnement à pas fixe de l'espace des données. Ce type de partitionnement nous a permis de montrer l'intérêt de notre approche, mais a mis en évidence le problème de l'explosion combinatoire du nombre de labels avec l'augmentation de la dimension de l'espace.

Pour résoudre ce problème, nous avons choisi d'utiliser un « partitionnement intelligent » de l'espace des données qui décrit le fonctionnement nominal du système à surveiller. Pour cela nous avons détaillé plusieurs méthodes de partitionnement de données en hyper-rectangles. Nous avons commencé par les méthodes R-Tree qui sont très limitées en grande dimension. Ensuite, nous avons exposé deux méthodes qui sont, FMMC et NGE. En s'appuyant sur la comparaison réalisée par Carpenter et autres, nous avons choisi d'utiliser fuzzy-ARTMAP comme algorithme d'apprentissage.

2.8 Références

ACTIA [Online]. - <http://www.actia.com/fr/>.

Basseville M and Nikiforov I. V Detection of abrupt changes - theory and applications [Book]. - Englewood Cliffs, N. J : Prentice Hall, 1993.

Beckmann N [et al.] The R*-tree: An efficient and robust access method for points and rectangles [Conference] // In Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data. - Atlantic City, NJ : [s.n.], 1990. - pp. 322-331.

Bellman R. E Adaptive control processes [Report] : Points: high dimensional data; curse of dimensionality. - Princeton, NJ : Princeton University Press, 1961.

Benkaci M, Doncescu A and Kabbaj N Selection of relevant paramètres based on conflict [Conference] // International Conference on Network-Based Information Systems. - Indianapolis, USA : [s.n.], 2009. - pp. 499 – 504.

Benkaci M, Jammes B and Doncescu A Sélection de paramètres pertinents à base de conflit [Conference] // Conférence internationale francophone de l'automatique. - 2008.

Bloch I and Hunter A Fusion : General concepts and characteristics [Journal] // International journal of Intelligent systems. - 2001. - Vol. 16. - pp. 1107-1134.

Bloch I Fusion d'information en traitement du signal et des images [Journal] // International journal of Intelligent systems. - 2001. - Vol. 16. - pp. 1107-1134.

Bloch I Fusion d'informations en traitement du signal et des images [Book]. - Paris, France : Hermès, 2003.

Byron J. G HYPER-RECTANGLE-BASED DISCRIMINATIVE DATA GENERALIZATION AND APPLICATIONS IN DATA MINING [Report] : PhD thesis / B.Sc., Simon Fraser University. - [s.l.] : Spring, 2007.

Cha Guang- Ho and Chung Chen-Wan The GC-Tree : A High-Dimensional Index Structure for Similarity Search in Image Databases [Journal] // IEEE Transactions On Multimedia. - March 2002. - Vol. 4 (1).

Dempster A A generalisation of bayesian inference [Journal] // Journal of the Royal statistical Society. - 1968. - Vol. 30. - pp. 205-247.

Fisher R. A On the mathematical foundations of theoretical statistics [Journal] // Phil. Trans. of the Royal Soc.. - 1922. - Vol. 220. - pp. 309–368.

Frank P. M Fault diagnosis in dynamic systems using analytical and knowledge based redundancy - A survey and new results [Journal] // Automatica. - 1990. - Vol. 26. - pp. 459-474.

Guttman A R-trees: Adynamic index structure for spatial searching [Conference] // ACM SIGMOD International Conference on Management of Data. - Boston, MA : [s.n.], 1984. - pp. 47-57.

Hartley R. V. L Transmission of Information [Journal] // Bell System Technical Journal. - 1928. - Vol. 7. - pp. 535-563.

Khayati P Système de diagnostic automatique à base de reconnaissance statistique des formes : Application à l'automobile [Report] : Thèse de Doctorat / Université Toulouse 3 (Paul Sabatier). - Toulouse : [s.n.], 2003.

Mooney R and DeJong G Learning Schemata for Natural Language Processing [Conference] // Proceedings of IJCAI-85. - Los Angeles, California : [s.n.], 1985. - pp. ,681-687.

Patton R. J, Frank P and Clark R Fault diagnosis in dunamic systems - theory and application [Book]. - [s.l.] : Prentice Hall, 1989.

Régis S [et al.] Evaluation de la pertinence de paramètres biochimiques et classification pour la caractérisation des états physiologiques dans un bioprocédé par la théorie de l'évidence [Conference] // XXIème rencontres de la société Francophone de Classification. - Bordeaux : [s.n.], 2004 b.

Régis S, Desachy J and Doncescu A Evaluation of biochemical sources pertinence in classification of cell's physiological states by evidence theory [Journal] // In FUZZY IEEE. - Budapest : [s.n.], 2004 a.

Salzberg S. L A Nearest Hyperrectangle Learning Method [Journal] // Machine Learning. - 1991. - Vol. 6. - pp. 251-276.

CHAPITRE 2

Salzberg S. L Learning with nested generalized exemplars [Report] : Ph.D. (Technical Report TR-14-89) / Department of Computer Science ; Harvard University. - Cambridge, MA : [s.n.], 1989.

Schubert J Finding a posterior domain probability distribution by specifying nonspecific evidence [Journal] // International Journal of Uncertainty. - 1995. - Vol. 3. - pp. 163-185.

Schubert J On nonspecific evidence [Journal] // International Journal of Intelligent Systems. - 1993. - Vol. 8. - pp. 711-725.

Schubert J Specifying nonspecific evidence [Journal] // International Journal of Intelligent Systems. - 1996. - Vol. 11. - pp. 525-563.

Sellis T, Roussopoulos N and Faloutsos C THE R+-TREE: A DYNAMIC INDEX FOR MULTI-DIMENSIONAL OBJECTS [Report] / Computer Science Department ; University of Maryland. - College Park, MD 20742 : [s.n.], 1987.

Shafer G A Mathematical Theory of Evidence [Report] / Princeton University Press. - New Jersey : [s.n.], 1976.

Shannon C A Mathematical Theory of Communication [Journal] // Bell System Technical Journal. - July and Oct 1948. - Vol. 27 and 28. - pp. 379-423 and 623-656. - published in two parts.

Simpson P Fuzzy min-max classification with neural networks [Journal] // Heuristics: J. Knowledge Eng. - 1991. - Vol. 4. - pp. 1-9.

Simpson P. K Fuzzy min-max neural networks. Part 1: classification [Journal] // IEEE Transactions on Neural Networks. - 1992. - Vol. 3. - pp. 776-786.

Smets Ph Belief functions [Book] / ed. London Academic Press. - London : [s.n.], 1988. - pp. 253-286. - In SMETS Ph, MAMDANI A., DUBOIS D. and PRADE H : Non standard logics for automated reasoning.

Thomas J [et al.] Diagnostic à base de connaissances hétérogènes: application au domaine de l'automobile [Journal] // Revue de l'électricité et de l'électronique (REE). - Marseille : [s.n.], Avril 2009. - Vol. 4. - pp. 49-54.

Willsky A. S A survey of design methods for failure detection in dynamic systems [Journal] // Automatica. - 1976. - Vol. 12. - pp. 601-611.

*“The difference between a crisis and the
opportunity is when you learn about it”*

Dans le chapitre précédent, nous avons opté pour un partitionnement en hyper-rectangles de l'espace des variables, et nous avons choisi le partitionnement par Fuzzy-ARTMAP. Dans le présent chapitre nous allons faire un bref rappel sur les réseaux de neurones, puis, nous détaillons l'algorithme Fuzzy-ARTMAP. Ensuite, nous proposons une amélioration de cet algorithme en le combinant avec l'algorithme des \mathcal{K} -moyennes. Enfin, le système de détection et d'isolation de fautes proposé dans le chapitre précédent sera mis en œuvre sur des données réelles d'une application automobile.

3 FDI à base de Fuzzy-ARTMAP

3.1 Introduction

Le cerveau humain est considéré comme le composant de calcul le plus complexe connu par l'homme. Les fonctionnalités multiples du cerveau comme sa pensée, sa sauvegarde et sa capacité de résoudre des problèmes ont inspiré de nombreux chercheurs qui ont tenté de créer un modèle de calcul qui correspond aux fonctionnalités du cerveau : ils ont obtenu le calcul neuronal.

3.1.1 Neurone

Le neurone est considéré comme l'unité fondamentale du système nerveux et, en particulier, du cerveau. Par définition, un neurone est une fonction non linéaire à valeurs bornées. La figure (Fig. 3.1) représente un neurone formel (Mcculloch, et al., 1943) (Minsky, et al., 1969) :

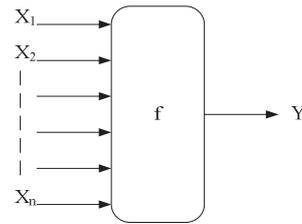


Fig. 3.1 Un neurone réalisant une fonction f , possédant X_i entrée et une sortie Y

Il existe plusieurs structures de neurones, deux d'entre elles sont les plus utilisées. Dans la première structure, la sortie y du neurone est une fonction non linéaire d'une combinaison linéaire des entrées X_i pondérées par les paramètres W_i appelés poids ou poids synaptiques, par analogie à la biologie. Un terme constant b_0 , appelé biais, s'ajoute à cette somme pondérée des entrées. La fonction principale est appelée fonction d'activation. Elle est généralement de type sigmoïdale, telle que la tangente hyperbolique ou l'arc tangent. Ce type de neurones représente l'élément de base des réseaux de neurones les plus utilisés à l'heure actuelle, les Perceptrons Multi-Couches (PMC, ou MLP pour Multi-Layer Perceptrons).

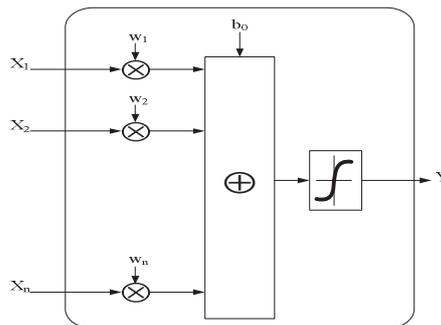


Fig. 3.2 Neurone de base de Perceptrons Multi-Couches

L'équation de la sortie Y est la suivante :

$$Y = f(X_1, X_2, \dots, X_n) = th \left(b_0 + \sum_{i=1}^n w_i X_i \right) \quad (3.1)$$

Concernant la deuxième structure, les paramètres influencent directement la non-linéarité de la fonction f . Cette fonction peut être une fonction à base radiale (Powell, 1987) (RBF pour Radial Basis Function), ou bien une fonction d'ondelette (Mallat, 1989). Par exemple l'équation suivante représente la sortie d'un neurone RBF avec une non-linéarité gaussienne :

$$Y = \exp \frac{\sum_{l=1}^n (X_l - \rho_l)^2}{2\sigma^2} \quad (3.2)$$

Les paramètres ρ_l représentent les coordonnées du centre de la gaussienne et σ son écart-type.

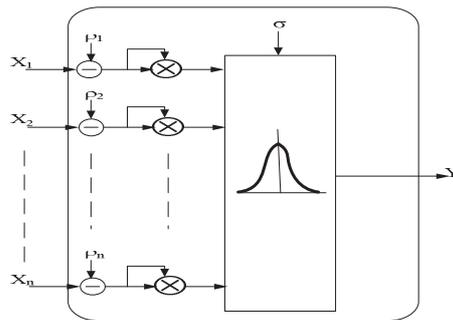


Fig. 3.3 Neurone RBF à non-linéarité gaussienne

Le choix du type de neurones dépend essentiellement de l'approximation que nous souhaitons réaliser. Les RBF ou les ondelettes par exemple ont des non-linéarités locales, qui tendent vers zéro dans toutes les directions de l'espace des entrées sauf pour des valeurs proches des centres des gaussiennes. Par conséquent, ce type de neurone sera essentiellement utilisé pour faire de la discrimination (classification). Par contre, les neurones à fonction d'activation sigmoïde seront utilisés pour effectuer des approximations globales et continues sur l'intégralité de l'espace des entrées (régression).

En résumé, un neurone réalise une fonction non-linéaire de l'ensemble de ses entrées, en intégrant des paramètres internes. L'intérêt majeur des neurones réside dans les propriétés résultantes de leurs combinaisons dans un réseau.

3.1.2 Les réseaux de neurones

Un réseau de neurones est composé de plusieurs neurones organisés de façon à réaliser une approximation d'une fonction souhaitée. En d'autres termes, un réseau de neurones est une cascade de fonctions non-linéaires générées par chacun des neurones membres. Il existe deux types de réseaux de neurones, les réseaux bouclés et les réseaux non-bouclés.

3.1.2.1 Réseaux de neurones non bouclés

Dans un réseau de neurones non-bouclé, les neurones sont connectés de façon à ce que l'information transite des entrées vers les sorties sans retour en arrière (pas de Feedback) (Dreyfus, et al., 2004). Ce type de réseau réalise une (ou plusieurs) fonctions algébriques de ses entrées, par composition des fonctions réalisées par chacun des neurones. Dans un réseau de neurone non bouclé, le temps ne joue aucun rôle. C'est pour cette raison qu'ils sont qualifiés de réseaux statiques. On appelle neurones d'entrées les éléments du réseau qui

reçoivent les informations à traiter, on appelle neurones de sortie, les neurones qui calculent la sortie du réseau. Les neurones qui effectuent les calculs intermédiaires sont appelés neurones cachés. Il existe plusieurs possibilités de connecter les neurones afin de former un réseau, l'architecture la plus globale est celle du réseau dit complètement connecté :

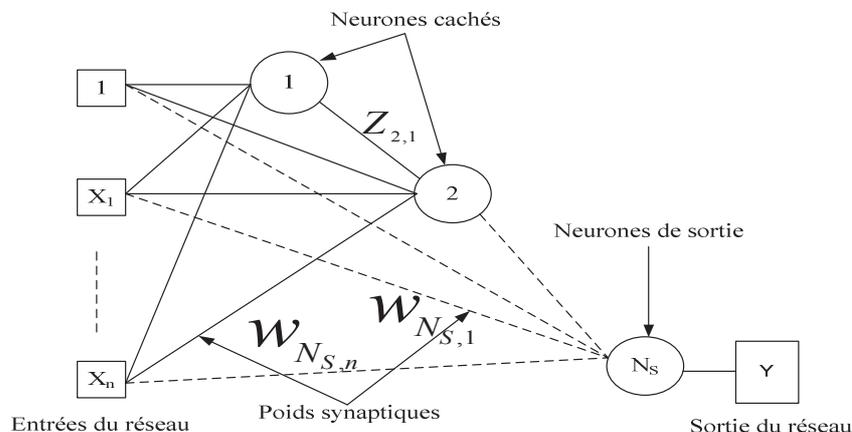


Fig. 3.4 Architecture d'un réseau de neurones non bouclé complètement connecté à n entrées, N_s-1 neurones cachés et 1 neurone de sortie. W : connexions entrées – neurones, Z : connexions entre neurones.

Ce modèle représente le modèle de base, en éliminant certaines connexions et en regroupant des neurones, il pourra être simplifié afin d'obtenir des modèles moins complexes.

3.1.2.2 Réseaux de neurones bouclés

Dans un réseau de neurones bouclé, dit réseau de neurones récurrent ou dynamique, il existe au moins un cycle à travers lequel l'information fait un retour à son point de départ. Cela suppose que la sortie d'un neurone peut être fonction d'elle-même, ce qui n'est possible que lorsque la notion de temps est prise en compte. Ce type de réseau est utilisé lors de la modélisation dynamique de processus non linéaires (Aussem, 1995).

Après ce bref rappel sur les réseaux de neurones, et avant d'aborder la classification par les réseaux fuzzy-ARTMAP, nous allons présenter en détail la théorie de résonance adaptative.

3.2 La théorie de résonance adaptative

La théorie de résonance adaptative (ART, pour Adaptive Resonance Theory) se focalise sur les problèmes de reconnaissance des formes et la modélisation neuronale depuis les

CHAPITRE 3

années 60. Les premières idées de théorie remontent à 1976, quand Grossberg (Grossberg, 1976) propose une introduction formelle des bases de la résonance adaptative, pour contrer le dilemme stabilité – plasticité dans les réseaux d'apprentissage à compétitivité. Ce dilemme met en évidence la capacité d'un système d'apprentissage à préserver ses connaissances acquises tout en conservant ses capacités à apprendre de nouvelles caractéristiques. Les modèles d'architecture ART peuvent obtenir des formes stables en temps réel. Plusieurs versions d'architectures neuronales de réseaux ART ont été proposées. L'architecture de base est ART1, proposée par Carpenter et autres (Carpenter, et al., 1987 a). ART1 peut apprendre et organiser des caractéristiques binaires. Le module ART2 a été proposé également par Carpenter et autres (Carpenter, et al., 1987 b), il est dédié à la classification des séquences de caractéristiques analogiques arbitraires. ART2a (Carpenter, et al., 1991 a) est une amélioration en terme de rapidité d'exécution de ART2. Dans le fuzzy-ART, Carpenter et ses collègues (Carpenter, et al., 1991 b) proposent une version plus sophistiquée de réseaux ART en remplaçant l'opérateur 'ET logique' par 'ET flou', c'est à ce type de réseaux que nous s'intéresserons dans cette thèse. Il existe une version supervisée de réseau ART qui est le réseau ARTMAP (Carpenter, et al., 1991), et une version floue qui est le fuzzy-ARTMAP (Carpenter, et al., 1992). Notre travail sera basé sur cette dernière architecture. La théorie de la résonance adaptative a été utilisée, par ces auteurs, dans la reconnaissance des formes visuelles invariantes (Carpenter, et al., 1988) ou ils ont discuté son équivalence en biologie (Carpenter, et al., 1990).

3.2.1 Théorie de la résonance adaptative

La théorie de la résonance adaptative décrit un type de réseau de neurones compétitif capable, à la fois de former des catégories en se basant sur les données présentées en ligne, de détecter des nouveautés et de préserver les catégories déjà apprises. C'est un algorithme d'apprentissage non-supervisé.

Un module ART possède quatre propriétés :

- **L'auto-organisation :** dans un module ART, le sous-système d'attention est basé sur un apprentissage compétitif permettant d'améliorer les catégories tout en éliminant le bruit.

- **L'auto-ajustement de la mémoire de recherche** : le module peut chercher dans la mémoire de façon parallèle et changer de façon adaptative son ordre de recherche.
- **Toute caractéristique apprise a un accès direct à sa catégorie correspondante.**
- **Le module ART peut progressivement modifier sa vigilance** en utilisant l'environnement appelé professeur (Teacher). Si l'environnement n'admet pas l'actuelle reconnaissance du système, le module ajuste ce paramètre pour être d'avantage vigilant.

Avant d'aborder le réseau fuzzy-ARTMAP, nous avons jugé utile de présenter le fonctionnement du module ART :

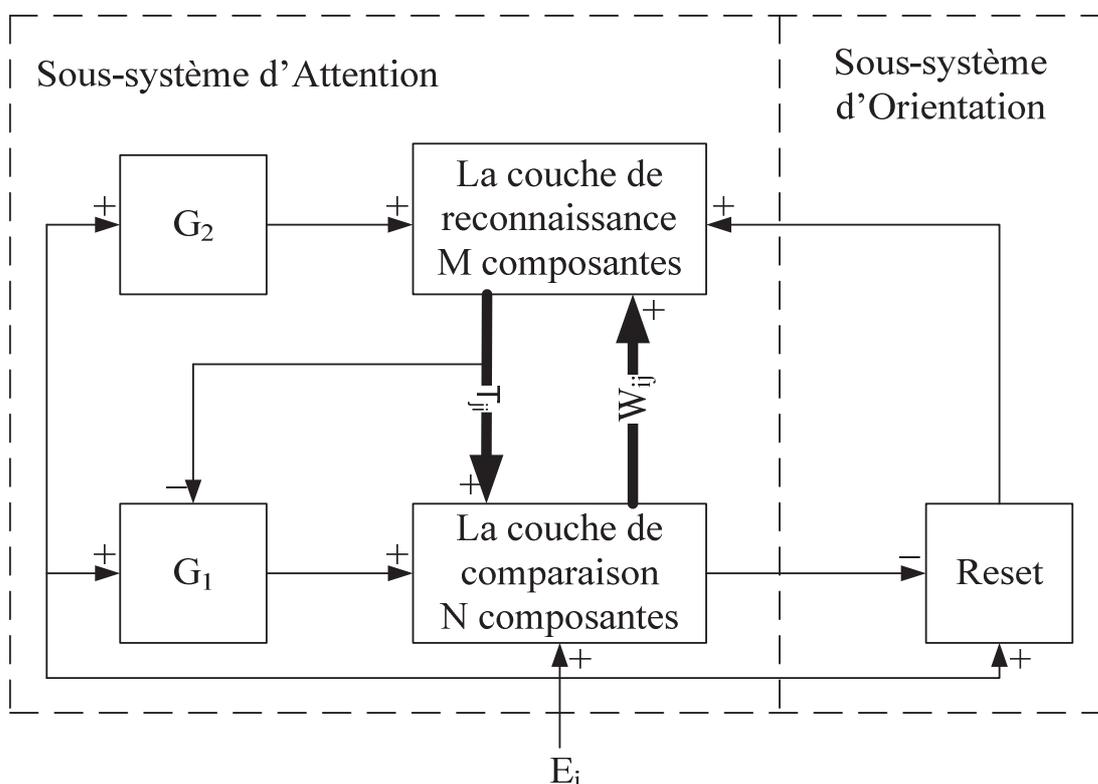


Fig. 3.5 Architecture du Module ART, N: dimension de vecteur d'entrées (i = 1:N) et M : nombre de catégories disponibles (j = 1:M)

Ce réseau est composé, comme illustré dans la figure (Fig. 3.5), de deux sous-systèmes : le sous-système d'attention et le sous-système d'orientation.

Le sous-système d'attention est composé de deux couches de neurones, la couche d'entrée F_1 dite de comparaison et la couche de sortie F_2 dite de reconnaissance, complètement

CHAPITRE 3

connectées entre elles par deux types de connexions : celles qui vont de la couche de comparaison vers la couche de reconnaissance et celles qui relient la couche de reconnaissance F_2 à la couche de comparaison F_1 . La dynamique du réseau ART est contrôlée par les gains G_1 et G_2 . Les connexions, $F_1 \Rightarrow F_2$ et $F_2 \Rightarrow F_1$, sont respectivement représentées par les matrices W et T . Chaque composante du vecteur d'entrées E est représentée par un neurone d'entrée i , $i = 1 : \dim(E)$. La matrice T représente les prototypes de la couche de reconnaissance. Les éléments de la couche de comparaison possèdent une dimension égale au nombre de catégories déjà existantes. Ces catégories représentent la sortie du réseau ART après apprentissage. Le nombre de catégories pourra ne pas être défini à l'avance, car, il sera modifié dynamiquement au fur et à mesure de la réception des données.

Dans sa dynamique, le réseau ART cherche à faire correspondre le vecteur d'entrées au prototype le plus proche. Cette correspondance est gérée par un test de vigilance caractérisé par paramètre de vigilance ρ . Ce test de vigilance est effectué au niveau du sous-système d'orientation. Si ce test est vérifié alors le prototype sera mis-à-jour en prenant en compte le nouveau vecteur d'entrées. Cette mis-à-jour continue est réalisée par un 'ET logique' entre le vecteur d'entrée et le prototype et démontre la capacité de sauvegarde des acquis du réseau ART. Le 'ET logique' permet une diminution progressive de la valeur des prototypes, ce qui permet la stabilité des catégories et la convergence du réseau. Si, lors du test de vigilance, aucun prototype de la couche de reconnaissance ne vérifie la contrainte de vigilance, le réseau ART crée une nouvelle catégorie (nouveau neurone de sortie au niveau de la couche F_1). Ce mécanisme montre la capacité des réseaux ART à détecter et classer de nouveaux prototypes. La figure suivante représente les entrées/sorties des deux couches de neurones d'un réseau ART :

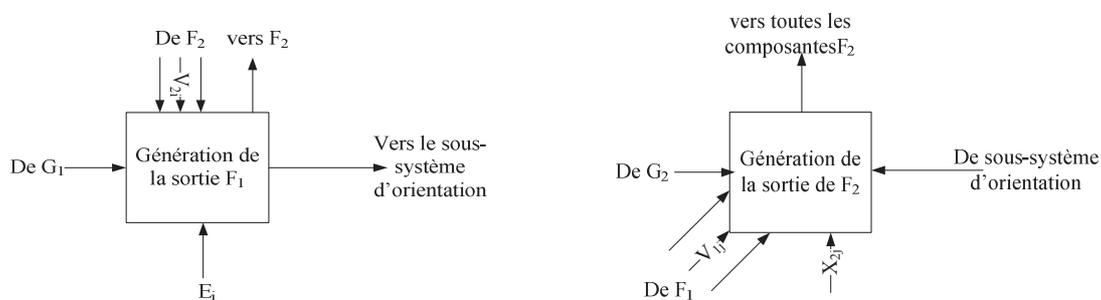


Fig. 3.6 a- Entrées/sorties de la couche F_1 . b- Entrées/sorties de la couche F_2

La couche F_1 reçoit trois types d'entrées : le vecteur d'entrée E_i , un gain de contrôle G_1 et une entrée interne V_{1i} résultante du produit scalaire des sorties de la couche F_2 avec les connexions associées T_{2j} .

Symétriquement, la couche F_2 reçoit trois entrées : une en provenance de sous-système d'orientation, le gain de contrôle G_2 et une entrée interne V_{2j} , produit scalaire de la sortie F_1 par les connexions appropriées associées W_{ij} .

La procédure de catégorisation par un réseau ART est la suivante : la couche d'entrée (couche de comparaison) reçoit les entrées en provenance du monde extérieur et les propage vers la couche de sortie (couche de reconnaissance) *via* les connexions W . Une fois ces informations propagées, les neurones de la couche F_2 seront activés. Le produit scalaire de l'entrée par les poids des connexions donne le niveau d'activation de chaque neurone de la couche F_2 . Une compétition est lancée au niveau de la couche F_2 , en utilisant la technique du « gagnant reçoit tout » (WTA, pour Winner Take All) : le neurone possédant le niveau d'activation le plus élevé sera sélectionné. C'est ce neurone qui présente la corrélation (similarité) la plus élevée avec les données d'entrée. Le prototype représenté par ce neurone sera propagé vers la couche de comparaison via les connexions T du neurone gagnant. A ce moment là, la couche F_1 active le mode comparaison pour réaliser le test de vigilance représenté par l'équation suivante :

$$\frac{\|E_k \wedge T_j\|}{\|E_k\|} \geq \rho \quad (3.3)$$

Où ρ est le paramètre de vigilance, \wedge est le 'ET Logique' et $\|E\| = \sum_i E_i$ est la norme du vecteur. Ce test détermine donc le pourcentage de bits communs entre le vecteur d'entrée E_k et le prototype gagnant T_j . Donc, si tous les bits de vecteur d'entrée sont égaux à ceux de ce prototype, la correspondance est maximale. Dans le cas opposé, c'est-à-dire, aucune correspondance entre les deux entités alors la correspondance est nulle. Si le test de vigilance échoue, les prototypes activés seront testés les uns après les autres (selon le degré d'activation) jusqu'à la satisfaction de test de vigilance.

Comme indiqué précédemment, dans le reste de notre travail, nous nous intéressons au réseau de neurone de type ARTMAP, en particulier, le réseau Fuzzy-ARTMAP.

3.3 Les modèles ARTMAP existants

Les réseaux ARTMAP représentent une nouvelle architecture des réseaux ART possédant la propriété d'auto-organisation. Cette architecture permet au réseau d'être stable, rapide, utilisable en applications en ligne, et adaptatif à de nouveaux vecteurs d'entrées. Cette dernière propriété est primordiale dans un processus d'apprentissage dans lequel nous devons faire face à des événements inattendus dans un environnement peu connu. Un algorithme d'apprentissage performant doit être capable d'apprendre de nouvelles données.

Dans la littérature, plusieurs améliorations des réseaux ARTMAP ont été proposées. Elles peuvent être classées selon leurs processus interne de correspondance (matching). Ce processus dépend de la procédure d'activation des catégories, qui peut être déterministe ou probabiliste. Parmi les réseaux déterministes nous pouvons citer ARTMAP, ART-EMAP, ARTMAP-IC, default ARTMAP, simplified ARTMAP, distributed ARTMAP etc... Les réseaux déterministes produisent des classes en utilisant un ou plusieurs hyper-rectangles.

D'autre part, on trouve les réseaux dits probabilistes parmi lesquels nous pouvons citer : PROBART, PFAM, MLANS, Gaussian ARTMAP, ellipsoid ARTMAP, Boosted ARTMAP, μ ARTMAP etc... Les classes élaborées par les réseaux probabilistes sont représentées par des fonctions de densité de probabilités.

3.3.1 L'auto-organisation de la prédiction de cartes

L'architecture ARTMAP est une version prédictive des réseaux ART. Dans un système ARTMAP, deux modules ART sont combinés afin de prédire les catégories d'un vecteur d'entrées inconnu en utilisant un vecteur de prototypes. Cette architecture est appelée ARTMAP, car, elle transforme un vecteur de caractéristiques donné en un vecteur de carte (prototypes) appris. L'apprentissage ARTMAP est supervisé.

3.3.2 Maximisation de la généralisation et minimisation de l'erreur de prédiction

Le réseau ARTMAP a été conçu de manière à maximiser la généralisation et à minimiser l'erreur de prédiction dans un processus d'apprentissage rapide en temps réel et en réponse à un vecteur d'entrées arbitraire. Tout système ARTMAP doit apprendre pour obtenir

une prédiction très précise en un temps optimal. Deux propriétés se dégagent de cette dernière contrainte ; l'efficacité, c'est-à-dire, l'utilisation de peu de cycles d'apprentissage, et la souplesse dans le sens où l'apprentissage stable permet d'avoir la continuité d'apprentissage vers d'autres bases de données sans modifier les connaissances acquises antérieurement. Dans les réseaux ARTMAP, la capacité de mémoire peut être choisie large, sans affecter la rapidité d'apprentissage ou le pouvoir de généralisation.

3.3.3 L'auto-organisation par un système expert

Le réseau ARTMAP réalise ses combinaisons de propriétés désirables en s'appuyant sur l'auto-organisation par un système expert. Il fait intervenir les propriétés de base d'un modèle ART pour produire un test d'hypothèses autonome et une mémoire de recherche parallèle pour des codes de reconnaissance appropriés. Le test d'hypothèse s'achève par un état de résonance qui persiste tant que l'entrée reste approximativement constante. La résonance génère une sorte d'attention qui sélectionne les caractéristiques vérifiant l'adéquation entre l'entrée activée et la catégorie (prototype) choisie. L'apprentissage de ces caractéristiques se produit lors de ces états de résonance, ce qui explique le nom de la résonance adaptative. Le schéma d'un réseau ARTMAP est le suivant :

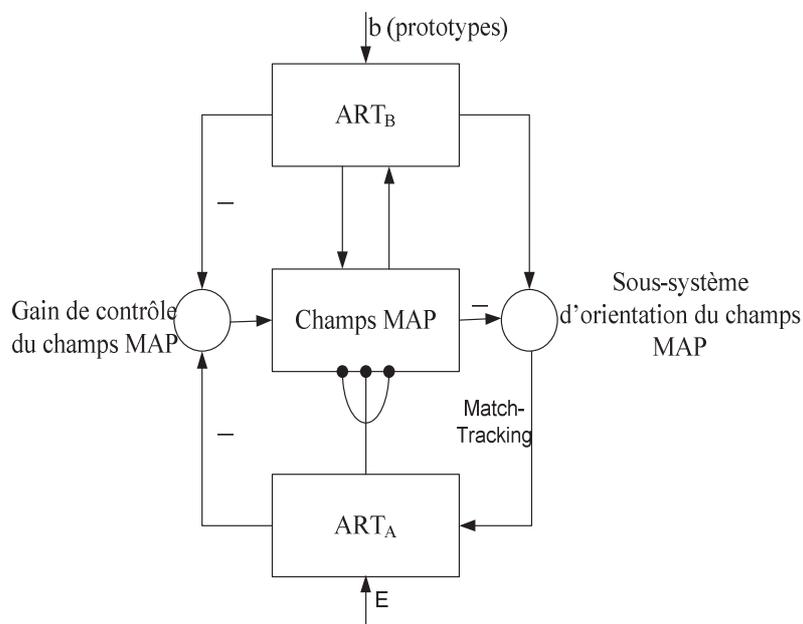


Fig. 3.7 Le diagramme ARTMAP (Carpenter and Al, 1991)

3.4 Partitionnement par le réseau Fuzzy-ARTMAP

Le réseau fuzzy-ARTMAP est la version floue du réseau ARTMAP. Les deux modules ART_a et ART_b sont remplacés par deux modules de type Fuzzy-ART. Dans ce réseau, les calculs spécifiques à la logique floue sont rajoutés au module ART. Il permet de passer de la théorie des ensembles, utilisée dans un module ART, à la théorie des ensembles flous (tous les opérateurs logiques seront remplacés par leurs équivalents flous). Donc, l'opérateur d'intersection (\cap) sera remplacé par l'opérateur ' ET flou' (\wedge). Ce changement de logique permet d'avoir une version très performante de ce type de réseaux. Avec l'incorporation de la logique floue, le fuzzy-ARTMAP permet d'apprendre, non seulement des données binaires, mais aussi des données réelles.

Dans fuzzy-ARTMAP deux modules fuzzy-ART sont reliés par un module inter-ART qui représente le champ MAP. Ce champ permet, d'une part, de prédire les associations entre catégories, et d'autre part, de lancer le processus de suivi de correspondance (match tracking), où le paramètre de vigilance du fuzzy- ART_a est incrémenté pour faire face à une non-correspondance de catégorie au niveau de fuzzy- ART_b . Ce processus réorganise les catégories de façon que la prédiction de l'erreur ne soit pas recalculée lors de nouvelles présentations d'entrées. L'une des propriétés les plus intéressantes de fuzzy-ARTMAP est l'utilisation d'un apprentissage séquentiel, propriété qui permet d'utiliser ce réseau dans les applications en ligne. Dans un module fuzzy-ART, l'apprentissage converge de façon continue parce que les poids adaptatifs de réseau sont monotonement décroissants.

3.4.1 L'activité des couches fuzzy-ARTMAP

Dans l'architecture fuzzy-ARTMAP (Fig. 3.8), une couche dite F_0 est introduite pour traiter les données avant de les présenter au réseau. La couche F_1 reçoit alors des informations à la fois de la couche F_0 et de la couche de sortie F_2 . Cette dernière couche est la couche de reconnaissance. Chaque couche possède un vecteur d'activité : le vecteur d'activité de la couche F_0 est : $E = (E_1, \dots, E_M)$ avec $E_i \in [0,1]$ et $i=1 : M$ (nombre de variables), le vecteur d'activité de la couche F_1 est $x = (x_1, \dots, x_M)$ et le vecteur d'activité de la couche F_2 est $y = (y_1, \dots, y_N)$ avec N le nombre de catégories de la couche F_2 .

Le vecteur de poids

Il correspond aux catégories de la couche F_2 , sa représentation est la suivante : $w_j \equiv (w_{j1}, \dots, w_{jM})$. Ces poids sont initialisés à 1.

$$w_{j1}(0) = \dots = w_{jM}(0) = 1 \quad (3.4)$$

Chaque catégorie est considérée non-active (non-assignée). Dès qu'elle sera sélectionnée, elle deviendra active.

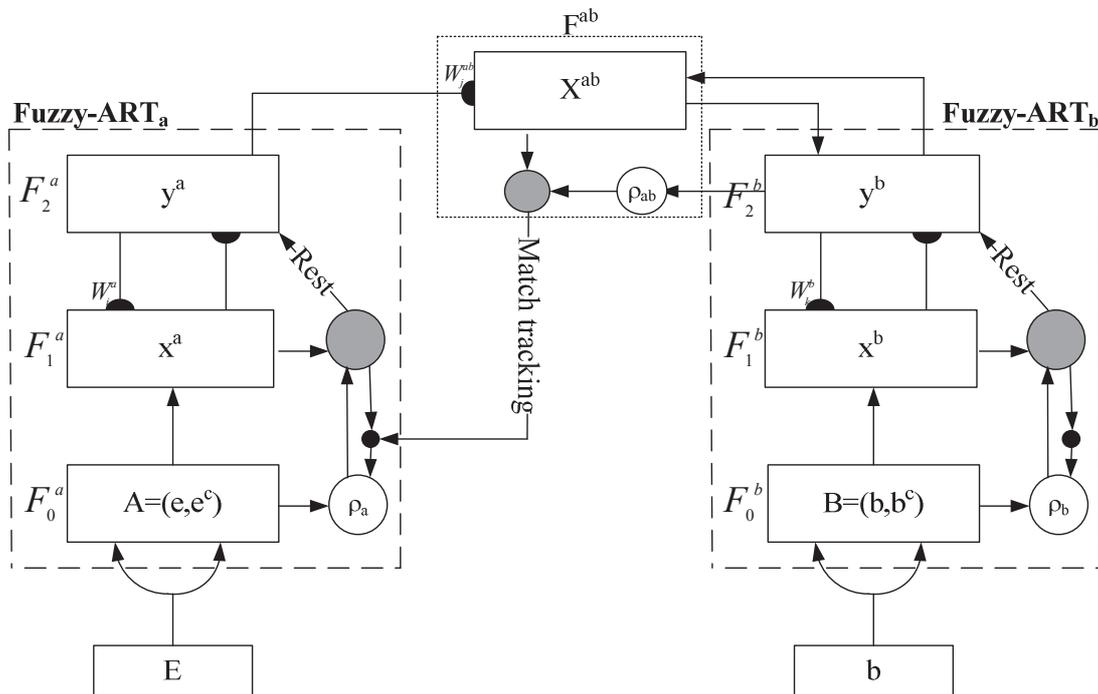


Fig. 3.8 L'architecture de réseau Fuzzy-ARTMAP (Carpenter et al. 1992)

3.4.2 Algorithme Fuzzy-ARTMAP

Le déroulement de l'algorithme fuzzy-ARTMAP peut être résumé en deux phases ; la première est la phase d'**apprentissage** et la deuxième est la phase de **test**.

3.4.2.1 Phase d'apprentissage

A. Initialisation :

A l'initialisation du réseau, aucun nœud de la couche de reconnaissance n'est activé. Les entrées (E , b), représentant le vecteur d'entrées et le vecteur des classes associées, sont

CHAPITRE 3

présentées au réseau. Dans le réseau FAM (abréviation de Fuzzy-ArtMap), les caractéristiques E_i doivent être codées en complément à 1 et elles doivent être comprises entre 0 et 1. Le complément d'un vecteur de caractéristiques d'une dimension M doit être de dimension $2*M$:

$$A = (E; E^c) = (e_1, e_2, \dots, e_M; e_1^c, e_2^c, \dots, e_M^c) \quad (3.5) \quad e_i \in [0, 1]$$

$$e_i^c = (1 - e_i) \quad (3.6)$$

B. Activation des catégories :

La présentation d'un vecteur A à l'entrée de la couche F_1 provoquera son activation. Ce vecteur sera propagé vers la couche F_2 à travers les connexions synaptiques de poids W . L'activation des nœuds j de la couche F_2 suivra la loi de Weber :

$$T_j(A) = \frac{|A \wedge w_j|}{\alpha + |w_j|}, \quad (3.7) \quad j = 1, 2, \dots, N$$

Où l'opérateur flou \wedge est défini par :

$$(a \wedge b)_i \equiv \min(a_i, b_i) \quad (3.8)$$

Et

$$|w_j| \equiv \sum_{i=1}^M w_i \quad (3.9)$$

Le paramètre de choix α est compris entre 0 et 1. Il gère le nombre d'itérations de recherche des catégories et permet donc, entre autre, de définir la profondeur de recherche de la couche de reconnaissance F_2 . En d'autres termes, ce paramètre détermine le nombre de catégories à créer, ainsi, si α a une valeur grande, le réseau favorise la création de nouvelles catégories, alors qu'une petite valeur de α œuvre à la réutilisation des catégories disponibles en mémoire. Le processus de compétitivité au sein de la couche de reconnaissance F_2 se base sur l'équation suivante :

$$T_j = \mathit{argmax}\{T_j; j = 1, \dots, N\} \quad (3.10)$$

Après cette étape, seul le nœud J de la couche F_2 ayant la valeur d'activation maximale restera actif. Dans le cas où plusieurs nœuds vérifient ce test, le nœud possédant l'indice le plus faible est choisi. Une fois ce choix effectué, le test de vigilance sera exécuté sur le nœud J . Ce test mesure la similitude entre la catégorie du nœud J , w_J , et le vecteur d'entrée A en utilisant le paramètre de vigilance ρ :

$$\frac{|A \wedge w_J|}{M} \geq \rho \quad (3.11)$$

Si le test est vérifié, la classe du nœud J est activée, sinon, ce nœud sera désactivé. Le fuzzy-ART_a cherche un nouveau nœud vérifiant le test de vigilance. Dans le cas où aucun nœud ne correspond à cette entrée, une catégorie non assignée de la couche F_2 lui est attribuée. Le paramètre de vigilance permet de contrôler la taille maximale des catégories qui est fixée à $2 \cdot (1 - \rho)$ et doit être comprise entre 0 et 1. Ce paramètre aura également une influence significative sur le nombre de catégories à créer. Une valeur importante de vigilance favorise la création de nouvelles catégories tandis qu'une petite valeur permet de réutiliser les catégories existantes et, par conséquent, de diminuer le nombre de catégories créées.

C. Activation des classes :

La sélection et l'activation du nœud gagnant J par la couche de reconnaissance F_2 provoque l'activation du champ MAP *via* les connexions synaptiques de poids W^{ab} et assigne une classe à l'entrée présentée. Si la réponse du nœud sélectionné ne correspond pas à celle associée au patron d'entrée (b), le processus de suivi de la correspondance (Match Tracking) est déclenché. Dans ce processus, le paramètre de vigilance ρ de fuzzy-ART_a est incrémenté avant de faire une nouvelle recherche au niveau de la couche F_2 . Cette procédure se poursuit jusqu'à ce qu'un nœud de cette couche ou un nœud non-actif soit affecté à cette entrée. L'incrément du paramètre de vigilance est réalisée par :

$$\rho = \frac{|A \wedge w_J|}{|A|} + \varepsilon \quad (3.12)$$

Le paramètre ε peut être positif ou négatif. Une grande valeur positive de ce paramètre favorisera la création de nouvelles catégories quand le premier test de vigilance n'est pas vérifié. Une grande valeur négative aide à réutiliser les catégories déjà existantes quand la première tentative de classification échoue. Dans ce cas, le nombre de catégories à créer est minimisé.

D. Apprentissage :

Dans le processus d'apprentissage, une entrée (observation) e met à jour les poids synaptiques W_J en créant un nouveau lien associatif vers la couche F^{ab} . Bien sur, dans le cas où J correspond à un nœud de F_2 nouvellement choisi, cette mise à jour est réalisée par l'équation suivante :

$$w_j^{(new)} = \beta(A \wedge w_j^{(old)} + (1 - \beta)w_j^{(old)}) \quad (3.13)$$

Où : β contrôle la vitesse d'apprentissage

Le paramètre β est ajusté selon la vitesse d'apprentissage souhaitée. Un apprentissage lent est obtenu en fixant $\beta \ll 1$, tandis qu'un apprentissage rapide est obtenu en mettant $\beta = 1$. Pour ce dernier cas, le réseau FAM fournira à sa sortie un hyper-rectangle de taille minimale enveloppant l'entrée e . En d'autres termes, une valeur faible de β réduit la vitesse de changement de la taille des catégories, alors qu'une valeur importante de ce paramètre rend le changement de la taille des catégories plus rapide.

3.4.2.2 Phase de test

Lors de la phase de test, l'entrée présentée au réseau est associée à un nœud de la couche F_2 par la fonction de choix. La classe du champ d'association du nœud gagnant est attribuée à l'entrée testée. Aucun test de vigilance ou de suivi de correspondance n'est effectué. Une analogie entre le module ART et le module fuzzy-ART est présentée sur la figure suivante afin de mettre en évidence le passage de ART à fuzzy-ART.

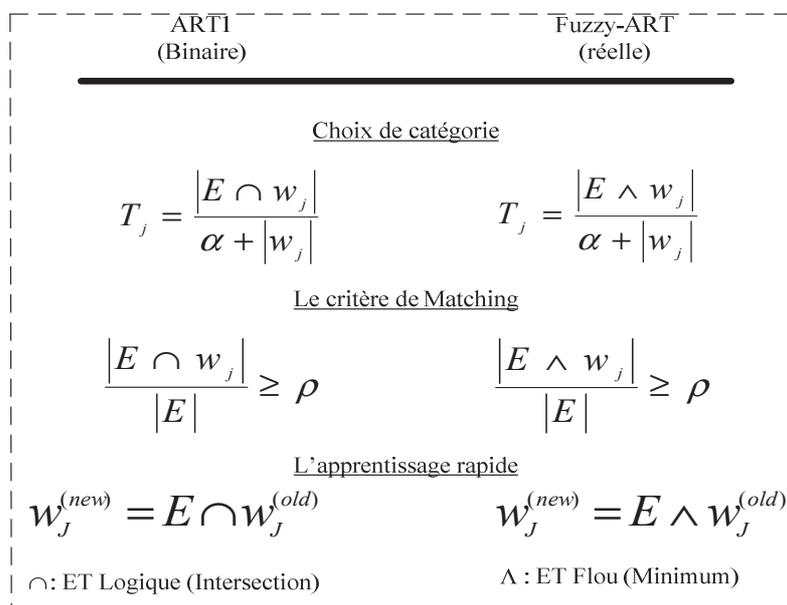


Fig. 3.9 Comparaison ART1 vs Fuzzy-ART (Carpenter et al. 1992)

3.4.3 Fuzzy-ARTMAP et ordre de passage de données

Comme nous l'avons spécifié auparavant, l'apprentissage utilisé par fuzzy-ARTMAP est un apprentissage séquentiel. L'ordre de passage des données joue donc un rôle très

important sur la formation des catégories, et surtout sur la qualité des résultats obtenus. Plusieurs travaux ont été réalisés pour faire face à cette problématique. L'une des propositions faite dans la littérature est de considérer plusieurs ordres de passage aléatoires des données, et de garder celui qui maximise les performances. L'inconvénient majeur de cette proposition est lié au nombre important de procédures d'apprentissage à faire pour en déduire l'ordre de passage satisfaisant les performances recherchées, ce qui engendre une augmentation significative de la complexité algorithmique de cette solution.

Dans l'article (Dagher, et al., 1999), les auteurs proposent un algorithme pour définir l'ordre de passage des données. Ordered fuzzy-Artmap est divisé en trois étapes :

La première consiste à choisir une première observation à présenter au réseau, cette observation n'étant qu'un des centres des clusters de l'ensemble d'apprentissage.

La deuxième étape consiste à chercher les centres des clusters restants.

Dans la troisième étape, les observations restantes sont ordonnées l'une après l'autre, selon la distance euclidienne croissante par rapport aux centres des clusters déterminés dans les phases 1 et 2.

Les auteurs indiquent que leur algorithme surclasse fuzzy-ARTMAP dans la résolution de plusieurs problèmes de classification et que la complexité algorithmique est égale à celle de fuzzy-ARTMAP.

Dans le paragraphe suivant, nous proposons un nouvel algorithme, se basant sur celui de Dagher, permettant d'améliorer les résultats de fuzzy-ARTMAP tout-en réduisant la complexité de calcul. Dans cette procédure, nous proposons, d'abord, d'éliminer les redondances des échantillons de la base d'apprentissage, ensuite de faire un clustering K-means, (Lloyd, 1982) (Steinhaus, 1956) sur les données restantes. Cet algorithme a été choisi pour sa rapidité de convergence et sa simplicité conceptuelle. Les clusters résultants seront envoyés un par un au réseau, par ordre décroissant de taille. Cette procédure va nous permettre de limiter au maximum la non vérification du test de vigilance et l'enclenchement des procédures de suivi de correspondance. Tout d'abord nous commençons par une présentation de l'algorithme K-means.

3.4.4 K-means avec fuzzy-ARTMAP

La méthode K-means, encore appelée *algorithme des centres mobiles* (Benzécri, 1973) représente les clusters par leur centre de gravité.

CHAPITRE 3

L'algorithme k-means mis au point par McQueen en 1967 est l'un des algorithmes de clustering les plus connus. Le principe de cette méthode est le suivant : nous donnons pour commencer, k centres arbitraires c_1, c_2, \dots, c_k où chaque c_i représente le centre d'une classe C_i . Chaque classe C_i est représentée par l'ensemble des individus les plus proches de c_i . Après cette initialisation, nous effectuons une deuxième partition en regroupant les individus autour des m_j qui prennent alors la place des c_j (m_j est le centre de gravité de la classe C_j , calculé en utilisant les nouvelles classes obtenues). Le processus est ainsi réitéré jusqu'à atteindre une configuration où aucune amélioration n'est possible. Cette méthode est convergente et surtout avantageuse du point de vue calcul. L'algorithme K-means est représenté sur la figure suivante :

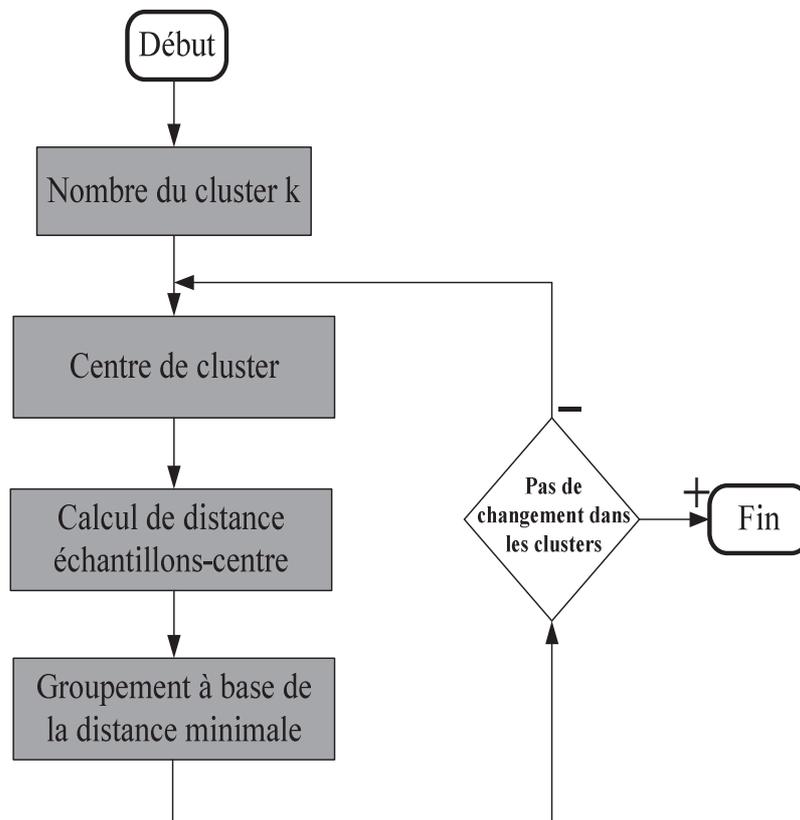


Fig. 3.10 Algorithme K-moyennes

Le schéma que nous avons proposé pour améliorer les résultats de l'apprentissage avec fuzzy-ARTMAP est représenté sur la figure (Fig. 3.11).

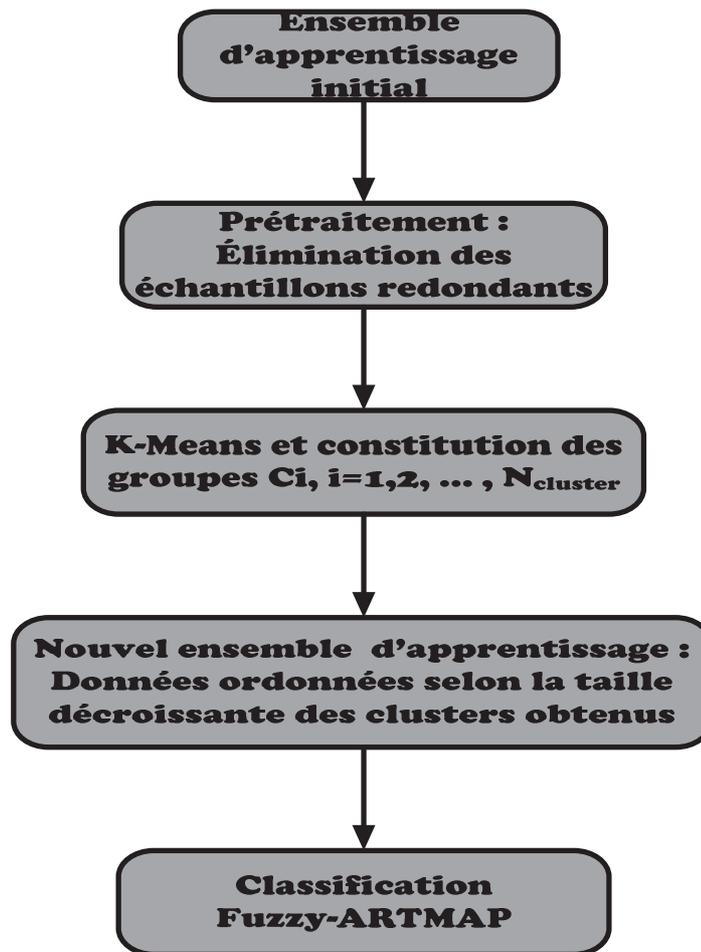


Fig. 3.11 Combinaison fuzzy-ARTMAP K-moyennes

Comme nous pouvons le constater sur la figure (Fig. 3.11), avant de lancer le clustering K-means nous réalisons un prétraitement des données en éliminant les échantillons redondants. Une telle procédure va nous permettre, si les redondances sont importantes, de réduire la complexité de calcul du schéma proposé, sans dégrader la qualité de la classification. La qualité de la classification ne se dégrade pas parce qu'un même hyperrectangle (où plusieurs dans le cas de chevauchement) sera attribué à un échantillon et tous ses redondants (Fuzzy-ARTMAP n'est pas un algorithme statistique).

Une fois que cette tâche est réalisée, nous réalisons un clustering K-means sur ces données. Pour le choix du nombre de clusters, nous avons opté pour l'approche qui propose de définir K égal au nombre de variables disponibles sur la base d'apprentissage divisé par deux. Les clusters résultants seront, ensuite, ordonnés selon leur cardinal, du plus peuplé au moins peuplé.

CHAPITRE 3

Avec cette nouvelle configuration (K-means + Fuzz-ARTMAP), nous allons minimiser le nombre de fois, où, le processus du suivi de correspondance sera exécuté. En effet les entrées sont envoyées par cluster, et comme les distances entre les éléments de ces cluster sont faibles et leur similitude est maximale, le test de vigilance sera rarement non vérifié ce qui réduira l'apparition du phénomène Reset.

3.5 Application en détection et isolation de fautes : application automobile

3.5.1 Description de l'application

Dans cette application, nous avons repris les données utilisées dans le deuxième chapitre. Nous tenons à rappeler que nous disposons de deux types de données, enregistrées par la société (ACTIA), qui concernent deux modes de fonctionnement d'un véhicule, le mode de bon fonctionnement et le mode défaillant.

Il est très intéressant de noter que les variables dont nous disposons sont très hétérogènes, ceci est lisible sur le tableau (2.1). Cette hétérogénéité rend le processus d'apprentissage lourd. Un prétraitement de ces données est primordial pour réussir le schéma que nous avons proposé.

Avant d'exposer les résultats obtenus par l'approche que nous avons proposée, nous présentons d'abord l'amélioration apportée à l'algorithme fuzzy-ARTMAP.

La figure (Fig. 3.12) représente une comparaison de l'évolution de l'erreur de classification des deux algorithmes (Fuzzy-ARTMAP et Fuzzy-ARTMAP + K-moyennes) en fonction du paramètre de vigilance ρ :

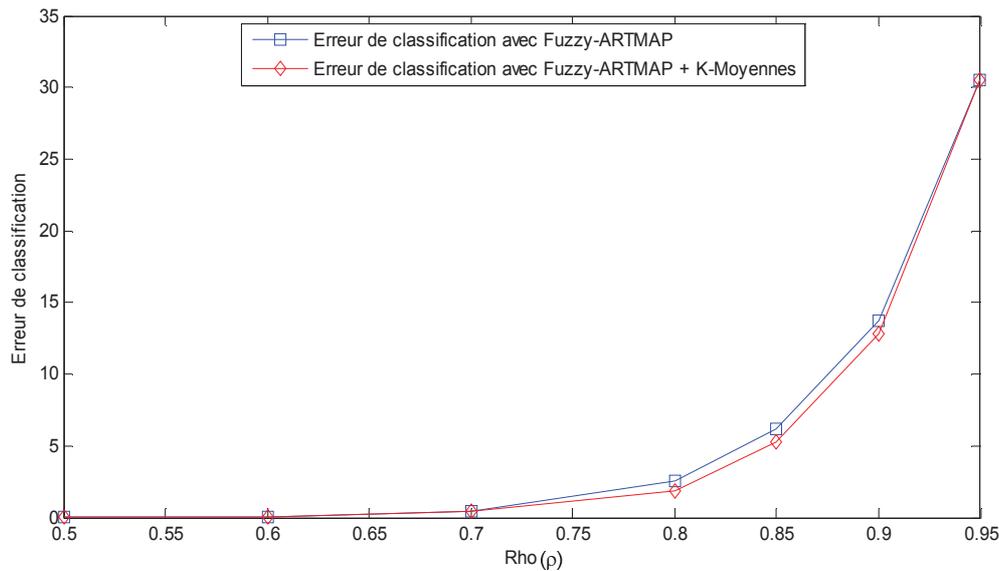


Fig. 3.12 Comparaison de l’erreur de classification (Fuzzy-ARTMAP vs Fuzzy-ARTMAP+K-means)

L’erreur de classification est calculée par :

$$E = \frac{\text{nombre de points mal classifiés}}{\text{nombre de points total}} \times 100\% \quad (3.14)$$

Tout d’abord, nous tenons à préciser que le meilleur rapport « compacité de classes/erreur de classification » est obtenu pour ρ variant entre 0.75 – 0.85. Pour des valeurs inférieures à 0.75, nous avons obtenu un modèle moins compact (de larges hyper-rectangles vérifiés par visualisation en deux et trois dimensions). Pour les valeurs supérieures à 0.85, l’erreur de classification est importante, cela est visible sur la courbe de l’erreur présentée précédemment. Sur cette même figure, nous remarquons qu’en terme d’erreur de classification, la combinaison de K-means avec Fuzzy-ARTMAP est meilleure que l’algorithme Fuzzy-ARTMAP. Cette supériorité est particulièrement visible sur l’intervalle [0.75 – 0.85].

3.5.2 Sélection des variables pertinentes pour FDI

Comme indiqué dans le premier chapitre, il existe plusieurs algorithmes de sélection des variables pertinentes. Parmi ces algorithmes, nous trouvons l’algorithme de sélection en arrière (BFS, pour Backward Feature Selection). Cet algorithme est initialisé sur l’ensemble

CHAPITRE 3

des variables, ensuite, les variables pertinentes sont sélectionnées à chaque itération à l'aide de la procédure de suivante :

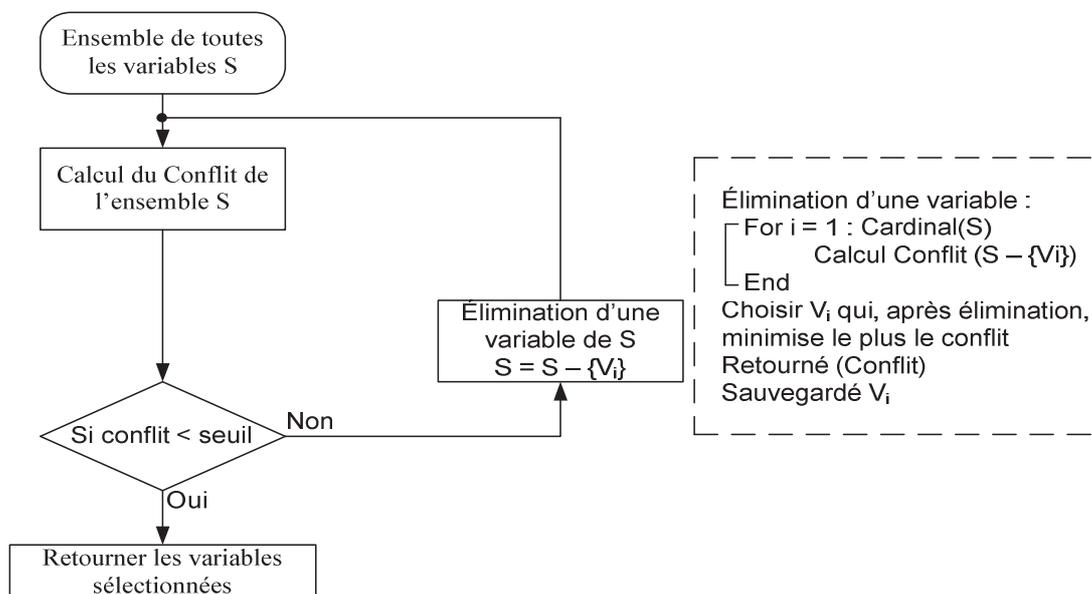


Fig. 3.13 Méthode de sélection des variables

Cet algorithme se base sur l'élimination d'une variable pertinente à chaque itération et l'évaluation de l'ensemble résultant. Si sur cet ensemble, le conflit est supérieur à un seuil, alors ce dernier possède encore des variables pertinentes pour la détection et l'isolation des défauts.

3.5.3 Résultats de l'approche FDI Fuzzy-ARTMAP sur l'application automobile

Dans cette application, une variable est pertinente en détection et isolation de défauts, si son absence provoque une diminution significative sur le conflit entre le contenu informationnel des variables. Sur les figures suivantes, nous exposons les résultats que nous avons obtenus sur les trois pannes décrites précédemment. Dans ces figures, les variables pertinentes sélectionnées sont indiquées dans un rectangle. L'absence de chacune de ces variables provoque une réduction significative du conflit représenté sur l'axe des ordonnées. Dans un premier temps, nous avons considéré les dix variables initialement retenues étant donné les pannes considérées. Avec ces dix variables, le conflit est maximal et égal à 1 pour les trois pannes considérées.

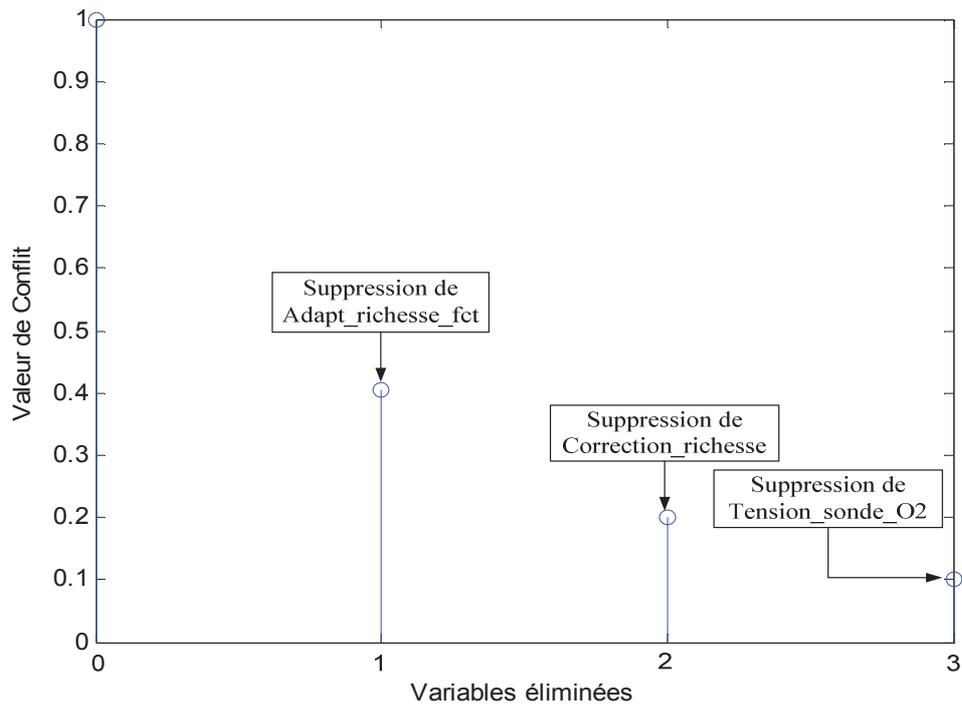


Fig. 3.14 Variables corrélées avec la panne 1

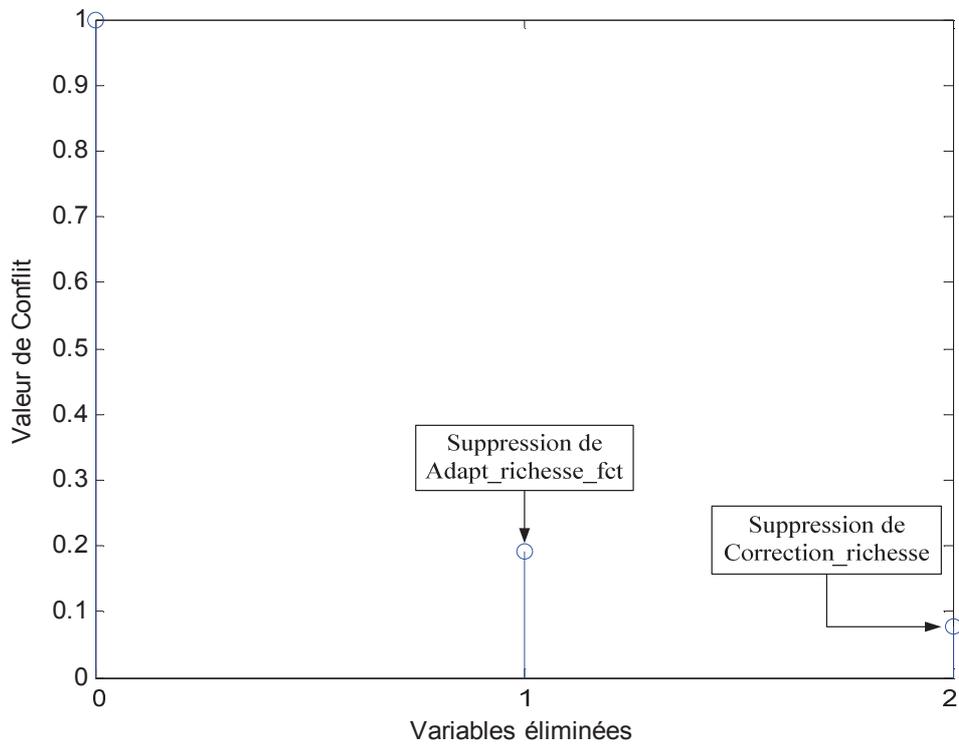


Fig. 3.15 Variables corrélées avec la panne 2

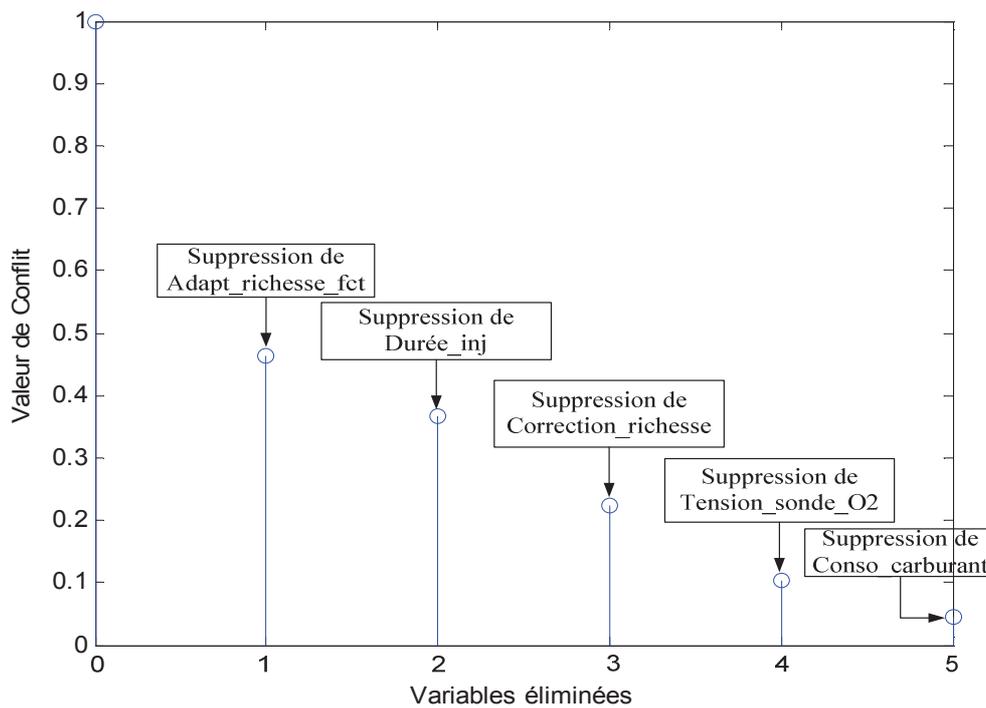


Fig. 3.16 Variables corrélées avec la panne 3

Nous remarquons sur la figure correspondant à la panne 1 (Fig. 3.14), que l'élimination de la variable `Adapt_richesse_fct` permet de réduire le conflit de 1 jusqu'à 0.4. L'élimination de la variable `Correction_richesse` puis de la variable `Tension_sonde_O2` permettra de réduire davantage la valeur de conflit.

Concernant la panne 2 (Fig. 3.15), ce sont les deux premières variables sélectionnées pour la panne 1 qui seront sélectionnées à nouveau.

Plusieurs variables sont sélectionnées lors du traitement de la panne 3 (Fig. 3.16). Tout d'abord, c'est la variable `Adapt_richesse_fct` qui a été sélectionnée. Ensuite, c'est la variable `Durée_inj` qui sera sélectionnée, puis la `Correction_richesse` suivit de la `Tension_sonde_O2`. Enfin, la variable `Conso_carburant` a été sélectionnée.

Sur les trois pannes traitées, nous avons observé que la variable `Adapt_richesse_fct` est toujours pertinente pour la détection de la panne, mais elle n'arrive pas à discriminer les trois pannes. De plus, sa présence maintient le conflit à 1. Donc cette variable est discriminante, à elle seule, des trois pannes considérées dans cette application.

Afin d'étudier le comportement de l'algorithme en l'absence de cette variable complètement discriminante, nous l'avons relancé en éliminant la variable `Adapt_richesse_fct`. Les résultats sont disponibles sur les figures suivantes.

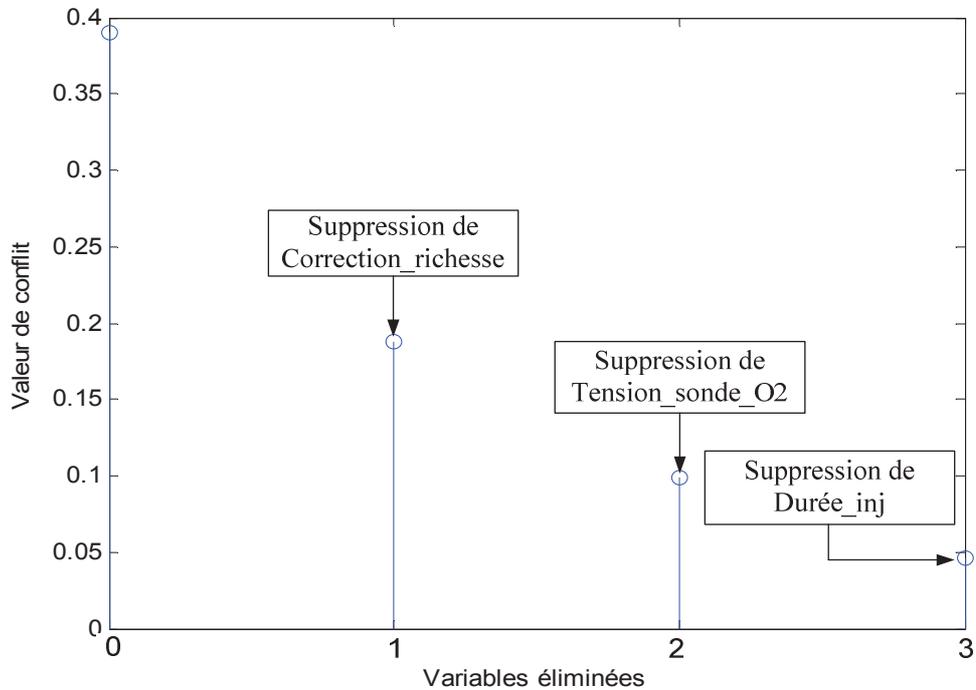


Fig. 3.17 Variables corrélées avec la panne 1 sans `Adapt_richesse_fct`

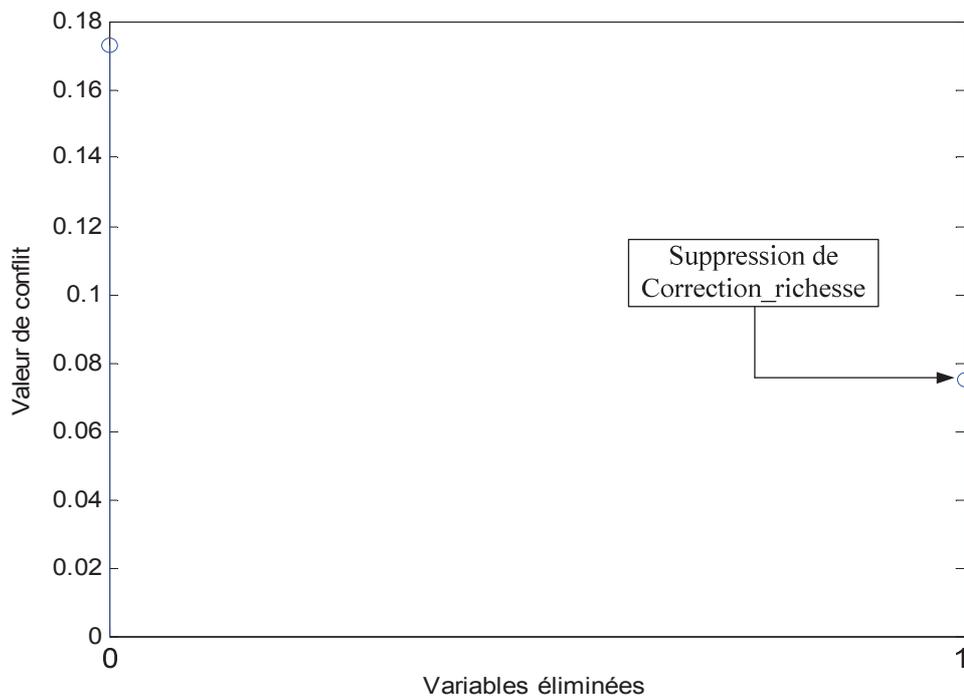


Fig. 3.18 Variables corrélées avec la panne 2 sans `Adapt_richesse_fct`

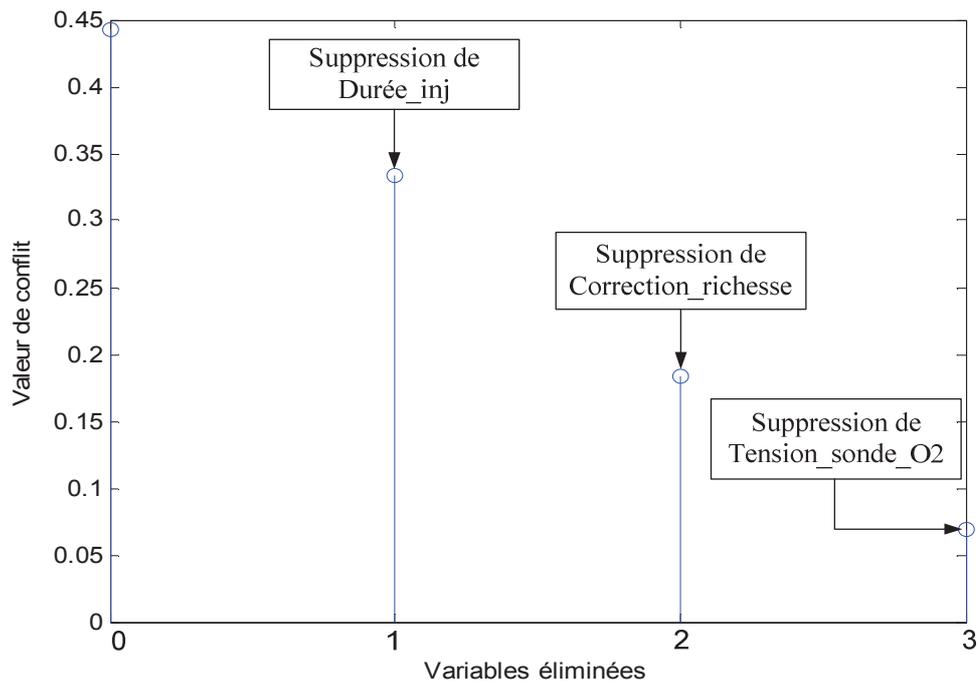


Fig. 3.19 Variables corrélées avec la panne 3 sans Adapt_richesse_fct

Concernant la première panne, nous retrouvons les mêmes variables sélectionnées en présence de la variable `Adapt_richesse_fct`. Nous avons fait le même constat pour la panne 2, où la variable `Correction_richesse` a été sélectionnée. Les variables `Durée_inj`, `Correction_richesse`, et `Tension_sonde_O2` ont été, respectivement, sélectionnées lors de traitement de la panne 3.

Avec ces résultats (Benkaci, et al., 2010) (Benkaci, et al., 2010), nous avons montré l'efficacité du système de détection et d'isolation que nous avons proposé. Tout d'abord, l'algorithme fuzzy-ARTMAP modifié nous permet d'avoir un modèle d'apprentissage global du système étudié. Ce modèle du bon fonctionnement est valable pour les différents défauts considérés. Cette capacité de généralisation est démontrée en l'appliquant sur différentes pannes de l'application automobile. La deuxième caractéristique de la technique que nous avons proposée est sa capacité d'isolation des défauts quelque soit l'influence des variables. Dans cette direction, nous avons remarqué que la présence ou l'absence d'une variable complètement discriminante (`Adapt_richesse_fct`) n'influence pas sur le résultat de l'étude.

3.6 Conclusion

Dans ce chapitre, nous avons démontré l'apport important de la notion du conflit que nous avons défini dans le chapitre précédent. Ce conflit est basé sur l'intersection des contenus informationnels de plusieurs variables afin de déterminer celles liées à la panne ou au défaut surgissant sur le système. Comme indiqué dans le chapitre précédent, la difficulté majeure pour calculer ce conflit est la réalisation d'un partitionnement adéquat permettant de générer des classes hyper-rectangles qui permettent, à leurs tour, de calculer l'intersection informationnelle entre les variables.

Ce partitionnement qui constitue, selon Salzberg, la meilleure méthode de classification de données conduisant à des classes compréhensibles par l'homme et optimale vis-à-vis de l'occupation de l'espace de données a été implémenté à l'aide d'un réseau Fuzzy-ARTMAP. Ce réseau se base sur un contrôle interne géré par le test de vigilance et réalise un apprentissage adaptatif et incrémental permettant d'avoir un bon taux de classification. Afin d'améliorer les résultats du Fuzzy-ARTMAP et de gagner sur la complexité calculatoire, nous avons proposé un prétraitement des données utilisant un clustering K-moyennes pour regrouper les données d'apprentissage. Ces groupes seront envoyés un par un à l'entrée du réseau Fuzzy-ARTMAP. Cette modification nous a permis d'améliorer les performances de l'algorithme Fuzzy-ARTMAP en termes de temps de calcul et en terme d'erreur de classification.

Le processus de détection et d'isolation de fautes que nous avons proposé a été appliqué au domaine automobile. Sur les trois pannes générées sur le véhicule, nous avons d'abord réussi à détecter la présence d'une panne dans le système. Ensuite, nous avons identifié les variables directement liés à la panne considérée avec des pondérations différentes. Dans cette application, nous avons montré qu'il existe des variables qui peuvent être complètement discriminantes, ce qui nous a amené à tester le système en absence de ces variables. Effectivement, nous avons retrouvé les mêmes variables sélectionnées avec la présence des variables complètement discriminantes.

Nous avons ainsi démontré l'efficacité de l'approche proposée, mais, il est intéressant de souligner que ce type de processus FDI nécessite des ressources calculatoires importantes, ce qui rend leur implémentation en temps réel délicate. Dans le chapitre suivant, nous proposons une nouvelle approche dans laquelle nous réalisons un processus FDI ne faisant

CHAPITRE 3

pas appel au traitement sélectif des données, ce qui réduit les ressources calculatoires. Cette approche se base sur la complexité du Kolmogorov et la notion de compression de données.

3.7 Références

ACTIA [En ligne]. - <http://www.actia.com/fr/>.

Aussem A Théorie et applications des réseaux de neurones récurrents et dynamiques à la prédiction, à la modélisation et au contrôle adaptatif des processus dynamiques [Rapport] : Thèse doctorat / Université de Paris 5. - Paris : [s.n.], 1995.

Benkaci M, Doncescu A et Jammes B Feature Selection for Fault Diagnosis using Fuzzy ARTMAP Classification and Conflict Intersection [Conférence] // International Workshop on Diagnosis of Complex Systems (DCS-2010), FIFTH International Conference on Broadband and Wireless Computing, Communication and Applications. - Fukuoka, Japan : [s.n.], 2010.

Benkaci M, Jammes B et Doncescu A Feature Selection for Medical Diagnosis using Fuzzy ARTMAP Classification and Intersection Conflict [Conférence] // Workshops WAINA, IEEE 24th International Conference on Advanced Information Networking and Applications. - Perth, Ustralia : [s.n.], 2010. - pp. 790 – 795.

Benzécri J. P L'analyse des données [Livre]. - Paris : Dunod, 1973.

Bonner R. H On some clustering techniques [Revue] // IBM Journal. - 1964. - Vol. 22.

Carpenter G. A [et al.] Fuzzv ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps [Revue] // IEEE Transactions on Neural Networks. - 1992. - Vol. 3 (5). - pp. 698-713.

Carpenter G. A et Grossberg S A massively parallel architecture for a self-organizing neural pattern recognition machine [Revue] // Computer Vision, Graphics and Image Processing. - 1987 a. - Vol. 37. - pp. 54-115.

Carpenter G. A et Grossberg S ART 2: Self-Organization of Stable Category Recognition Codes for Analog Input Patterns [Revue] // Applied Optics. - 1987 b.

Carpenter G. A et Grossberg S ART 3 hierarchical search: Chemical transmitters in self-organizing pattern recognition architectures [Conférence] // International Joint Conference on Neural Networks. - Washington, DC : [s.n.], 1990. - Vol. 11. - pp. 30-33.

Carpenter G. A et Grossberg S The ART of Adaptive Pattern Recognition by a Self-Organizing Neural Network [Revue] // Computer. - 1988.

Carpenter G. A, Grossberg S et Reynolds J. H ARTMAP: A self-organizing neural network architecture for fast supervised learning and pattern recognition [Conférence] // International Joint Conference on Neural Networks. - Piscataway, N J : IEEE Service Centre, 1991. - pp. 863-868.

- Carpenter G. A, Grossberg S et Rosen D. B** ART-2A: An Adaptive Resonance Algorithm for Rapid Category Learning And Recognition [Revue] // Neural Networks. - 1991 a. - Vol. 4.
- Carpenter G. A, Grossberg S et Rosen D. B.** Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System [Revue] // Neural Networks. - 1991 b. - Vol. 4. - pp. 759-771.
- Celeux G, Diday E et Govaert G** Classification automatique de données environnement statistique et informatique [Livre]. - [s.l.] : Dunod Informatique, 1989.
- Dagher I [et al.]** An Ordering Algorithm for Pattern Presentation in Fuzzy ARTMAP That Tends to Improve Generalization Performance [Revue] // IEEE Transactions on Neural Networks. - 1999. - Vol. 10 (4).
- Dreyfus G [et al.]** Réseaux de neurones : Méthodologie et applications [Revue]. - [s.l.] : Eyrolles, 2ème édition, 2004.
- Grossberg S** Adaptive Pattern Classification and Universal Recoding : I. Parallel Development and coding of Neural Feature Detectors [Revue] // Biological Cybernetics. - 1976. - Vol. 23.
- Lloyd S. P** Least squares quantization in pcm [Revue] // IEEE Transactions on Information Theory. - 1982. - Vol. 28 (2). - pp. 129–137.
- MacQueen J. B** Some methods for classification and analysis of multivariate observations [Conférence] // Proc. of the 5 th Berkeley Symposium on Math. Statistics and probability. - Berkeley : University of California Press, 1967. - Vol. 1.
- Mallat S** A theory for multiresolution signal decomposition : the wavelet transform [Revue] // IEEE Transactions on Pattern Analysis and Machine Intelligence. - 1989. - Vol. 11. - pp. 674-693.
- Mcculloch W. S et Pitts W** A logical calculus of the ideas immanent in nervous activity [Revue] // Bulletin of Mathematical Biophysics. - 1943. - Vol. 5. - pp. 115-133.
- Minsky M et Papert S** Perceptrons [Livre]. - [s.l.] : MIT Press, 1969.
- Powell M. J. D** Radial basis functions for multivariable interpolation : a review [Conférence] // Algorithms for approximation. - 1987. - pp. 143-167.
- Steinhaus H** Sur la division des corp matériels en parties [Revue] // . Bulletin L'Académie Polonaise des Sciences. - 1956. - Vol. 4. - pp. 801–804.

*“When the going gets tough,
the tough relax”*

Dans ce chapitre, nous allons exploiter la notion de complexité de Kolmogorov et, en particulier, la distance normalisée de compression dans un processus de détection et d'isolation de fautes. Afin de démontrer l'apport significatif de cette approche, nous allons l'appliquer sur deux applications différentes : dans la première, nous démontrons la capacité discriminative de l'approche et dans la deuxième nous démontrons sa capacité de détection.

4 Complexité de Kolmogorov pour la détection et l'isolation de défauts

4.1 Introduction

La théorie de la complexité introduite par Andrei Nikolaevich Kolmogorov (Kolmogorov, 1965) a été appliquée aux problèmes de classification d'objets, tels que des textes d'écrivains, des partitions de musiques ou des espèces naturelles (phylogénie). Plusieurs applications ont été réalisées, parmi elles nous pouvons citer les travaux de Levorato (Levorato, 2008) consacrés à la classification prétopologique et les travaux de détection d'anomalies sur des séries temporelles de Keogh et autres (Keogh, et al., 2007).

Dans ce chapitre, nous proposons d'étudier l'apport de la complexité de Kolmogorov aux problèmes de détection et discrimination des défauts. Ces deux propriétés (détection et discrimination) seront étudiées séparément. Dans un premier temps, nous démontrons l'apport de la complexité de Kolmogorov à la discrimination des différentes pannes agissant sur le véhicule. Dans cette première application, nous déterminons un prototype pour chacune des pannes, ensuite, nous calculons, à l'aide de la distance de compression, les similarités existantes entre ces prototypes et les fichiers de chaque panne qui représentent l'historique des défauts enregistrés dans les calculateurs embarqués (logfiles). Concernant la mise en œuvre de cette approche sur la détection des défauts, nous calculons des distances de compression entre la base de données représentant le bon fonctionnement du véhicule et d'autres correspondant aux modes de fonctionnement défectueux. Ces distances vont nous

CHAPITRE 4

permettre de détecter la présence d'un défaut. Dans l'approche que nous proposons, la complexité de Kolmogorov est estimée par la compression des données. Il existe des métriques qui permettent de calculer des distances de similarité entre différents fichiers après compression.

4.2 Mesure de la complexité

Il existe deux théories de la complexité : la théorie de la complexité algorithmique et la théorie de la complexité stochastique. La première mesure l'information contenue dans une séquence par la longueur du plus petit programme qui génère cette séquence, tandis que la théorie de la complexité stochastique évalue la quantité d'information contenue dans une séquence par la difficulté de prédire un bit à partir de ceux qui le précèdent. Avant d'aborder la complexité de Kolmogorov, nous donnons d'abord quelques définitions nécessaires pour la suite de ce chapitre.

4.2.1 Représentation des séquences

N'importe quelle séquence peut être définie comme une suite de symboles ou de nombres. Dans le cas où la séquence est composée d'une suite de symboles, ils seront tirés d'un alphabet. Par exemple, dans un texte, ces symboles se suivent dans un ordre défini pour constituer une séquence composée d'un nombre fini de symboles. De son côté, l'alphabet est un ensemble énumérable, au sens mathématique du terme. Par exemple, l'ensemble des nombres naturels \mathbb{N} représente un alphabet. Voici quelques définitions utiles (Pigeon, 2001):

Définition 4-1 : Alphabet fini

Soit un alphabet $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_n\}$, une énumération finie de symboles arbitraires. Nous qualifions cet alphabet de partiellement ordonné si la relation suivante est vérifiée : $\sigma_1 \leq \sigma_2 \leq \dots \leq \sigma_n$. Le même alphabet est qualifié de strictement ordonné si $\sigma_0 < \sigma_1 < \dots < \sigma_n$. Cet ordre est appelé ordre lexicographique des symboles. $S = \{a, b, c, d, \dots, z\}$ est un exemple d'un alphabet fini.

Définition 4-2 : Alphabet infini

Soit un alphabet $\Sigma = \{\sigma_1, \sigma_2, \dots\}$, une énumération infinie de symboles arbitraires. Nous qualifions cet alphabet de partiellement ordonné si la relation

suivante est vérifiée : $\sigma_0 \leq \sigma_1 \leq \dots$. Le même alphabet est qualifié de strictement ordonné si $\sigma_0 < \sigma_1 < \dots$. Cet ordre est appelé ordre lexicographique des symboles. Nous pouvons citer comme exemple l'ensemble des nombres naturels \mathbb{N} et l'ensemble des nombres réels \mathbb{R} .

Définition 4-3 : Séquence

La séquence est une suite de symboles ordonnés et finie, inspirée d'un alphabet de référence.

Définition 4-4 : Longueur de séquence

La longueur d'une séquence S représente le nombre de symboles la composant. Cette longueur est définie par $|S|$.

Définition 4-5 : Ensemble universel

L'ensemble universel de l'alphabet S renferme tous les « mots » formés des lettres de S de longueur $0,1,2,3,\dots$. Posons S^n , le produit cartésien de S avec lui-même n fois. S^0 représente le « mot vide », noté \perp . Soit $S^* = S^0 \cup S \cup S^2 \cup S^3 \cup \dots$, l'ensemble universel de S . Toutes les composantes de l'ensemble universel ne sont pas de longueur infinie. Exemple : Si $S = \{0,1\}$, $S^* = \{\perp, 0,1,00,01,10,11,000,001,010,100, \dots\}$.

Définition 4-6 : Concaténation

La concaténation de deux mots a et b est représentée par $(a : b)$. si $a, b \in \Sigma^*$ alors $(a : b) \in \Sigma^*$. La concaténation n'est pas réversible, c'est-à-dire, la connaissance de $(a : b)$ ne permet pas toujours de retrouver a et b , elle respecte : $|(a : b)| = |a| + |b|$.

4.3 La Complexité de Kolmogorov

4.3.1 Introduction

Cette théorie a émergé dans les années soixante, elle a été introduite dans les travaux de Solomonoff, Kolmogorov et Chaitin. L'origine de cette théorie remonte aux tentatives faites pour définir l'incohérence d'un mot binaire. Dans ses travaux, Solomonoff (Solomonoff, 1964 a) (Solomonoff, 1964 b) propose la théorie de la complexité et de l'incohérence d'un ensemble fini d'objets. Cette théorie est appelée actuellement complexité de Kolmogorov ou

CHAPITRE 4

complexité algorithmique. L'idée de cette théorie réside dans le fait qu'un objet fini n'est pas aléatoire si une part de l'information qu'il contient est redondante. Alors, un tel objet peut avoir une description plus compacte que sa description d'origine, en éliminant les redondances.

Dans cette section, nous commençons par traiter la théorie des probabilités classiques. Cette théorie est, en général, incapable de décortiquer quelques phénomènes intrinsèques aux séquences. Pour palier cette insuffisance, il est plus que nécessaire d'introduire une théorie qui, au lieu de s'intéresser à la probabilité d'occurrence d'une séquence particulière, s'intéressera à la structure interne de la séquence. Plusieurs définitions de l'aléatoire sont proposées, nous pouvons citer celle de Von Mises qui qualifie une séquence S d'aléatoire si la connaissance de ses $(n-1)$ bits ne permet pas de prédire son $n^{ième}$ bit. Cette définition de l'aléatoire ne tient pas compte de la structure interne de la séquence étudiée.

Il est connu que l'étude et la prédiction des systèmes complexes, qui sont caractérisés par un grand nombre de constituants en interactions, se réalise à l'aide des sciences de la complexité. Une des caractéristiques majeures des systèmes complexes est qu'ils ont un comportement holistique qui peut être aléatoire ou organisé.

La théorie de l'information s'est révélée efficace pour élucider l'opposition entre la complexité aléatoire et organisée. Deux orientations se sont développées : la logique mathématique en proposant une définition claire de la notion d'algorithme et la modélisation statistique.

L'objectif de la modélisation statistique est d'extraire l'information à partir d'un ensemble de données. L'idée d'information que nous développons par la suite n'est pas celle de Shannon ou Fisher, mais celle liée à une machine qui génère ces mêmes données sous forme de chaînes de caractères. Nous parlons dans ce cas de la complexité des données. Le but du calcul de la complexité est d'établir des classes en se basant sur à la difficulté rencontrée par un algorithme dans le processus d'extraction d'information utile. L'information utile est liée aux propriétés régulières contenues dans les données. Le problème est de trouver une méthode qui mesure la quantité d'information apprise par un modèle sélectionné. La théorie de la complexité via la statistique de Kolmogorov, permet de mesurer cette quantité d'information.

La complexité de Kolmogorov, $K(S)$, est liée à l'entropie de Shannon, dans le sens où la valeur de $K(S)$ d'une séquence aléatoire est approximativement égale, à une constante près, à l'entropie de la source d'information qui génère cette séquence. La complexité de Kolmogorov diffère de l'entropie parce qu'elle caractérise plutôt une chaîne de caractères qu'une source d'information. On dit qu'une chaîne est complexe quand sa complexité n'est pas trop différente de sa longueur réelle.

D'un point de vue pratique l'information traitée par un ordinateur est digitalisée, donc, il est pertinent de considérer chaque objet comme une chaîne finie de 0 et de 1. $K(S)$ correspond alors à la longueur minimale d'un programme qui peut générer cette chaîne spécifiée.

Sachant que toutes les chaînes de caractères peuvent être codées en binaire nous pouvons considérer qu'il est possible d'en déterminer la complexité de Kolmogorov.

Pour donner une idée de cette notion, considérons une chaîne de longueur n : $S_n=0101010...01$. Sa complexité est faible parce que nous pouvons écrire un programme p qui affiche 0 et 1 n fois. Une autre façon est de voir p comme la forme la plus réduite de S , $K(S)$ devient alors la limite inférieure d'une possible compression de S . Les chaînes aléatoires ne sont donc pas compressibles en général, ce qui explique pourquoi il n'existe toujours pas d'algorithme de compression qui s'applique uniformément à toutes les chaînes et qui compresse sans pertes.

4.3.2 Complexité de Kolmogorov et Machines du Turing

Les machines universelles de Turing peuvent être comparées par l'intermédiaire de la longueur des programmes qu'elles génèrent. Il existe une correspondance entre les machines universelles de Turing et la complexité de Kolmogorov d'une chaîne donnée. En effet la complexité de Kolmogorov entre deux machines diffère d'une constante connue ou calculable. Nous attirons l'attention sur le fait que pour une chaîne S donnée, $K(S)$ est un nombre qui ne peut pas être calculé parce qu'il représente une limite idéale correspondant à une machine idéale : **La Machine Universelle de Turing**. Cette difficulté majeure peut être surmontée en utilisant la longueur de n'importe quel programme produisant S comme une limite supérieure de $K(S)$.

La théorie de la complexité algorithmique, comme citée précédemment, se base sur le calcul de la longueur du plus petit programme qui permet de générer la séquence étudiée. Par

CHAPITRE 4

conséquence, nous pouvons conclure que la théorie de la complexité s'appuie elle-même sur la théorie de la calculabilité introduite par Allan Mathison Turing est basée sur sa machine théorique qui sera nommée la machine de Turing. La figure suivante représente une machine de Turing :

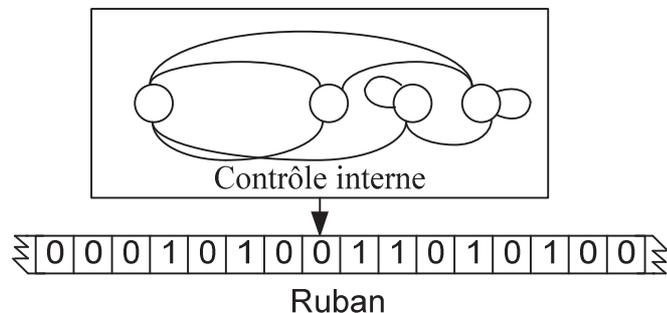


Fig. 4.1 Machine de Turing

Cette machine de Turing est une description idéalisée d'un ordinateur général (Turing, 1937). Elle utilise un ruban comme une mémoire qu'elle peut dérouler à gauche ou à droite pour écrire ou lire un symbole. Le nombre d'états distincts que peut prendre l'opération de contrôle de la machine est fini. Dans sa définition, Turing considère que tout problème effectivement calculable (l'existence d'un algorithme qui permet de calculer la solution), l'est sur une machine de Turing.

En résumé, la théorie de la complexité algorithmique des séquences est basée sur la longueur du plus petit programme (fonction calculable sur une machine de Turing) qui produit une séquence donnée. La séquence est qualifiée de « facile » si le programme est plus court qu'elle-même, ou de « difficile » si le programme est aussi long que la séquence elle-même. Ce dernier cas caractérise une séquence aléatoire.

A présent, nous donnons quelques définitions concernant la complexité du Kolmogorov repris de la thèse de Pigeon citée précédemment.

Définition 4-7 : Complexité de Kolmogorov

Considérons que ϕ est une machine de Turing, et p est un programme à exécuter sur ϕ . La complexité de Kolmogorov d'une séquence S est donnée par :

$$K_{\phi}(S) = \min\{|p| \mid \phi(p) = S\} \quad (4.1)$$

En d'autres termes, $K_\phi(S)$ est la longueur du plus petit programme p exécuté sur la machine de Turing qui produira la séquence S . Il est essentiel de souligner que le programme p n'utilisera aucune information extérieure pour produire S .

Définition 4-8 : Complexité conditionnelle de Kolmogorov

La complexité conditionnelle de Kolmogorov d'une séquence S_1 connaissant S_2 est donnée par l'équation suivante :

$$K_\phi(S_1 | S_2) = \min\{|p| \mid \phi(p, S_2) = S_1\} \quad (4.2)$$

C'est la longueur du plus petit programme p qui en s'appuyant sur l'information de la séquence S_2 générera la séquence S_1 .

La complexité inconditionnelle et la complexité conditionnelle sont liées par l'inégalité suivante :

$$K_\phi(S_1 | S_2) \leq K_\phi(S_1) \quad (4.3)$$

D'après ces équations, nous remarquons que la complexité d'une séquence donnée dépend fortement de la qualité intrinsèque de la machine ϕ .

La complexité de la concaténation est régie par l'équation suivante :

$$K_\phi((S_1 : S_2)) \leq K_\phi(S_1) + K_\phi(S_2) \quad (4.4)$$

Définition 4-9 : Le paradigme d'incompressibilité

Un objet fini est aléatoire s'il n'existe pas une description plus compacte que sa description initiale.

4.3.2.1 Propriétés de la complexité de Kolmogorov (Pigeon, 2001)

4.3.2.1.1 La description minimale

Elle indique que la description d'un objet est certainement plus restreinte que l'objet lui-même.

Proposition 1 : $\exists c \in \mathbb{R}, \forall x \in \{0,1\}^*; K(x) \leq |x| + c$

Démonstration : Dans le cas contraire, l'hypothèse que $K(x)$ est minimal est contredite par $|x|$ plus court.

CHAPITRE 4

Soit x un mot binaire, ϕ la machine de Turing universelle (ϕ s'arrête pour tout mot en entrée et donne ce même mot en sortie). Alors (ϕ, x) est une description évidente de x . Si $c = |\langle \phi \rangle| + 2$ alors $|\langle \phi, x \rangle| = |x| + c$. Par définition de la minimalité de $K(x)$ nous avons l'inégalité recherchée.

4.3.2.1.2 Notion de répétition

La propriété suivante indique que la description d'une répétition n'est pas une répétition de la description.

Proposition 2 : $\exists c \in \mathbb{R}, \forall x \in \{0,1\}^*; K(xx) \leq K(x) + c$

Démonstration : Soit x un mot binaire et (ϕ, w) une description de x tel que $K(x) = |\langle \phi, w \rangle|$. Soit également \mathcal{M} une machine de Turing avec une entrée $\langle p, v \rangle$, où (p, v) décrit un mot s , et renvoie ss en sortie. Alors $(\mathcal{M}, \langle \phi, w \rangle)$ est donc une description de xx de longueur $|\langle \mathcal{M}, \langle \phi, w \rangle \rangle| = k(x) + c$ où $c = |\langle \mathcal{M} \rangle| + 2$.

4.3.2.1.3 Optimalité de la description

Nous pouvons légitimement nous demander si la description *via* les machines de Turing est optimale, ou si un autre langage de description pourrait assurer une meilleure compressibilité. Le théorème suivant répond à cette interrogation. Nous appelons désormais langage de description toute fonction $P: \{0,1\}^* \rightarrow \{0,1\}^*$ calculable. Nous notons K_p la complexité de Kolmogorov associée à ce langage, c'est-à-dire, $K_p(x) = \min\{|s| \mid s \in \{0,1\}^*, p(s) = x\}$.

Théorème 1 : Soit p un langage de description : $\exists c_p \in \mathbb{R}, \forall x \in \{0,1\}^*, K(x) \leq K_p(x) + c_p$

Démonstration : Tout vient du fait que p est calculable. En effet, il existe une machine de Turing ϕ_p , qui pour $s \in \{0,1\}^*$ en entrée renvoie le mot $p(s)$.

Soit alors s et x deux mots binaires tel que $K_p(x) = |s|$ et $p(s) = x$ (s est donc une description de x relativement à p). Alors (ϕ_p, s) est une description de x . Et si $c_p = |\langle \phi_p \rangle| + 2$, nous avons $|\langle \phi_p, s \rangle| = K_p(x) + c$. Nous avons finalement, par minimalité de $K(x)$: $K(x) \leq K_p(x) + c_p$.

4.3.2.1.4 Approximation du K

Nous savons que K fournit une information quant à la compressibilité d'une chaîne binaire. C'est un outil qui pourrait donc s'avérer primordial quand au stockage de données. Si K ne peut pas fournir cette information, alors elle n'est pas calculable. Voici un lemme intéressant pour la suite :

Lemme 1 : Pour toute fonction f , il existe $c \in \mathbb{R}$ tel que : $\forall x \in \{0,1\}^*, K(f(x)) \leq K(x) + C$

Théorème 1 : K n'est pas calculable

Démonstration : Afin de démontrer cela, nous utilisons le formalisme du paradoxe bien connu « du plus petit entier naturel qui ne peut être décrit en français en moins de vingt mots ».

Supposons que K est calculable. Alors la fonction f , définie sur \mathbb{N} par $f(n) =$ premier mot x dans l'ordre lexicographique tel que $K(x) > n$, l'est également (cette fonction est bien définie car le nombre de mots dont la description est de longueur inférieure ou égale à n est au plus 2^n).

D'abord, nous avons, par définition, $K(f(n)) > n$ pour tout $n \in \mathbb{N}$. Nous avons également, par le lemme 1, $c \in \mathbb{R}$ tel que $K(f(n)) \leq K(n) + c$. De plus, la propriété de description minimale donne $c' \in \mathbb{R}$ tel que $K(n) \leq |n| + c'$.

Comme $|n| = \lfloor \log_2(n) \rfloor$ (n est confondu avec sa représentation binaire), nous avons finalement une constante $d \in \mathbb{R}$ tel que $\forall n \in \mathbb{N}, n < \log_2(n) + d$. Ce qui est absurde.

4.3.2.2 Exemple

Soit la séquence suivante : 1111111 ... 1111 une séquence de n fois 1. Cette séquence est clairement non aléatoire, elle représente une suite de 1. Le seul inconnu cette séquence est sa taille n . Le programme simple qui permet de générer les séquences de cette classe à partir d'un nombre n est donné par :

```
Séquence(n)
{
  For (i=1 ; i<n ; i++) return (1)
}
```

Ce programme exige une trentaine de symboles pour être écrit et prend le paramètre n en entrée. Dans ce cas très simple, nous remarquons qu'il n'existe pas de moyen plus simple pour générer ce type de séquence. Par contre, la séquence suivante

CHAPITRE 4

0010100111001000110100001111, paraît plus difficile à générer. L'objectif est alors de chercher le programme le plus court possible qui génère une telle séquence. Supposons que ce programme soit donné par la fonction p qui est calculable au sens de Turing. Ce programme se base sur une machine de Turing ϕ spécialement conçue en vue d'une implémentation efficace du programme p .

4.3.3 Estimation de la Complexité de Kolmogorov par la compression

4.3.3.1 Introduction à la compression

Si la mesure exacte de la Complexité de Kolmogorov n'est pas possible, nous pouvons l'approximer par une borne inférieure en utilisant les différentes formes compressibles de S . Il est clair que chaque algorithme de compression nous donne une estimation donnée de la chaîne S .

La compression de données est représentée par l'ensemble des méthodes que nous utilisons pour réduire la taille d'un message suffisamment long, sans ou avec perte d'information. Cette opération est un usage quotidien qui n'est pas spécifique aux applications technologiques. Comme exemples de ces opérations quotidiennes, nous pouvons citer le fait d'utiliser des abréviations pour prendre des notes manuscrites plus rapidement, ou l'utilisation d'un langage simplifié pour envoyer un SMS.

Parmi les méthodes de compressions les plus anciennes, nous pouvons citer le code Morse. Introduit par Samuel Finley Breese Morse (1791 – 1872), ce code a été conçu en 1837, mais, il n'a connu le succès qu'en 1845. Les messages traduits en utilisant le code Morse étaient transmis via les télégraphes, un appareil électromécanique capable de représenter deux états (ouvert ou fermé). Ce code est composé de trois caractères différents : le point « . », le trait « – » et la « pause » pour délimiter les codes. Ce code a été utilisé par les prisonniers américains lors de la guerre du Vietnam pour échanger leurs messages. Ces prisonniers, au lieu d'utiliser un télégraphe, utilisent le tapotement des doigts contre le mur de la cellule. Il existe également un code ancien qui, en se basant sur un tableau d'alphabet sous forme de matrice, envoie à la place de la lettre la position de la lettre dans la matrice. Dans ce code, le numéro de la rangée est envoyé en premier, suivi d'une pause, puis du numéro de la colonne. Ce code facile mais peu efficace peut être amélioré en utilisant une série d'abréviations (par exemple, GNST : Good Night, Sleep Tight).

Les codes cités précédemment ne prennent pas en compte la fréquence d'occurrence de chaque symbole. Il est possible d'améliorer les performances de ces méthodes en utilisant moins de frappes pour coder les lettres les plus fréquentes, et plus pour les lettres les moins fréquentes.

Après avoir présenté la chronologie et l'historique de la compression des données, il est très utile de répondre à la question suivante : pourquoi la compression ? Avec les avancées technologiques, l'utilisation des codes cités précédemment ne sont plus d'actualité. L'introduction de l'informatique, et les échanges par courriers électroniques et l'internet ont impliqué la manipulation de quantités d'informations colossales. Les informations échangées, comme les documents textes, les images, la musique et les vidéos, sont en général très riches en connaissances utiles, mais nécessitent de puissantes ressources de calcul et des moyens de transmission très performants. Pour réduire ces exigences, la réduction de la taille des informations est plus que nécessaire.

4.3.3.2 Les domaines d'application de la compression

Il existe plusieurs applications technologiques de notre époque qui utilisent la compression de données pour améliorer leurs performances. Nous trouvons parmi ces applications, les télécommunications. Dans ce sens, tout moyen de communication est défini principalement par sa bande passante qui définit le nombre de *bits* par seconde que nous pouvons transmettre. Ce nombre de *bits* par seconde est limité par les caractéristiques physiques du moyen de transmission. La compression est plus que nécessaire à la fois pour les modems relativement lents ainsi que les médias à grande vitesse, comme les réseaux locaux ou même internet qui n'est d'ailleurs pas toujours très rapide. Dans ces applications, les temps de transferts sont inversement proportionnels au taux de compression.

Une autre application qui utilise des techniques de compression est celle de stockage des données. Un moyen de stockage de données est défini par sa capacité, c'est-à-dire, le nombre de *bits* qu'il peut emmagasiner. Donc, nous remarquons que l'utilisation des techniques de compression est plus que nécessaire dans ces applications et permettent d'augmenter la quantité d'information stockée et de remédier à la surcapacité introduite par les techniques de correction d'erreur. De plus, les techniques de compression de données permettent de créer des archives qui représentent des copies de sécurité structurées. Ces techniques de compression permettent de gagner à la fois un temps de transfert considérable

CHAPITRE 4

et une puissance de calcul minimale si ces fichiers voyagent dans un réseau sous forme compressée.

Les techniques de compression de données ont été appliquées à d'autres domaines comme les applications de poche (dictionnaire électronique de poche, lecteur MP3, GPS etc...) et les applications liées au monde des jouets. Dans notre travail, nous utiliserons la compression de données comme moyen de détection et d'isolation de défauts.

4.3.3.3 Compression de données

Il existe des méthodes de compression de données sans perte d'information et d'autres avec perte d'information. Les méthodes de compression sans perte d'information permettent de restituer l'information totale compressée, tandis que les méthodes avec perte permettent la restitution de l'information significative. Ce critère de catégorisation est le plus utilisé pour classer les méthodes de compression. Dans cette thèse, comme elle n'est pas consacrée spécifiquement à la compression de données, nous nous contenterons de présenter ces deux types de compression de données.

4.3.3.3.1 Compression sans perte d'information

La compression sans perte (lossless compression) exige la récupération totale de l'information. Elle peut être résumée comme suit : supposons que nous avons un fichier X à compresser, la compression de X nous donne un fichier Y de taille inférieure. Si la compression de X est sans perte, alors, la décompression de Y doit permettre de retrouver le fichier initial. Ces méthodes de compression permettent, lors de la décompression, de préserver l'information originale contenue dans la source. La figure suivante résume la compression sans perte.

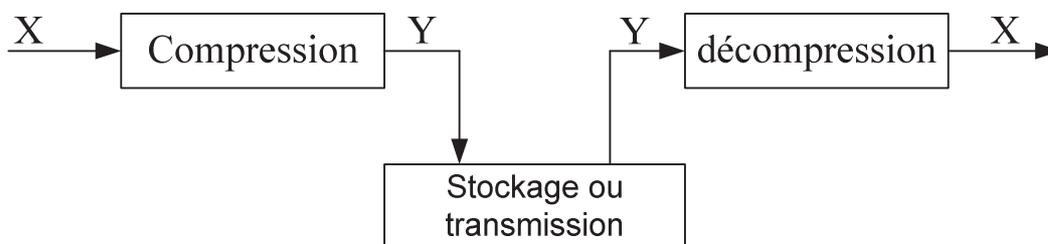


Fig. 4.2 Compression de données sans perte d'information

4.3.3.3.2 Compression avec perte d'information

Les méthodes de compression avec perte d'information (lossy compression), sélectionnent l'information utile contenue dans la source pour assurer un meilleur taux de compression. Ces méthodes sont, en général, appliquées dans le cas où la redondance d'information est forte. L'opération de compression avec perte d'information peut être résumée comme suit : supposons que nous avons un fichier source X à compresser, la compression de X nous donne un autre fichier Y de taille inférieure à celle de X , et la décompression de Y permet de récupérer un nouveau fichier Z , tel que, $Z \approx X$. Dans certains algorithmes de compression avec perte, des parties de la source à compresser sont complètement supprimées. Ce type de méthodes a certainement des taux de compression très intéressants, mais il existe un compromis entre ce taux de compression et la qualité de la compression résultante. Parmi les méthodes de compression avec perte d'information nous pouvons signaler la compression d'image, de son, et la compression par les ondelettes. La figure suivante représente le schéma général des méthodes de compression avec perte d'information.

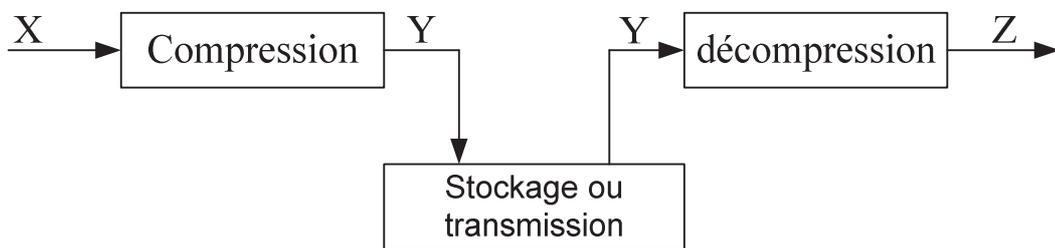


Fig. 4.3 Compression de données avec perte d'information

4.3.3.4 Les méthodes de compression

Il existe plusieurs techniques utilisées dans la compression de données.

Tout d'abord, il faut noter l'existence des méthodes *ad hoc* (méthode Hall, compression RLE, compression Bitpack etc...) construites pour des sources de données particulières qui donnent, en général, de mauvais résultats sur d'autres sources d'information. Ces méthodes utilisent souvent toute l'information intuitive spécifique aux données d'intérêt et produisent ainsi des performances acceptables.

CHAPITRE 4

Il existe trois types de techniques générales de compression sans perte d'information ; les algorithmes à base de dictionnaire, les algorithmes à base de transformation et les algorithmes à base de modélisation statistique.

Les algorithmes à base de dictionnaire utilisent un ensemble de mots enregistrés, ces mots seront ensuite indexés. Lors de la compression, si un mot donné n'existe pas dans le dictionnaire, il sera rajouté, sinon, ce mot sera remplacé par son index. Si tous les mots sont indexés, nous obtiendrons une compression optimale.

Les algorithmes se basant sur la transformation imposent une nouvelle structure de données pour mettre en avant les répétitions existantes pour les exploiter au maximum lors de la compression. Ces transformations représentent une phase de prétraitement pour ensuite appliquer la technique du dictionnaire ou la technique de modélisation statistique qui se base sur un calcul de probabilités pour réaliser une bonne compression.

Les techniques à base de modélisation statistiques cherchent à estimer la fonction de distribution de la prochaine donnée pour en générer un code efficace. Elles, même si elles mènent à une compression optimale, sont caractérisées par une complexité accrue et exigent des ressources importantes.

De l'autre côté, les méthodes de compression avec perte reposent sur des techniques de décomposition et de discrétisation du signal, et sur des méthodes de discrétisation des scalaires et des vecteurs. Les techniques de décomposition reposent sur les méthodes d'analyse de fréquentielle comme la transformée de Fourier ou les méthodes de décomposition en ondelettes. Dans cette discrétisation, nous réduisons la précision d'une manière intelligente pour ne pas pénaliser l'opération de décompression.

4.3.3.5 Calcul de la complexité par compression

Pour calculer la complexité, nous disposons d'un nombre considérable d'outils de compression : Lempel-Zev (LZ), une combinaison de LZ et du codage de Huffman (ZIP et gzip), ou la combinaison entre l'algorithme de Burrows-Wheeler et le codage de Huffman (bzip2). Il existe également l'outil de compression PAQ, avec lequel, nous avons obtenu les meilleurs taux de compression. L'inégalité suivante représente la relation d'ordre entre la vraie valeur de la complexité de Kolmogorov et l'estimation de celle-ci par la compression

par l'algorithme PAQ (à noter que nous avons utilisé la version PAQ8a qui donne les meilleurs taux de compression): soit x un fichier à compresser

$$0 < K(x) < PAQ8a(x) < bzip2(x) < gzip(x) \quad (4.5)$$

Cette relation nous permet d'utiliser la taille d'un fichier compressé comme approximation de $K(x)$. Il est évident que pour une meilleure estimation de $K(x)$ nous devons appliquer de très bonnes méthodes de compression sans pertes. Avant de détailler l'algorithme de compression choisi, nous définissons un compresseur normal.

Définition 4-10 : Compresseur normal

Un compresseur est dit normal s'il possède les propriétés suivantes :

- **Monotonie** : $C(xy) \geq C(x)$
- **Idempotence** : $C(xx) = C(x)$ et $C(\gamma) = 0$ avec γ est un mot vide
- **Symétrie** : $C(xy) = C(yx)$
- **Distribution** : $C(xy) + C(z) \leq C(xz) + C(yz)$

4.3.3.6 Compresseur PAQ

4.3.3.6.1 Algorithme

Vu la non disponibilité d'une documentation détaillée sur l'algorithme de compression de type PAQ, nous nous contenterons de reproduire quelques détails techniques sans les approfondir. L'algorithme PAQ se base sur l'approche du mélange de contexte qui réalise la compression en deux étapes ; la prédiction et le codage arithmétique. Cette approche est similaire à celle de prédiction par correspondance partielle (PPM, Prediction by Partial Matching). Cependant, leurs divergences résident dans la manière de prédire le prochain symbole. Dans l'approche de mélange des contextes, la prédiction est réalisée en se basant sur la combinaison pondérée des estimations de probabilité issues de plusieurs modèles conçus dans différents contextes. L'algorithme de mélange de contexte, sur lequel se base le compresseur PAQ, surclasse l'algorithme PPM dans le fait que le contexte n'est pas nécessairement dans le voisinage (morceaux dispatchés). Pour mieux comprendre l'algorithme de base du PAQ, il est nécessaire que nous définissions la notion de contexte.

Définition 4-11 : Contexte

Consiste à rassembler un certain nombre d'unités (bits) pour constituer une unité de mesure pour la prédiction.

Il existe plusieurs contextes qui sont utilisés dans l'algorithme PAQ, par exemple :

- ***N-octet*** : le contexte est constitué des N dernières octets avant le symbole à prédire.
- ***Contexte Creux*** : Choisir des octets aux dépens d'autres.
- ***Contexte analogique*** : représente les bits significatifs des 8 ou 16 bits.

4.3.3.6.1.1 Modèle de mélange du contexte

Cette technique a été utilisée par les compresseurs du type PAQ. Dans cette technique les données d'entrée sont considérées comme une série temporelle de bits. Pour chaque bit, chaque modèle renvoie indépendamment deux nombres, n_0 et n_1 . Ces deux nombres représentent, respectivement, la mesure d'évidence pour que le prochain bit soit, respectivement 0 ou 1. En d'autres termes, ces deux nombres représentent la confirmation, par le modèle, que le prochain bit sera 0 avec une probabilité n_0/n ou 1 avec une probabilité n_1/n tel que le nombre $n = n_0 + n_1$ représente la confiance relative du modèle de prédiction. En considérant que les modèles sont indépendants, la confiance générée est significative si nous comparons deux prédictions d'un même modèle. De ce fait, les modèles seront combinés avec une somme pondérée des n_0 et n_1 de chaque modèle. La formalisation de cette opération est donnée par :

$$S_0 = \varepsilon + \sum_i w_i n_{0i} \quad (4.6) \text{ représente l'évidence de 0}$$

$$S_1 = \varepsilon + \sum_i w_i n_{1i} \quad (4.7) \text{ représente l'évidence de 1}$$

$$S = S_0 + S_1 \quad (4.8) \text{ l'évidence totale}$$

$$p_0 = \frac{S_0}{S} \quad (4.9) \text{ la probabilité que le prochain bit soit 0}$$

$$p_1 = \frac{S_1}{S} \quad (4.10) \text{ la probabilité que le prochain bit soit 1}$$

$w_i \geq 0$, représente le poids du $i^{\text{ème}}$ modèle, n_{0i} et n_{1i} sont les sorties n_0 et n_1 du $i^{\text{ème}}$ modèle, ε est une constante garantissant que : $S_0, S_1 > 0$ et $0 < p_0, p_1 < 1$.

4.3.3.6.1.1 Actualisation des poids

Une fois que tous les bits de la donnée à traiter sont codés, les poids seront ajustés de façon à favoriser les modèles améliorant la précision de la prédiction du bit en question. Soit x récemment codé, l'équation de la mise-à-jour des poids est la suivante :

$$w_i = \max \left[0, w_i + \frac{(x - p_1)(S \cdot n_{1i} - S_1 \cdot n_i)}{S_0} \cdot S_1 \right] \quad (4.11)$$

Avec $n_i = n_{0i} + n_{1i}$. Le terme $(x - p_1)$ représente l'erreur de prédiction.

Cette équation est très importante, elle possède un seul paramètre d'entrée qui a une influence minime sur le taux de compression. Les poids w_i augmentent de manière logarithmique avec $S_0 \cdot S_1$. Les poids peuvent être initialisés de façon à favoriser un modèle sur lequel une meilleure connaissance à priori est disponible, ou, ils peuvent être simplement initialisés à 0. Ce dernier cas de figure permet d'avoir un apprentissage rapide.

4.3.3.6.1.1.2 Codage arithmétique

Une fois les prédictions disponibles, elles sont codées arithmétiquement. Ce type de codage, très utilisé en compression de données, remplace une suite de symboles par un nombre réel. Par exemple, le code arithmétique d'un vecteur x est la longueur de x plus un nombre contenu dans l'intervalle $[p_{<x}, p_{<x} + p(x)]$, avec $p_{<x}$ la probabilité qu'un vecteur tiré aléatoirement soit lexico-graphiquement inférieur à x . Donc, ce type de codage permet de représenter une séquence de symboles par un intervalle de nombre réels compris entre 0 et 1. A noter que toute valeur appartenant à cet intervalle représentera d'une manière exclusive la séquence à coder. Cet intervalle se resserre avec l'allongement de la séquence à coder et le nombre de bits nécessaires pour représenter cet intervalle s'accroît. Les symboles successifs d'une séquence réduisent cet intervalle suivant la probabilité d'apparition du symbole. Finalement, les données compressées sont représentées par l'une des bornes de l'intervalle final. L'algorithme du codage arithmétique est le suivant :

1. Soit x , une séquence de m symboles, $x = x_1 x_2 \dots x_m$ appartenant à $S = \{s_1, s_2, \dots, s_n\}$
2. Calculer la probabilité de chaque symbole de la source S , $p_{s_i} = \text{proba}(S = s_i)$
3. Associer à chaque symbole s_k l'intervalle semi-ouvert $[VI_{s_k}, VS_{s_k}[$, avec : $VI_{s_k} \leq p_{s_k} \leq VS_{s_k}$, VI_{s_k} et VS_{s_k} étant, respectivement, la valeur inférieure et supérieure des probabilités associées au symbole s_k

CHAPITRE 4

4. Initialiser la limite inférieure VI de l'intervalle de calcul à 0 et la limite supérieure VS à 1
5. Tant qu'il existe des symboles de la séquence non codés mettre :
 - $Largeur = VS - VI$
 - $VI = VI + Largeur * VI_{S_k}$
 - $VS = VI + Largeur * VS_{S_k}$
6. Quand tous les symboles de x sont codés, toutes les valeurs de $[VS, VI[$ représentent d'une manière unique cette séquence

Généralement, c'est la borne inférieure de cet intervalle qui sera choisie. Un autre choix très récurrent est de prendre la moyenne des deux bornes. Ce codage génère soit une table de probabilité ou une table de fréquence de chaque symbole.

Lors de la décompression, le décodage de la séquence x se fait d'une manière très simple, en utilisant l'une des deux tables générées à la compression. Les étapes de décodage sont les suivantes :

1. Initialiser la limite inférieure VI et la limite supérieure, respectivement, à 0 et 1
2. Tant qu'il reste des symboles à décoder :
 - $Largeur = VS - VI$
 - Chercher le semi-intervalle $[VI_{S_k}, VS_{S_k})$ du symbole à décoder tel que :
$$VI_{S_k} \leq (code(x) - VI) / proba(s_k) \leq VS_{S_k}$$
 - Décoder le symbole s_k
 - $VI = VI + Largeur * VI_{S_k}$
 - $VS = VI + Largeur * VS_{S_k}$
3. Le résultat final représente la séquence x

Afin de mieux comprendre le mécanisme du codage arithmétique, nous proposons un exemple illustratif.

Soit la séquence « DIAGNOSTIC » à coder. D'abord nous déterminons les fréquences d'apparition de chaque symbole. Ces fréquences et les intervalles correspondants sont présentés dans le tableau (4-1) :

Symbole	Fréquence	Probabilité	Intervalles
D	1	0.1	$0 \leq I_1 \leq 0.1$
I	2	0.2	$0.1 \leq I_2 \leq 0.3$
A	1	0.1	$0.3 \leq I_3 \leq 0.4$
G	1	0.1	$0.4 \leq I_4 \leq 0.5$
N	1	0.1	$0.5 \leq I_5 \leq 0.6$
O	1	0.1	$0.6 \leq I_6 \leq 0.7$
S	1	0.1	$0.7 \leq I_7 \leq 0.8$
T	1	0.1	$0.8 \leq I_8 \leq 0.9$
C	1	0.1	$0.9 \leq I_9 \leq 1.0$

Tableau 4-1 Détermination des intervalles

Une fois les intervalles correspondants à chaque symbole définis, nous procédons au calcul de la valeur de l'estimation selon les équations données précédemment. Le tableau suivant résume cette étape de calcul.

Symbole	Borne inférieure	Borne supérieure
	0	1
D	0	0.1
I	0.01	0.03
A	0.016	0.018
G	0.0168	0.017
N	0.01690	0.01692
O	0.016912	0.016914
S	0.0169134	0.0169136
T	0.01691354	0.01691358
I	0,016913562	0,016913566
C	0,0169135656	0,016913566

Tableau 4-2 Calcul de la valeur du codage

CHAPITRE 4

Ainsi, la valeur représentant le codage arithmétique du message « DIAGNOSTIC » est 0,0169135656. Dans ce cas nous avons pris la borne inférieure de l'intervalle comme codage. Si nous considérons la moyenne de l'intervalle, nous obtenons 0,0169135658.

Pour décoder cette séquence, nous utilisons l'algorithme de décompression. A chaque fois, nous chercherons à quel intervalle appartient le rapport $(code(x) - VI)/proba(s_k)$. Le tableau suivant résume cette reconstitution : à noter que nous avons choisi la borne inférieure comme code de la séquence.

$(code(x) - VI) / Proba(s_k)$	Intervalle correspondant	Symbole correspondant
0,0169135656 (valeur initiale)	$0 \leq I_1 \leq 0.1$	D
0,169135656	$0.1 \leq I_2 \leq 0.3$	I
0,34567828	$0.3 \leq I_3 \leq 0.4$	A
0,4567828	$0.4 \leq I_4 \leq 0.5$	G
0,567828	$0.5 \leq I_5 \leq 0.6$	N
0,67828	$0.6 \leq I_6 \leq 0.7$	O
0,7828	$0.7 \leq I_7 \leq 0.8$	S
0,828	$0.8 \leq I_8 \leq 0.9$	T
0,14	$0.1 \leq I_2 \leq 0.3$	I
	$0.9 \leq I_9 \leq 1.0$	C

Tableau 4-3 Reconstitution

A préciser qu'il existe une version dynamique de ce codage appelée codage arithmétique adaptatif.

Une fois la compression faite, il est possible de calculer la distance séparant deux fichiers en utilisant les distances de compression. Cette métrique a été introduite par Bennet (Bennett, et al., 1998), elle est appelée également la distance informationnelle parce qu'elle compare le contenu informationnel des deux fichiers.

4.3.4 La Distance de Compression

Du point de vue traitement de données ou extraction de connaissances, la distance est un outil très performant pour la décision ou pour la classification de nouvelles données. Classiquement, une mesure de distance, ne porte pas sur l'information elle-même, mais sur un

mode de représentation structuré dans une syntaxe de codage de l'information. Cette syntaxe de codage est une quantification de la forme et résulte d'une segmentation de cette dernière dans un contenu. Si nous souhaitons étendre la notion de distance à une mesure entre information, il est nécessaire de pouvoir :

- Faire abstraction du format des données pour que la même mesure de distance soit applicable sur n'importe quel mode de présentation.
- Rechercher la plus petite unité commune de codage existante entre tous les modes de présentation de l'information.

L'évaluation de la distance de compression est réalisée en utilisant une fonction de coût minimale en bits engendrés par un programme p , qui permet de générer une séquence X tout en connaissant une autre séquence Y et vice-versa.

Dans ses travaux, Cilibrasi (Cilibrasi, et al., 2005) a démontré que cette fonction de coût représente une distance absolue entre le contenu informationnel de deux séquences X et Y . Cette distance possède toutes les caractéristiques mathématiques propres aux distances. L'équation suivante représente cette fonction de coût :

$$\begin{aligned} F_{c_u}(x, y) &= \min\{|p| : u(x, p) = y, u(y, p) = x\} \Leftrightarrow F_c(x, y) \\ &= \max(K(x|y), K(y|x)) \quad (4.12) \end{aligned}$$

La distance normalisée est obtenue par :

$$d(x, y) = \frac{F_c(x, y)}{\max(K(x|y), K(y|x))} \quad (4.13)$$

Avec F_c : la fonction de coût.

Comme la complexité de Kolmogorov est réellement incalculable, Vitanyi et Cilibrasi ont proposé d'utiliser la sortie d'un compresseur sans perte pour approximer la valeur de cette dernière. Avec cette approximation, la distance de compression normalisée (NCD) ou de Bennet devient :

$$d(x, y) = \frac{\mathit{Comp}(xy) - \min\{\mathit{Comp}(x), \mathit{Comp}(y)\}}{\max\{\mathit{Comp}(x), \mathit{Comp}(y)\}} \quad (4.14)$$

Comp est la fonction de compression sans perte qui retourne une longueur exprimée en bits. x et y sont les séquences porteuses d'informations dont nous voulons calculer la distance et xy est la concaténation des deux séquences. Selon Cilibrasi, cette distance peut être considérée comme une mesure de la divergence de Kullback-Leibler de la moyenne des distributions X et

Y. En pratique, l'application de cette distance est indépendante du format, de la mise en forme et de la modalité de présentation de l'information. Il s'agit d'une distance universelle. Il est important de souligner que l'implémentation du calcul de cette distance ne nécessite pas de ressources informatiques importantes. Il s'agit d'une opération mathématique effectuée sur des données compressées. La plus grande partie des ressources est consacrée à la compression. Cette facilité d'utilisation nous permet d'exploiter cette distance en détection et isolation de défauts.

4.4 Applications

4.4.1 Application 1 : Capacité discriminatoire de l'approche

Afin de tester cette technique et d'évaluer l'apport de la complexité de Kolmogorov sur la problématique FDI, nous allons appliquer cette méthode sur une autre application automobile. Dans cette deuxième application, nous disposons de fichiers représentant l'historique des défauts enregistrés dans les différents calculateurs embarqués sur un véhicule. Ces fichiers, appelés logfiles, ont été générés par PSA Peugeot Citroën dans le cadre du projet DIAPA (Diagnostic Automobile Par Apprentissage). Les partenaires du projet sont : DELPHI, PSA Peugeot Citroën, Freescale Semiconductor, UTC Compiègne et Serma Ingénierie. Ce projet ANR a pour objectif de proposer des procédures de diagnostic des véhicules par apprentissage des logfiles associés à des défauts répertoriés.

Trois différents défauts ont été artificiellement créés sur un véhicule qui a ensuite été placé dans différentes conditions de fonctionnement, afin de générer 30 logfiles différents pour chaque panne. Ces logfiles contiennent les codes des défauts détectés par l'ensemble des calculateurs du véhicule au cours du test, mais également les valeurs de quelques grandeurs analogiques au moment du défaut. Pour des raisons de confidentialité des données, il ne nous est pas possible de décrire plus en détail le contenu des logfiles. Il est important de noter que la nature des défauts créés correspondant aux pannes 2 et 3 conduisent à des logfiles sont parfois très proches.

Sur cette application, nous allons utiliser la complexité de Kolmogorov, en particulier la distance de compression dans le but de discriminer les trois différents défauts. Afin de réaliser cette tâche, nous allons d'abord définir un prototype de chacune des trois pannes.

4.4.1.1 Définition d'un prototype

Il existe plusieurs méthodes et techniques pour déterminer un prototype. En classification, un prototype représente un ensemble d'individus (points, caractères ... etc.) appartenant à une population possédant les mêmes caractéristiques.

Afin de déterminer un prototype de chaque panne générée dans cette application, nous avons choisi aléatoirement 20 logfiles comme ensemble d'apprentissage, parmi les 30 logfiles dont nous disposons pour cette application. Pour la validation, nous avons utilisé la totalité des logfiles.

Pour chacune des pannes, nous avons choisi comme prototype le logfile possédant la complexité minimale. Ce choix nous permet de dégager trois logfiles types représentant efficacement les trois pannes étudiées. Nous avons ensuite calculé la distance de compression normalisée entre les 90 logfiles à notre disposition et les 3 prototypes. Ces résultats sont considérés panne par panne sur les figures Fig. 4.4 à Fig. 4.6. Sur ces figures les '+' bleus représentent les distances entre les logfiles des pannes 1 à 3 respectivement, et le prototype de la panne 1, les '+' verts représentent les distances entre les logfiles des pannes 1 à 3 respectivement, et le prototype de la panne 2. Les '+' rouges représentent les distances entre les logfiles des pannes 1 à 3 respectivement, et le prototype de la panne 3.

4.4.1.2 Résultats

4.4.1.2.1 Panne 1 :

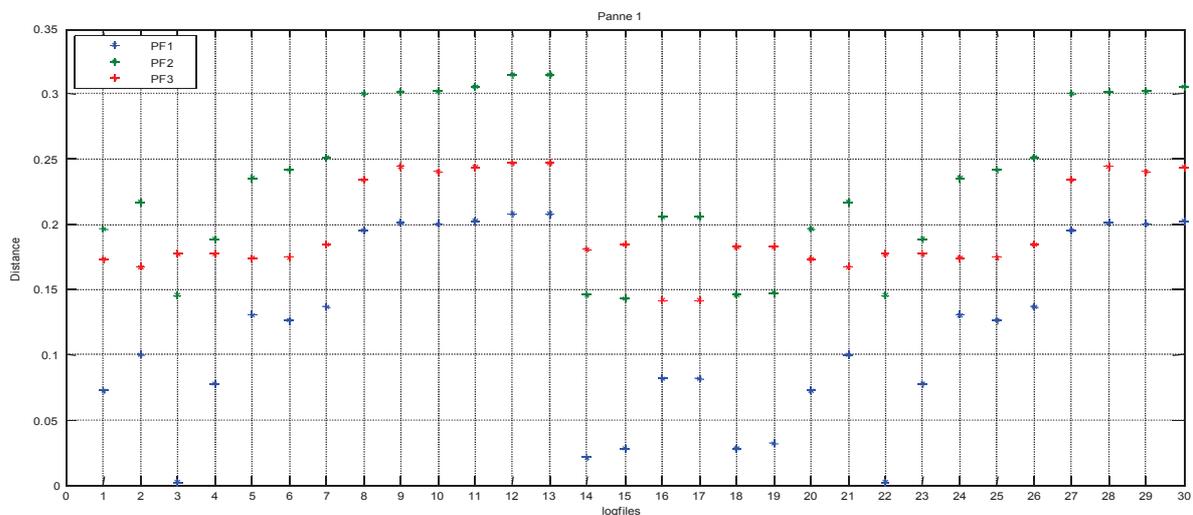


Fig. 4.4 Distance des logfiles de la panne 1 aux prototypes des trois pannes

CHAPITRE 4

4.4.1.2.2 Panne 2:

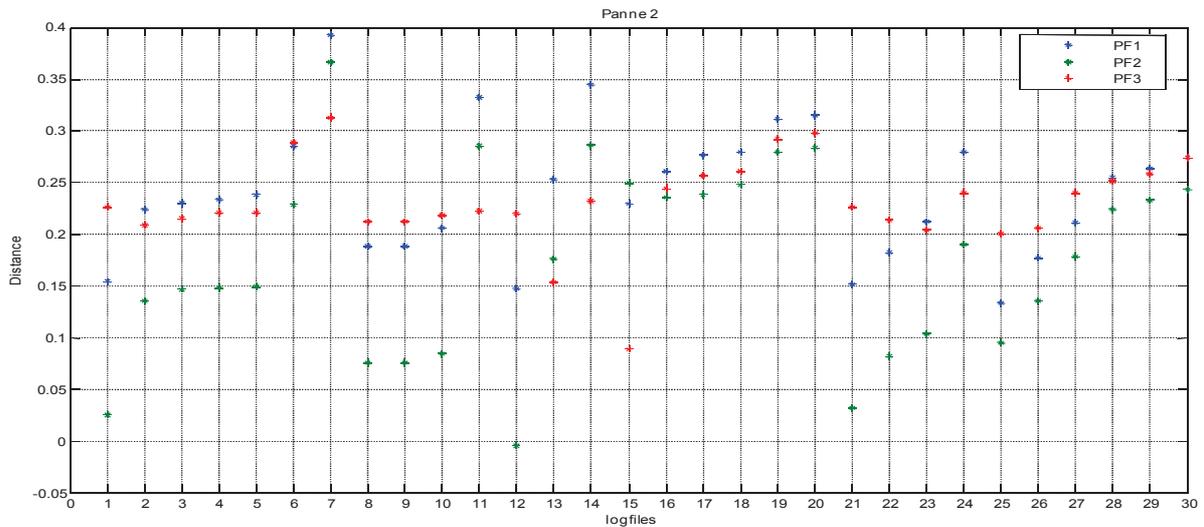


Fig. 4.5 Distance des logfiles de la panne 2 aux prototypes des trois pannes

4.4.1.2.3 Panne 3

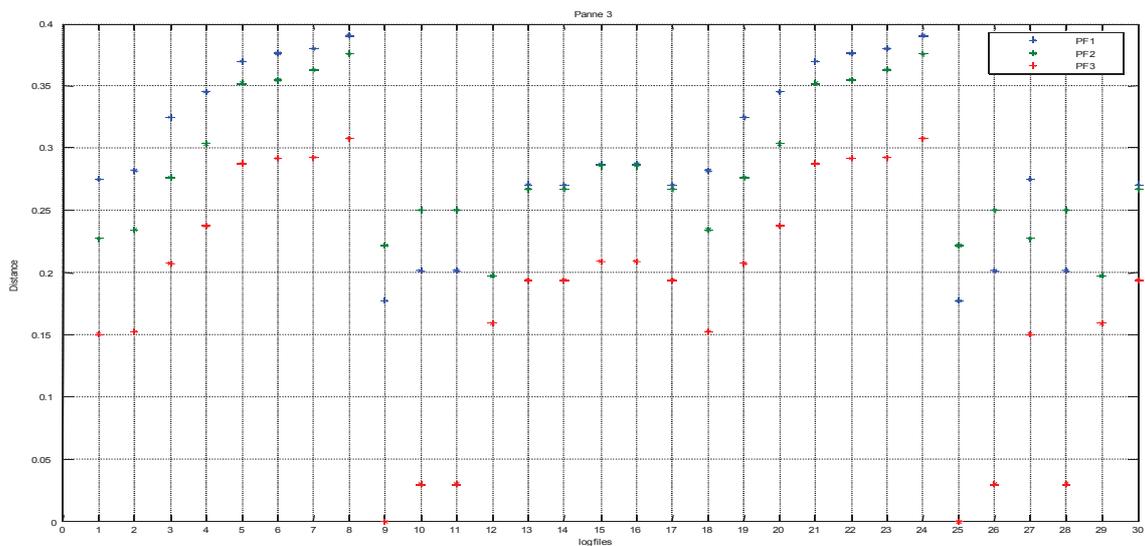


Fig. 4.6 Distance des logfiles de la panne 3 aux prototypes des trois pannes

Nous constatons que les distances entre les logfiles relatifs à la panne 1 et le prototype de cette panne sont inférieures aux distances aux prototypes des pannes 2 et 3. Nous avons obtenu des résultats semblables pour les logfiles des autres pannes, mais sur la panne 2, il existe 5 logfiles (numéro 7, 11, 13, 14 et 15) qui sont plus proches de la panne 3. Ces exceptions sont dus à la ressemblance significative des deux pannes (2 et 3).

Sur cette application, nous avons démontré l'intérêt de la complexité de Kolmogorov pour la discrimination de pannes en automobile. Plus précisément, nous avons mis en

évidence l'apport de la compression de données et, en particulier, la distance par compression normalisée à la discrimination de différentes pannes qui peuvent apparaître sur un système donné (un véhicule dans notre cas).

4.4.1.3 Conclusion

Dans cette application, nous avons pu vérifier la capacité discriminatoire de notre approche, mais comme nous ne disposons pas de données de bon fonctionnement de ce véhicule, nous n'avons pas pu tester la capacité de détection. Afin de vérifier cette propriété nous disposons de données correspond aux échanges entre un outil de diagnostic et les calculateurs d'un véhicule en bon état de fonctionnement et après génération de différentes pannes en déconnectant certains capteurs.

4.4.2 Application 2 : Capacité de détection de l'approche

La propriété de détection est une des propriétés essentielles d'un système FDI. Elle incarne le premier contact avec le défaut ou la panne. Le traitement de la panne est réalisé avec succès et sans dégradation du système si la tâche de détection intervient au moment opportun. La détection doit remarquer un comportement inhabituel sur le fonctionnement du système. Dans cette 3^{ième} application, également issue du projet DIAPA, nous allons exploiter les messages échangés, puis enregistrés sous forme de logfiles, entre l'outil de diagnostic et les différents calculateurs embarqués sur le véhicule. A noter que ces fichiers ont été enregistrés par (Delphi-France), et que pour des raisons de confidentialité nous ne détaillons pas ces différentes pannes, que nous contentons d'appeler panne 1, panne 2 et panne 3, ni le contenu des logfiles.

Dans cette application, nous allons effectuer le calcul des distances de compression normalisées entre les différents fichiers enregistrés pour chaque panne. Une même distance sera calculée entre ces fichiers de panne et le fichier enregistré après avoir reconnecté les différents capteurs pour les différentes expérimentations. Cette démarche va nous permettre de mettre en évidence la faculté de notre approche à détecter l'apparition d'une panne sur le véhicule. Afin de mieux présenter les résultats obtenus, nous utilisons les notations suivantes :

- Pour la panne i :
 - File i 1 : Fichier enregistré après avoir déconnecter le capteur correspondant à la panne 1.

CHAPITRE 4

- Filei2 : Fichier enregistré après un laps de temps, le capteur étant toujours déconnecté.
- Filei3 : Fichier enregistré après un deuxième laps de temps.
- Filei4 : Fichier enregistré après un troisième laps de temps.
- Filei5 : Fichier enregistré après avoir reconnecté le capteur.

Les distances sont calculées entre chaque fichier d'une panne donnée avec les autres fichiers de la même de la même panne. Les résultats sont présentés sur les figures suivantes :

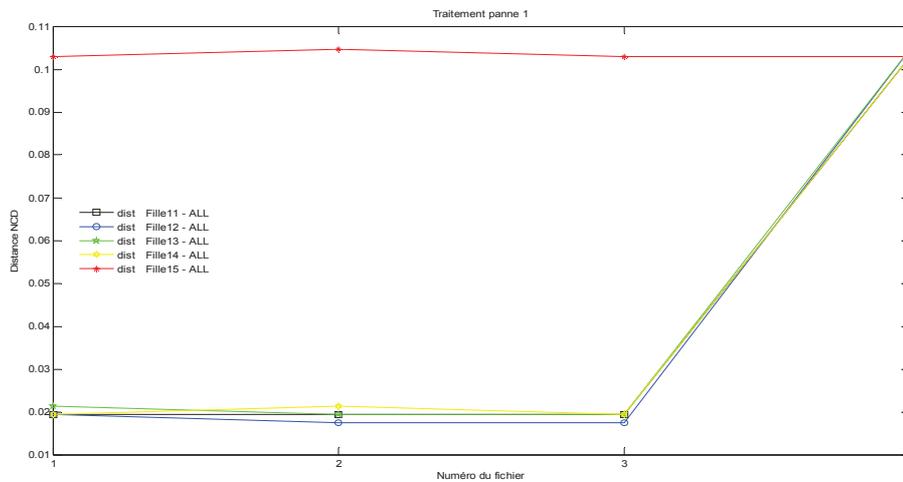


Fig. 4.7 Distance NCD pour la panne 1

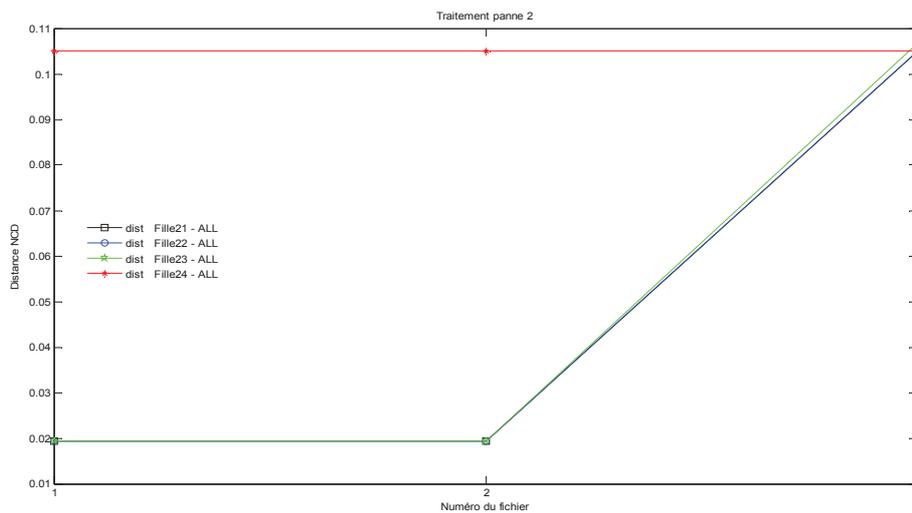


Fig. 4.8 Distance NCD pour la panne 2

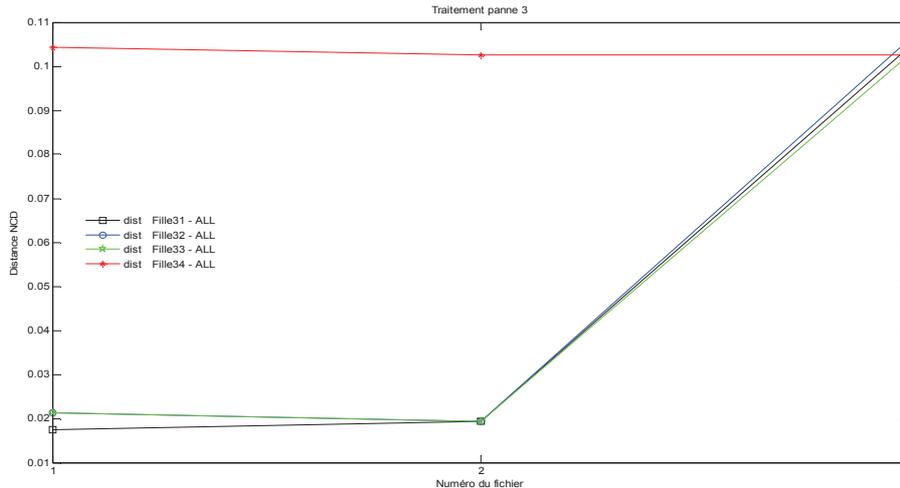


Fig. 4.9 Distance NCD pour la panne 3

4.4.2.1 Interprétation

Au vue des résultats, nous pouvons dire, d'abord, que des observations identiques sont relevées pour les trois expérimentations. Nous remarquons qu'en général les distances sont plutôt faibles, ce qui peut s'expliquer par le volume important de l'entête de chaque fichier, entête qui détaille la marque, le type du véhicule et la date de l'expérience. Ce fichier d'entête est commun à tous les fichiers, ce qui augmente la redondance entre les différents fichiers, et par conséquent, réduit la distance entre ces fichiers. Au-delà de la faiblesse de ces distances, nous pouvons observer deux groupes de fichiers (observations valable pour les trois pannes) ; le premier groupe constitué de fichiers possédant des distances extrêmement faibles, et le deuxième groupe regroupant des fichiers possédant des distances suffisamment importantes en comparaison au premier groupe.

Pour la panne 1, nous remarquons que les distances entre File11 et, File12, File13 et File14 sont très faibles comparativement à celles File11 et File15. Le seul fichier qui est à une distance importante des autres fichiers est le File15, ce fichier correspond aux enregistrements réalisés après avoir reconnecter le capteur. Cela nous renseigne sur la réelle capacité de la technique que nous proposons à détecter efficacement la présence d'une panne. En effet, la distance effective calculée entre le fichier représentant le bon fonctionnement et ceux représentant le mauvais fonctionnement est si importante qu'elle permet de conclure sur l'existence d'une anomalie dans le système (véhicule).

Concernant les pannes 2 et 3, c'est la même conclusion que nous pouvons tirer, sauf que, pour ces deux pannes, nous possédons trois fichiers contenant des données du mauvais

CHAPITRE 4

fonctionnement et un seul fichier de données du bon fonctionnement. Ces résultats permettent de mettre en évidence que la complexité de Kolmogorov permet de mettre en évidence la présence d'un défaut avec une efficacité remarquable sans se préoccuper du contenu des données à analyser.

4.5 Conclusion

Dans ce chapitre, nous avons traité une nouvelle technique de reconnaissance des formes qui ne fait pas appel à un traitement de données. A première vue, cette approche présente l'avantage majeur de ne pas exiger des ressources calculatoires importantes, contrairement à toutes les méthodes de reconnaissance de formes. Avec cette approche, il suffit de trouver un algorithme de compression performant.

Dans ce chapitre, nous avons proposé d'utiliser la complexité de Kolmogorov dans un processus de détection et d'isolation de défauts. Nous avons mis l'accent sur l'impossibilité de calculer, en pratique, cette grandeur mathématique. Pour cela plusieurs travaux sont allés dans ce sens en proposant des alternatives qui permettent d'estimer cette grandeur importante. Parmi ces travaux, nous nous sommes intéressés aux travaux de Cilibrasi (Cilibrasi, et al., 2005) qui proposent d'estimer la complexité du Kolmogorov par un algorithme de compression.

Une fois cette grandeur estimée, nous avons utilisé la distance de compression normalisée, proposée par Cilibrasi, dans deux applications automobiles différentes. La première application a pour objectif de discriminer plusieurs pannes générées sur un véhicule. A partir de trois différents jeux de logfiles correspondant à trois pannes, nous avons déterminé un prototype pour chaque panne. Ce prototype correspond au logfile qui possède la plus petite complexité de Kolmogorov parmi les 20 logfiles choisis aléatoirement dans la base de données de chacune des pannes. A signaler qu'il existe plusieurs méthodes pour déterminer un prototype. Avec la distance de compression normalisée nous avons pu discriminer les trois différentes pannes considérées dans cette application, et ainsi démontré la capacité discriminative de la complexité de Kolmogorov, et en particulier, la distance de compression normalisée.

Afin de démontrer la capacité de détection de notre approche, nous avons traité une autre application automobile dans laquelle nous avons à notre disposition plusieurs fichiers générés par un outil de diagnostic sur un véhicule après déconnexion d'un capteur pour

générer une panne ou après avoir reconnecter ce capteur pour éliminer la panne. Trois différentes pannes ont été générées. Après avoir calculé les distances de compression entre les différents fichiers correspondant aux pannes, nous avons remarqué que ces distances sont importantes quand il s'agit de comparer un fichier avec capteur déconnecté et un fichier avec capteur reconnecté. Par contre, quand il s'agit de deux fichiers de même nature, les distances sont très faibles. Ceci nous renseigne sur le pouvoir de la distance de compression à détecter l'apparition des pannes dans un véhicule.

L'approche que nous venons de proposer présente plusieurs avantages, notamment, ceux liés au très faible besoin en ressource calculatoire et la simplicité de la mise en œuvre de la méthode. Tous ces avantages nous permettent d'envisager une application en temps réel, dans laquelle, les défauts sont détectés et isolés au cours du fonctionnement du véhicule.

4.6 Références

Bennett C. H [et al.] Information Distance [Revue] // IEEE Transactions on Information Theory. - 1998. - Vol. 44 (4). - pp. 1407–1423.

Cilibrasi R [et al.] On the complexity of several haplotyping problems, Algorithms in Bioinformatics [Conférence] // 5th International Workshop, WABI, Proceedings (Lecture Notes in Computer Science / Lecture Notes in Bioinformatics). - Mallorca, Spain : [s.n.], 2005. - pp. 128-139.

Cilibrasi R et Vitanyi P. M. B Clustering by Compression [Revue] // IEEE Transactions on Information Theory. - APRIL 2005. - 4 : Vol. 51. - pp. 1523–1545.

Delphi-France [En ligne]. - <http://www.delphi.com>.

Kabbaj N, Doncescu A et Benkaci M Complexity of Kolmogorov as Method of Diagnosis [Conférence] // International Workshop on Diagnosis of Complex Systems (DCS-2010), FIFTH International Conference on Broadband and Wireless Computing, Communication and Applications. - Fukuoka, Japan : [s.n.], 2010.

Keogh E [et al.] Compression-based data mining of sequential data [Revue] // Data Mining and Knowledge Discovery. - 2007. - Vol. 14 (1). - pp. 99-129.

Kolmogorov A. N Three approaches to the quantitative definition of information [Revue] // Problems of Information Transmission. - 1965. - Vol. 1. - pp. 4-7.

Levorato V Contributions à la Modélisation des Réseaux Complexes : Prétopologie et Applications [Rapport] : Thèse doctorat / Université Paris 8. - Paris : [s.n.], 2008.

Pigeon S Contributions à la compression de données [Rapport]. - Montréal : Université de Montréal, 2001.

CHAPITRE 4

Solomonoff R A formal theory of inductive inference, part I [Revue] // Information and control. - 1964 a. - Vol. 7. - pp. 1–22.

Solomonoff R A formal theory of inductive inference, part II [Revue] // Information and control. - 1964 b. - Vol. 7. - pp. 224–254.

Turing A. M On computable numbers, with an applications to the Entscheidungsproblem [Revue] // London Mathematical Society . - London : [s.n.], 1937. - Vol. 42. - pp. 230-265.

Conclusion générale

Dans cette thèse nous nous sommes focalisés sur la problématique de localisation des défaillances, avec une application dans le domaine du service après-vente automobile. L'introduction massive d'électronique dans les véhicules au cours de ces 20 dernières années a permis de réaliser des progrès considérables en terme de performances, de sécurité et de confort, mais l'augmentation du nombre de calculateurs embarqués a rendu la recherche de pannes de plus en plus difficile. L'objectif du travail présenté dans ce document était donc de développer un système de détection et surtout d'isolation de fautes à la fois innovant et efficace. Pour atteindre cet objectif, nous avons proposé deux approches :

- Une basée sur une modélisation par hyper-rectangles de la zone de l'espace des paramètres associée au bon fonctionnement du véhicule, qui utilise une notion de conflit que nous avons proposé entre le modèle de bon fonctionnement et les acquisitions effectuées sur le véhicule présentant un défaut.
- La deuxième se base sur la complexité de Kolmogorov et la distance de compression normalisée entre la base de données associée au bon fonctionnement ou à des pannes bien identifiées et les enregistrements effectués sur le véhicule présentant un défaut. L'un des intérêts de cette dernière approche est de pouvoir traiter des informations de natures diverses (valeurs numériques, code défauts, symptômes clients, ...etc.).

Pour mettre en œuvre la méthode de localisation basée sur des modèles par hyper-rectangles, nous avons utilisé l'algorithme fuzzy-ARTMAP qui réalise un découpage optimal en hyper-rectangles de la zone de l'espace des variables occupée par les mesures. Une fois le modèle déterminé, la détection, pas essentielle dans le cas de l'après-vente automobile, et la localisation sont réalisées en se basant sur la notion de conflit que nous avons proposée. En considérant que la présence d'une panne doit systématiquement générer un conflit entre les mesures et le modèle de bon fonctionnement, l'existence de ce conflit va nous permettre, si nécessaire de faire la détection, tandis que sa valeur va nous permettre de localiser le défaut en sélectionnant les variables corrélées à ce dernier. Cette approche a été appliquée, avec succès, sur des données réelles issues d'une campagne d'essais mené chez ACTIA pour le développement d'un outil de diagnostic par reconnaissance de forme. La principale difficulté de mise en œuvre de ce type d'approche est liée à la complexité algorithmique. Une

CONCLUSION GENERALE

amélioration de la technique d'apprentissage, tout en gardant la même forme de classes, fait donc partie des prospectives de ce travail.

Dans ce manuscrit, nous avons proposé une autre méthode de détection et localisation de fautes qui permet de traiter des données de nature hétérogène et possède l'immense avantage de nécessiter peu de ressources calculatoires et de présenter une complexité algorithmique réduite. Cette technique se base sur la complexité de Kolmogorov, qui est estimée en utilisant la technique de compression de données. Nous considérons que la taille d'un fichier compressé par un algorithme de compression performant est la borne supérieure de la complexité de Kolmogorov. La similitude entre fichiers est alors quantifiable par une distance de compression normalisée. Nous avons appliqué cette méthode à des fichiers de données réels, générés dans le cadre du projet ANR DIAPA et montré la capacité discriminative et le pouvoir de détection de l'outil développé. La capacité de discrimination a été validée sur l'une des bases de données proposées dans DIAPA qui contenait les codes défauts et les variables locales mémorisés dans l'ensemble des calculateurs du véhicule. Suivant le mode opératoire du véhicule, un même défaut génère des enregistrements différents, ou, au contraire, des défauts différents peuvent générer des enregistrements très proches.

Le pouvoir de détection a été validé sur un autre lot de fichiers contenant des données récupérées sur les calculateurs d'un véhicule, soit en bon état de fonctionnement, soit après génération d'un défaut. Dans cette dernière application, nous avons constaté que les distances entre les données en mode nominal et celles enregistrées en mode défaillant sont plus importantes que les distances entre les fichiers d'un même état de fonctionnement. Il est important de noter que la faiblesse relative de ces distances est liée principalement à une certaine redondance d'informations présentes dans les fichiers : type, année et équipements du véhicule.

Dans l'approche FDI se basant sur la complexité de Kolmogorov, la compression de données joue un rôle fondamental. En effet, pour que le système soit performant, nous devons obtenir la meilleure approximation possible de la complexité de Kolmogorov. Pour cela et, au delà des résultats obtenus par les algorithmes de compression actuels, il nous semble nécessaire de proposer des algorithmes de compression dédiés à la détection et l'isolation de pannes.

Contributions de la thèse

La notion de conflit ou désaccord entre acquisitions et le modèle de bon fonctionnement à base d'hyper-rectangles que nous avons proposés nous parait fondamentale pour une tâche de détection et d'isolation de panne. Cette notion tire sa force du fait que toute anomalie, dans n'importe quel système industriel, doit provoquer une incohérence sur l'intersection informationnelle des variables vis-à-vis de son comportement nominal. Donc, l'exploitation de cette notion de conflit est très bénéfique et permet de proposer des systèmes FDI performants.

La complexité algorithmique est une problématique récurrente dans plusieurs domaines, notamment en détection et isolation de pannes. L'exploitation de la complexité de Kolmogorov et l'utilisation de la distance de compression normalisée dans un système FDI représente une alternative importante et efficace aux méthodes déjà existantes. Cette approche nécessite peu de ressources calculatoires et possède une complexité algorithmique très réduite, liée principalement à la tâche de compression. Cet avantage doit nous permettre d'envisager une mise en œuvre en temps réel de ce type d'outil dans les systèmes embarqués, et en particulier dans l'automobile.

Title: Monitoring of Mechatronics automotive systems using learning Methods

Abstract:

Mechatronic systems monitoring, especially those built on today's vehicles, is increasingly complicated. The interconnections of these systems for increased performance and comfort of vehicles increases the complexity of information needed for decision-making in real time.

This PhD thesis is devoted to the problem of detection and isolation (FDI Fault Detection & Isolation) of faults in automotive systems using algorithms based on research and evaluation of information by mono-criterion approaches. Relevant variables for rapid detection of faults are selected in an automatic manner by using two different approaches:

- I. The first is to introduce the notion of conflict between all the measurable variables of mechatronic system and to analyze these variables using their projections in hyper-rectangles spaces classification.
- II. The second approach is to use Kolmogorov complexity as a tool for classification of fault signatures. The estimate of the Kolmogorov complexity by compression algorithms, without loss of information, allows defining a dictionary of faults and giving a score of criticality with respect to the healthy functioning of the vehicle.

The two proposed approaches have been successfully applied to many types of automotive data in the ANR-DIAP project.

Key words: Diagnosis, Fault detection and isolation, Fuzzy-ARTMAP, Relevance, Conflict, Classification and pattern recognition, Clustering, Automotive industry.

AUTEUR : Mourad BENKACI

TITRE : Surveillance des systèmes mécatronique d'automobile par des méthodes d'apprentissage

DIRECTEURS DE THESE : Andrei DONCESCU et Bruno JAMMES

LIEU ET DATE DE SOUTENANCE : LAAS-CNRS 24 Février 2011

RESUME : La surveillance des systèmes mécatroniques, en particulier, ceux intégrés sur les véhicules d'aujourd'hui est de plus en plus difficile. Les interconnexions de ces systèmes en vue de l'accroissement des performances et du confort de véhicule augmentent la complexité de l'information nécessaire à la prise de décision en temps réel.

Cette thèse est consacrée à la problématique de détection et d'isolation (FDI, Fault Detection & Isolation) de pannes automobiles en utilisant des systèmes de recherche et d'évaluation de l'information par des approches monocritères. Les variables pertinentes pour la détection rapide des pannes sont sélectionnées d'une manière automatique en utilisant deux approches différentes :

- I. La première consiste à introduire la notion de conflit entre toutes les variables mesurables du système mécatronique et les analyser à partir des projections dans des espaces de classification hyper-rectangles.
- II. La deuxième approche consiste à utiliser la complexité de Kolmogorov comme outil de classification des signatures de pannes. L'estimation de la complexité de Kolmogorov par des algorithmes de compression sans perte d'information permet de définir un dictionnaire de pannes et de donner un score de criticité par rapport au bon fonctionnement du véhicule.

Les deux approches proposées ont été appliquées avec succès sur plusieurs types de données automobiles dans le cadre du projet ANR-DIAPA

Mots-clés : Diagnostic, Détection et isolation de pannes, Fuzzy-ARTMAP, Pertinence de variables, Conflit, Classification et reconnaissance de formes, Clustering, Industrie automobile.

DISCIPLINE ADMINISTRATIVE : SYSTEMES EMBARQUES

INTITULE ET ADRESSE DE L'U.F.R. OU DU LABORATOIRE : LAAS-CNRS ; 7, avenue du colonel Roche 31077 Toulouse