

وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR- ANNABA UNIVERSITY

UNIVERSITE BADJI MOKHTAR ANNABA



جامعة باجي مختار - عناية

Faculté: Sciences de l'ingénieur  
Département: Electronique

Année : 2011

## MEMOIRE

Présenté En Vue De L'obtention Du Diplôme De Magister

Intitulé

**MODELISATION ET DIAGNOSTIQUE DES  
SYSTEMES NON-LINEAIRES : Une approche par  
Analyse en Composantes Principales et Carte  
Topologique de Kohonen**

Option

Commande Avancée

Par

FADDA Med Lamine

Directeur de mémoire : Mr L. BENNACER

Dr

U. Annaba

DEVANT LE JURY

Président :

Mr S. YAHMEDI

Pr

U. Annaba

Examineurs :

Mr M.T. KHADIR

Pr

U. Annaba

Mr M.F. HARKAT

Dr

U. Annaba

Mr M. BOUGHAZI

Dr

U. Annaba

## **Remerciements**

Je tiens à exprimer ma profonde reconnaissance au directeur de la thèse M<sup>r</sup> Docteur L.BENNACER, pour son suivi, ses nombreux conseils et ses critiques constructives pour l'élaboration de ce travail.

Mon vif remerciement à Monsieur le professeur M<sup>r</sup> S.YAHMEDI pour avoir aimablement accepté de présider le jury de ce mémoire de Magister.

Mes chaleureux remerciements et toute ma gratitude aux M<sup>r</sup> P<sup>r</sup>.M.T. KHADIR, M<sup>r</sup> D<sup>r</sup>.M.F.HARKAT et M<sup>r</sup> D<sup>r</sup>.M.BOUGHAZI pour avoir accepté d'être examinateurs de ce travail et pour leur encouragement.

Je remercie mes collègues de travail pour leur aide et encouragements.

Mes remerciements à tous ceux qui ont contribué de près ou de loin pour l'accomplissement de ce travail.

## **Dédicace**

Je voudrais dédier cet humble travail à ma  
chère maman et mon cher père.

A ma sœur Moufida.

A mes frères Faiçal et Chams Eddine.

Et à tous mes amis.

## Résumé :

Pour étudier, résumer, représenter des données multidimensionnelles comprenant à la fois des variables quantitatives (à valeurs continues réelles) et qualitatives (discrètes, ordinales ou nominales), nous avons mis à notre disposition deux méthodes de traitement de ces données qui sont : l'Analyse en Composantes Principales (ACP), et une méthode neuronale basée sur l'utilisation des cartes de Kohonen. Ces deux techniques sont destinées généralement pour la classification des données. Dans notre cas, nous les avons utilisés pour faire de la détection et de la localisation des défauts. Nous avons démontré dans la suite de notre travail, comment on peut se servir de ces deux techniques pour représenter les données multidimensionnelles, détecter et localiser les défauts dans ces données.

Les résultats de test obtenus montrent l'efficacité du choix adopté. L'utilisation de la démarche ainsi retenue, constitue une voie prometteuse pour les systèmes d'aide au diagnostic.

## المخلص :

لدراسة ، تلخيص وتمثيل البيانات المتعددة الأبعاد بما في ذلك المتغيرات الكمية ( ذات القيم الحقيقية المستمرة) وكذلك النوعية (المنفصلة، أو الترتيبية الاسمية ) ، قمنا بالاعتماد على طريقتين لمعالجة هذه البيانات و هما : تحليل العناصر الرئيسية (ACP) ، وأسلوب مرتكز على استخدام الخرائط العصبية لـ Kohonen . عموماً ، يكون استغلال هاتين التقنيتين في تصنيف البيانات. ولكن في حالتنا، لقد تم استخدام هاتين التقنيتين في كشف وتحديد مواقع الخلل، حيث أظهرنا في أعمالنا، كيفية استخدام هاتين التقنيتين في تمثيل البيانات المتعددة الأبعاد، وكذلك في كشف وتحديد مواقع الخلل في هذه البيانات.

النتائج التي حصل عليها أظهرت فعالية من خلال الخيار المعتمد . استخدام النهج المعتمد وبالتالي، وسيلة واعدة لدعم نظم التشخيص.

## Abstract:

To investigate, summarize, represent multidimensional data including both quantitative variables (continuous real-valued) and qualitative (discrete, ordinal or nominal), we have at our disposal two methods for processing such data are: the Principal Component Analysis (PCA), and a method based on using self organized maps Kohonen (SOM).

These two techniques are generally designed for data classification. In our case, we have used in faults detection and isolation. We demonstrated in our work, the employment of these two techniques to represent multidimensional data, detect and locate of faults. the obtained results show the effectiveness of the adopted choice. The use of the approach thus adopted, is a promising avenue for support systems for diagnosis.

# Table des Matières

<b>INTRODUCTION GENERALE</b> .....	1
<b>CHAPITRE I : DIAGNOSTIC - PRINCIPE ET METHODOLOGIES</b> .....	2
I.1 Principe .....	2
I.2 Détection de défauts et diagnostic .....	2
I.3 Classification des méthodes de diagnostic .....	5
I.3.1 Diagnostic Quantitatif .....	8
- Les méthodes basées sur l'approche mono-signal .....	8
- Redondance matérielle .....	8
- Approches statistiques .....	8
- L'approche multi-signaux .....	8
- Espace de parité .....	10
- Observateurs .....	11
- Estimation paramétrique .....	11
I.3.2 Diagnostic Qualitatif .....	11
- Modèles issus de l'intelligence artificielle (IA) .....	11
- Réseaux de neurones artificiels (RNA) .....	12
- Systèmes d'inférence flous (SIF) .....	12
- Reconnaissance de formes .....	13
I.4 Méthodes de classification .....	14
I.4.1 Algorithmes par hiérarchisation .....	15
I.4.2 Algorithmes par partitionnement .....	16
I.4.3 Algorithmes issus de la théorie des graphes .....	17
I.4.4 Algorithme de résolution mixte « Mixture-Resolving » .....	18
I.4.5 Classification par le plus proche voisin .....	18
I.4.6 Classification Floue .....	18
I.5 La méthode LAMDA .....	18
I.6 Conclusions .....	19
<b>CHAPITRE II : DETECTION ET LOCALISATION DES DEFAUTS PAR ACP</b> .....	20
II.1 Principe .....	20
II.2 Modélisation par ACP .....	20
II.3 Détection .....	23
II.3.1 : Première méthode : test sur les dernières composantes principales .....	23
II.3.2 Deuxième méthode : Test sur l'erreur de prédiction .....	25
II.3.3 Conditions de détectabilité .....	26
II. 3 Localisation .....	27
II.3.1 Reconstruction des variables .....	28
II.3.2 Méthodes de localisation des défauts .....	31
II.3.2.1 Première méthode : test sur les dernières composantes .....	31
II.3.2.2 Deuxième méthode : test sur la quantité SPE. ....	32
II.3.2.3 La contribution classique au critère SPE .....	34
II.4 Conclusion .....	35

<b>CHAPITRE III : DETECTION ET LOCALISATION DES DEFAUTS PAR CARTE TOPOLOGIQUE DE KOHENEN (SOM)</b> .....	36
III.1. Principe .....	36
III.2.Phase d'apprentissage (Modélisation) .....	36
III.2.1.Principe .....	36
III.2.2.Modélisation par carte de Kohonen.....	36
III.2.3.Architecture.....	37
III.2.4.Apprentissage par la carte.....	37
III.3.Détection et Localisation des défauts .....	38
III.3.1.Principe .....	38
III.3.2.Décision.....	38
III.4.Reconstruction des données.....	40
III.5. Conclusion .....	41
<b>CHAPITRE IV : RESULTATS EXPERIMENTAUX</b> .....	42
IV.1. Les Données .....	42
IV.2. Modélisation .....	43
IV.2.1.Choix des dimensions de la carte .....	43
IV.3. Détection et localisation des défauts par la carte .....	45
IV.4 Reconstruction des données .....	47
IV.5. Cas de deux défauts simultanés (Multiple).....	48
- Exemple 2 .....	48
IV.5.1 Résultats de l'ACP .....	49
IV.5.2 Résultats de carte de Kohonen.....	52
IV.6.conclusion .....	54
V.CONCLUSION ET PERSPECTIVE .....	55
BIBLIIOGRAPHIE .....	56
ANNEXE .....	60

## Liste des figures

Figure I.1 Différents types de défauts : a) défaut abrupt, b) défaut intermittent, c) défaut graduel.....	3
Figure I.2 Schéma général d'un système de diagnostic.....	4
Figure I.3 Classification des méthodes de diagnostic.....	6
Figure I.4 Architecture générale de la détection de défauts à base de modèles.....	10
Figure I.5 Différentes approches de classification non supervisée.....	15
Figure I.6 Classes formées en utilisant le MST (« <i>Minimal Spanning Tree</i> »).....	17
Figure. II.1 Représentation d'observations et des parties modélisée $\hat{X}$ et résiduelle $\tilde{X}$ de l'une d'entre elles $X$ dans le cas de deux variables ( $m = 2$ ).....	22
Figure . II.2 Méthode de détection.....	24
Figure II.3 $In_1$ sur les données sans défaut (1 <sup>ère</sup> courbe) et (2 <sup>ème</sup> courbe) $In_1$ sur les données avec défaut.....	25
Figure II.4 SPE avec anomalie.....	26
Figure II.6 Reconstruction des variables sans $x_1$ .....	29
Figure II.7 Détermination de $f_i \xi_i$ .....	30
Figure. II.8 Carré de la dernière composante principale après reconstruction des variables.....	32
Figure II.9 Indice de validité des variables.....	33
Figure. II-10 SPE après reconstruction de chacune des variables.....	34
Figure III.1. Schéma d'une carte de Kohonen connectée à un stimulus. Les connections sont complètes.....	37
Figure IV.1 Signaux retenus.....	42
Figure IV.2 Résultats de la phase d'apprentissage.....	44
Figure IV.3.a Fonction d'activation des données.....	45
Figure IV.3.b Fonction d'activation des données (des défauts affectant $x_1$ à l'instant (10-30) et $x_2$ à l'instant (90-130) appliqué aux signaux de l'exemple 1).....	45
Figure IV.4.a Distances entre les données et les neurones gagnants (cas d'un fonctionnement normal appliqué aux signaux de l'exemple 1).....	46
Figure IV.5 Fonction logique pour localisation et identification des défauts.....	47
Figure IV.6 Reconstruction des données.....	48
Figure IV.7 Valeurs propres utilisés pour la reconstruction du modèle ACP de l'exemple 2.....	49
Figure IV.8 Reconstruction des variables sans $x_1$ de l'exemple 2.....	49
Figure IV.9 $In_i$ test sur les dernières composantes après reconstruction.....	50
Figure IV.10 $SPE_i$ après reconstruction des données sans la $i^{ième}$ variable de l'exemple 2.....	50
Figure IV.11 Contribution $cont_i^{SPE}$ au après reconstruction des données sans la $i^{ième}$ variable de l'exemple 2.....	51
Figure IV.12 Reconstruction des données par la carte de Kohonen (SOM) de l'exemple 2..	52
Figure IV.13 Distances entre les données et leurs classes de l'exemple 2 (cas d'un fonctionnement normal et avec des défaut simultanés).....	52
Figure IV.14 Détection et localisation des défauts de l'exemple 2.....	53
Figure A.1 Evolution des différents critères de sélection du nombre de composantes en fonction du nombre de composantes pour l'exemple 2.....	64

## INTRODUCTION GENERALE

Le diagnostic de défauts est aujourd'hui primordial pour la fiabilité, la disponibilité, la maintenabilité et l'efficacité des systèmes ainsi que pour la sécurité ou la protection de l'environnement. Effectuer le diagnostic de fonctionnement d'un système consiste à détecter et localiser les défauts internes (affectant le processus lui-même), les défauts externes (affectant les actionneurs, les capteurs) puis à estimer les caractéristiques principales des défauts mis en évidence (leurs amplitudes). Pour élaborer une procédure de diagnostic, plusieurs méthodes ont vu le jour. On peut les classer en deux catégories : les méthodes basées sur les outils de traitement du signal, et les méthodes à base de modèle. Pour la première catégorie, les outils de traitement du signal font appel à des méthodes statistiques (analyse spectrale, analyse temps-fréquence, ondelettes) ou bien à des méthodes de reconnaissance de formes qui permettent de classer des signatures de défaut. Il faut dans ce cas disposer de suffisamment d'expériences pour pouvoir constituer une base d'apprentissage. (cas des réseaux de neurone). Les méthodes à base de modèle supposent une connaissance approfondie du procédé sous forme de modèle numérique. Le principe consiste à tester la cohérence entre les mesures et les calculs des modèles par des grandeurs caractéristiques de l'état du procédé, appelées résidus. Le vecteur des résidus, nul en fonctionnement normal, est comparé en ligne aux signatures de pannes. La détection est réalisée en vérifiant le dépassement d'un seuil par les résidus et la localisation en reconnaissant la signature.

La suite de notre travail est de quatre principaux chapitres organisés comme suit :

Le premier chapitre aborde les différentes méthodes de diagnostic des systèmes – rencontrées dans la littérature.

Le second chapitre, est consacré aux techniques d'identification, et de localisation des défauts par le modèle de l'analyse en composantes principales (ACP).

Le troisième chapitre, consacré à la partie sensible de notre travail et concerne la méthodologie de détection et de localisation des défauts par carte auto organisatrice de Kohonen

Le quatrième chapitre, réservé à la partie expérimentale. Nous mettons en œuvre pratique les méthodes adoptées aux chapitres précédents (chapitre II et III).

Nous terminons notre travail par une conclusion et les perspectives qui sous-tendent ce travail.

# CHAPITRE I : DIAGNOSTIC : PRINCIPE ET METHODOLOGIES

## I.1 Principe

L'objectif du diagnostic est de constater l'apparition d'un défaut, d'en trouver la cause puis d'en déduire la marche à suivre afin d'assurer la sûreté de fonctionnement d'un procédé.

Dans ce qui suit, nous présentons une revue non exhaustive de plusieurs méthodes de diagnostic à partir de différentes perspectives.

## I.2 Détection de défauts et diagnostic

Pour mieux étayer nos propos, nous introduisons quelques définitions utilisées dans le domaine du diagnostic [TOS 05] :

- **Fonctionnement normal d'un système.** Un système est dit dans un état de fonctionnement normal lorsque les variables le caractérisant (variables d'état, variables de sortie, variables d'entrée, paramètres du système) demeurent au voisinage de leurs valeurs nominales. Le système est dit défaillant dans le cas contraire.
- **Défaut.** C'est une déviation en dehors d'un intervalle acceptable, d'une variable observée ou d'un paramètre associé au procédé [HIM 78]. C'est-à-dire, un défaut est un processus anormal ou symptôme, tel que la hausse de température dans un réacteur ou la baisse de qualité du produit.
- **Défaillance.** C'est la cause d'une anomalie, telle qu'une panne d'une pompe de refroidissement ou d'un régulateur.
- **Détection de défauts.** La détection d'un défaut consiste à décider si le système se trouve ou non dans un état de fonctionnement normal.
- **Localisation d'un défaut.** A l'issue de la détection d'un défaut, il s'agit de déterminer le ou les éléments à l'origine du défaut.

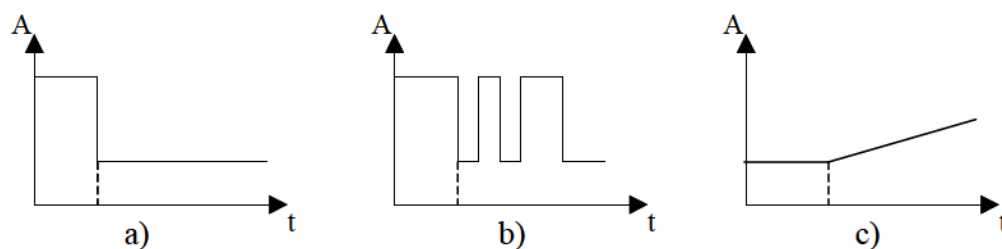
Une classification des défauts à partir de leurs évolutions temporelles les définit comme :

a- **abrupts** : la caractéristique principale de ce type de défauts est la discontinuité dans l'évolution temporelle de la variable. Cette évolution, si elle ne correspond pas aux évolutions dynamiques normales attendues pour la variable (changement de consigne), est caractéristique d'une panne brutale de l'élément en question : arrêt total ou partiel,

b- **intermittents** : il s'agit d'un type de défauts caractéristiques de faux contacts ou de pannes intermittentes de capteurs. C'est un cas particulier de défaut brutal sur un capteur avec perte aléatoire de signal,

c- **graduels** : ce type de défauts est caractéristique d'un encrassement ou d'une dérive dans les paramètres caractéristiques du procédé. Il s'agit de défauts très difficiles à détecter, car leurs évolutions temporelles sont les mêmes que celles d'une modification paramétrique lente représentant une non-stationnarité du procédé.

La Figure I.1 résume cette classification des défauts à partir de leurs évolutions temporelles.



**Figure I.1** Différents types de défauts : a) défaut abrupt, b) défaut intermittent, c) défaut graduel

La présence de défauts multiples constitue une situation fréquente dans la pratique. Cette situation a lieu lorsqu'une ou plusieurs variables (ou paramètres) sortent de leurs évolutions normales à cause d'un dysfonctionnement simultané d'éléments d'une installation.

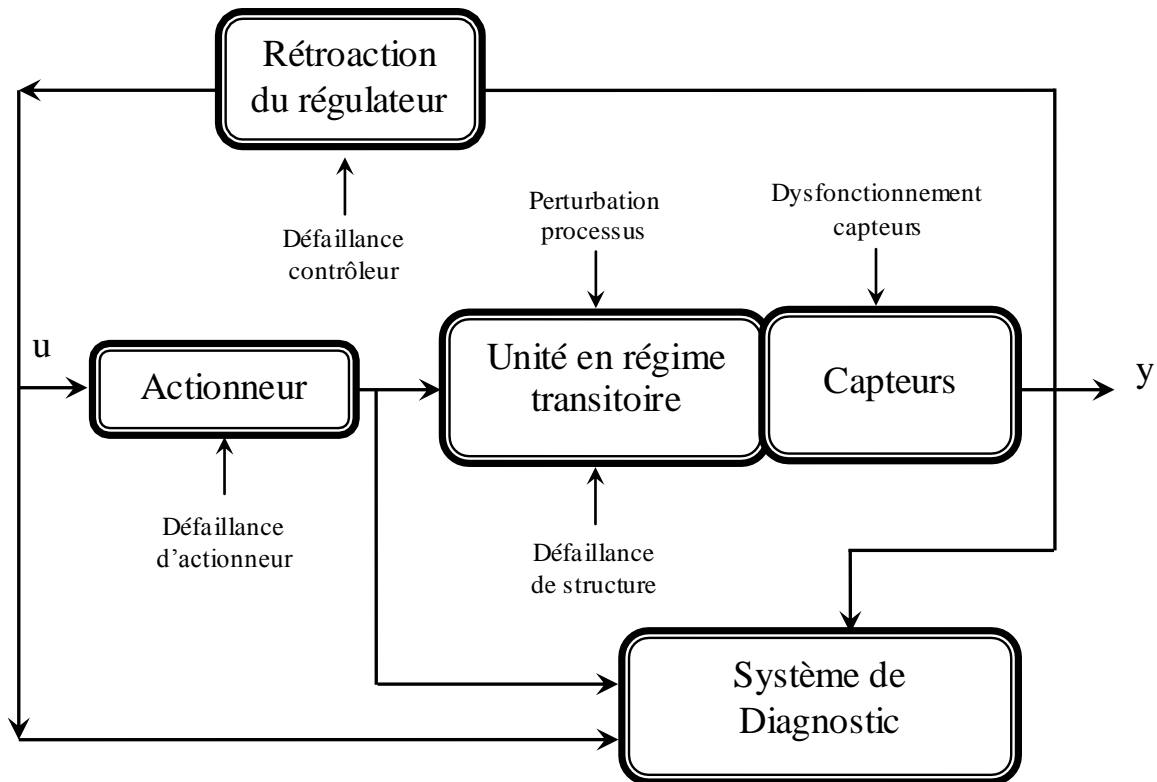
Pour remédier aux états de défaillance et trouver des solutions adaptées à chaque procédé, la détection précoce de défauts, le diagnostic automatisé et l'assistance aux opérateurs sont donc des domaines où la recherche scientifique est très active .

À cause de la difficulté d'effectuer en temps réel la détection de défauts et le diagnostic de procédés, plusieurs approches ont été développées. Citons comme techniques les plus récentes : les arbres de défaillances et digraphes, les approches analytiques, les systèmes à base de connaissance et encore plus récemment les réseaux de neurones.

Dans le cas où une modélisation du procédé est disponible ou est envisageable, il est possible d'utiliser des méthodes de diagnostic qui requièrent soit des modèles précis du procédé, soit des modèles semi-quantitatifs, voire même des modèles qualitatifs.

En revanche, il existe des méthodes qui ne nécessitent aucune modélisation proprement dite, mais qui se basent sur des informations/données historiques représentatives du fonctionnement

du procédé lors de différents scénarios. La Figure I.2 résume le contexte général du diagnostic de défauts.



**Figure I.2** Schéma général d'un système de diagnostic

La figure présente un système de processus contrôlé et indique les différentes origines de ses défaillances. En général, il existe trois classes de défaillance ou de dysfonctionnement décrites comme suit [VEN 03] :

- **Changement paramétrique dans un modèle.** Un exemple de cette défaillance est le changement par exemple dans la concentration d'un réactif à partir de sa valeur normale ou état stable dans l'alimentation d'un réacteur.
- **Changement structurel.** On se réfère aux changements dans le processus même. Quelques exemples sont la défaillance d'un régulateur, la saturation d'une vanne, une fuite dans une canalisation.
- **Défaillances des capteurs et des actionneurs.** Ce peut être une déviation constante (positive ou négative) ou une défaillance de sortie de plage admissible.

Les autres défauts sont : les incertitudes de structure (défauts a priori non modélisés), bruits de processus (différence entre le processus actuel et les prédictions par le modèle) et le bruit de mesure (correspond généralement à l'addition d'un terme de haute fréquence à la sortie des capteurs).

L'ensemble des caractéristiques souhaitées qu'un système de diagnostic devrait posséder [VEN 03] est :

- a) Détection rapide et diagnostic.
- b) Isolation, c'est l'habilité pour différencier les défauts.
- c) Robustesse vis-à-vis de certains bruits et d'incertitudes.
- d) Identification de nouveauté, on se réfère à la capacité de décider si le processus est en état normal ou anormal. Dans le cas d'anomalie, il faut identifier s'il s'agit d'un défaut connu ou d'un nouveau défaut.
- e) Estimation de l'erreur de classification du défaut (diagnostic) en vue de sa fiabilité.
- f) Adaptabilité : le système de diagnostic devrait être adaptable aux changements de conditions du processus (perturbations, changements d'environnement).
- g) Facilité d'explication de l'origine des défauts et de la propagation de celui-ci. Ceci est important pour la prise de décision en ligne.
- h) Conditions de modélisation : pour le déploiement rapide et facile des classificateurs de diagnostic en temps réel, l'effort de modélisation devrait être aussi minimal que possible.
- i) Facilité de mise en œuvre informatique (faible complexité dans les algorithmes et leur implémentation) et capacité de stockage.
- j) Identification de multiples défauts : pour de grands processus, l'énumération combinatoire de multiples défauts est trop importante et ils ne peuvent être explorés de manière exhaustive.

### **I.3 Classification des méthodes de diagnostic**

Les premières méthodes de diagnostic furent basées sur la redondance de matériels jugés critiques pour le fonctionnement du système. La redondance matérielle est très répandue dans les domaines où la sûreté de fonctionnement est cruciale pour la sécurité des personnes et de l'environnement, comme dans l'aéronautique ou le nucléaire. Les principaux inconvénients de la redondance matérielle sont liés aux coûts dus à la multiplication des éléments ainsi que l'encombrement et aux poids supplémentaires qu'elle génère.

Les spectaculaires progrès réalisés dans le domaine des calculateurs numériques combinés à une baisse des coûts permettent aujourd'hui la mise en œuvre, dans le milieu industriel, des méthodes modernes de l'automatique et de l'intelligence artificielle. Cette nouvelle approche permet d'éliminer en partie, voire même en totalité, la redondance matérielle pour le diagnostic

des systèmes industriels. On peut globalement distinguer deux grandes familles dans les méthodes de diagnostic :

- Les méthodes basées sur une modélisation des systèmes ou des signaux, que nous dénommerons « diagnostic quantitatif ».
- Les méthodes basées sur l'intelligence artificielle que nous appellerons « diagnostic qualitatif ».

Le fait de distinguer ce qui est de l'ordre du quantitatif et du qualitatif, ne doit pas laisser penser que ces deux aspects sont disjoints. En réalité, ces deux types d'approche coexistent souvent au sein d'un même système de diagnostic. L'utilisation conjointe de méthodes quantitatives et qualitatives permet l'exploitation de l'ensemble des connaissances disponibles concernant le fonctionnement du système.

Nous présentons à la Figure I.3 un panorama général des différentes méthodes de diagnostic dans l'une ou l'autre des catégories précédemment présentées

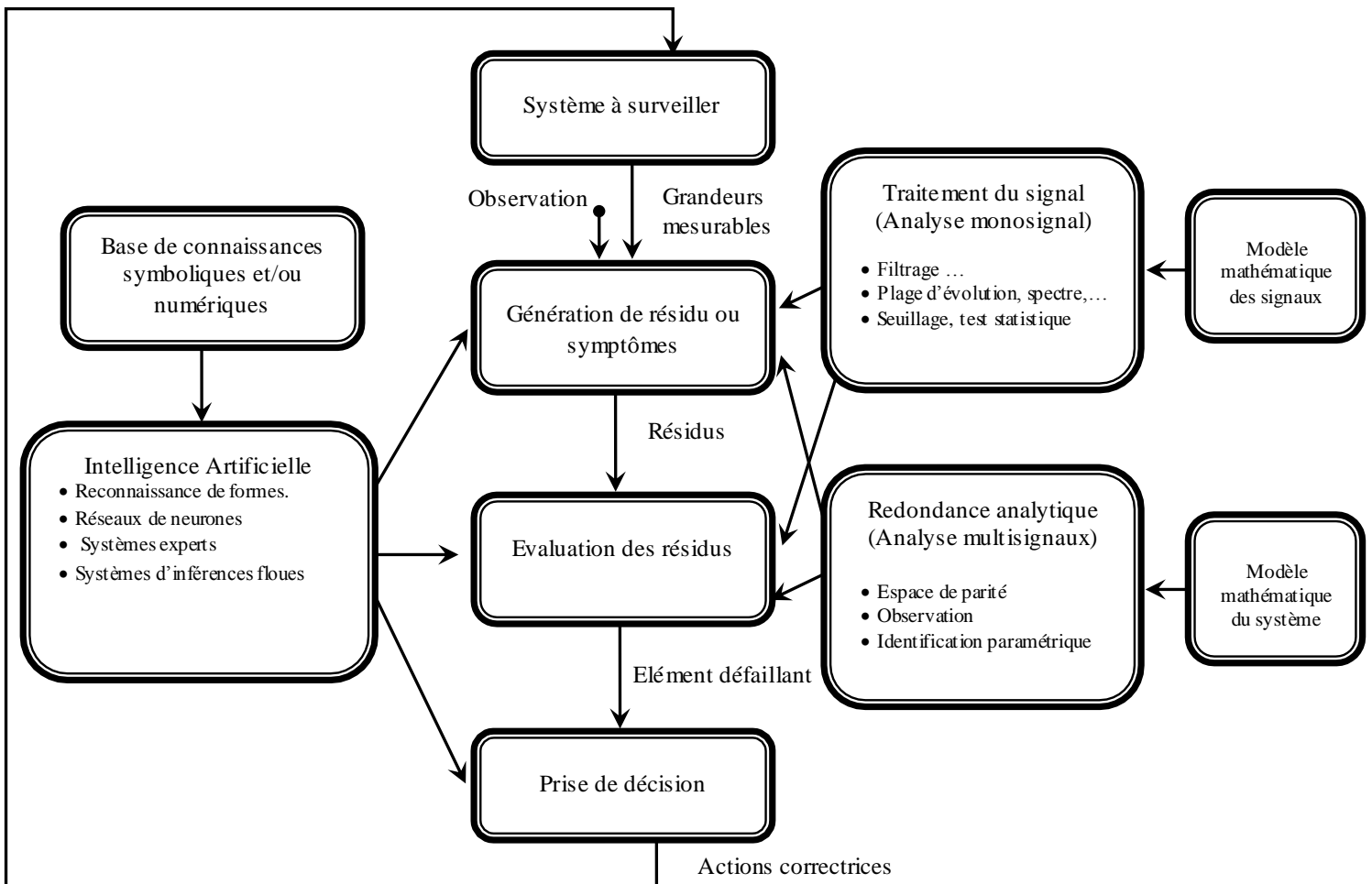


Figure I.3 Classification des méthodes de diagnostic

En ce qui concerne le diagnostic quantitatif, on peut distinguer deux types d'approches suivant que l'on considère les mesures prises isolément les unes des autres ou qu'au contraire on présuppose des relations mathématiques les reliant. La première approche est connue sous le nom d'analyse mono signal, la deuxième est dénommée analyse multi-signaux ou redondance analytique.

L'analyse mono-signal consiste dans l'étude des signaux, pris isolément les uns des autres, dans le but d'extraire des informations révélatrices de défauts. Cette approche repose sur les techniques de traitement du signal. La méthode la plus simple consiste à fixer un ou plusieurs seuils limitant la plage d'évolution du signal considéré. D'autres approches, plus élaborées, sont basées sur les propriétés statistiques des signaux (moyenne, écart type, etc.), ou sur l'analyse spectrale.

L'analyse multi-signaux met en jeu un ensemble de variables et les relations de causalité existant entre elles. Cette approche repose sur les techniques de l'automatique. En automatique, le concept clé est celui de modèle mathématique d'un processus physique. Ce modèle mathématique, censé représenter une réalité objective, est exploité afin d'élaborer des lois de commande permettant l'accroissement des performances dynamiques et statiques du système étudié. Cette connaissance analytique d'un système physique peut être mise à profit pour le diagnostic. En effet, le modèle étant censé décrire le comportement dynamique du système à surveiller, tout écart entre le comportement prévu par le modèle et le comportement mesuré traduit l'apparition d'un défaut. Dans la suite, nous entendrons essentiellement par diagnostic quantitatif, l'ensemble des méthodes fondées sur l'utilisation d'une représentation mathématique du système à surveiller.

Dans le cas du diagnostic qualitatif, les connaissances utilisables reposent sur le savoir et l'expérience d'opérateur humain ayant une parfaite maîtrise de l'installation à surveiller (base de connaissance symbolique) et/ou sur l'existence d'une base de connaissances numériques correspondant aux divers modes de fonctionnement de l'installation. Suivant cette approche, on trouve toutes les méthodes liées à l'intelligence artificielle. Elle inclut les systèmes experts, les systèmes d'inférence floue, la reconnaissance de forme, les réseaux neurones. Ces différentes approches sont alors utilisées pour construire un modèle de diagnostic qualitatif permettant de remonter des symptômes observés aux causes.

### I.3.1 Diagnostic Quantitatif

#### A) Les méthodes basées sur l'approche mono-signal :

- **Redondance matérielle** : c'est une méthode employée dans des installations critiques (l'aérospatiale, le nucléaire). L'utilisation de plusieurs capteurs en vue d'obtenir la même information sur une variable permet de détecter les déviations par rapport à un état normal et de localiser un défaut de capteur. Les inconvénients de cette méthode sont l'accroissement du coût de l'installation et l'augmentation de la probabilité de pannes de capteurs – donc d'un besoin de maintenance supplémentaire.

- **Analyse spectrale** : Les signaux sont analysés en état normal de fonctionnement; les hautes fréquences sont reliées au bruit et les basses fréquences aux évolutions propres de l'état du procédé. Ensuite, toute déviation des caractéristiques fréquentielles d'un signal est reliée à une situation de défaillance. Cette approche se révèle très utile pour analyser des signaux qui montrent des oscillations avec des périodes longues (les courants électriques, les débits, les pressions...). L'inconvénient est la sensibilité aux bruits de mesure quand ceux-ci coïncident avec la zone fréquentielle d'intérêt et la nécessité d'un échantillonnage fréquent pour permettre de reconstituer le signal de départ tout en minimisant la perte de fréquence. Les méthodes d'auto-corrélation, la densité spectrale des signaux, la transformée de Fourier, les ondelettes sont bien appropriées dans le cas où les fréquences représentatives de défauts sont connues. Dans le cas contraire, il est cependant préférable d'utiliser des modèles paramétriques des signaux qui permettent d'estimer en ligne les fréquences et les valeurs moyennes des paramètres.

- **Approches statistiques** : Ces approches se basent sur l'hypothèse de changements rapides (et non sur l'amplitude) des caractéristiques des signaux ou des paramètres des modèles par rapport à des dynamiques considérées comme étant lentes (procédés quasi stationnaires). Elles sont utilisées pour la détection de changements graduels avec des seuils de détection faibles. Les informations fournies par le nombre croissant de capteurs installés sur les procédés rendent très difficile l'analyse des résultats. L'analyse en composantes principales (*ACP*) et les moindres carrés partiels (*MCP*) permettent de réduire le nombre de variables à traiter par la détermination de relations linéaires entre elles (variables latentes), permettant ainsi d'expliquer la variance dans des séries de données.

B) **L'approche multi-signal**. Ce sont de méthodes qui utilisent plus d'information que celles apportées par les seuls capteurs physiques. Ces informations peuvent provenir de la

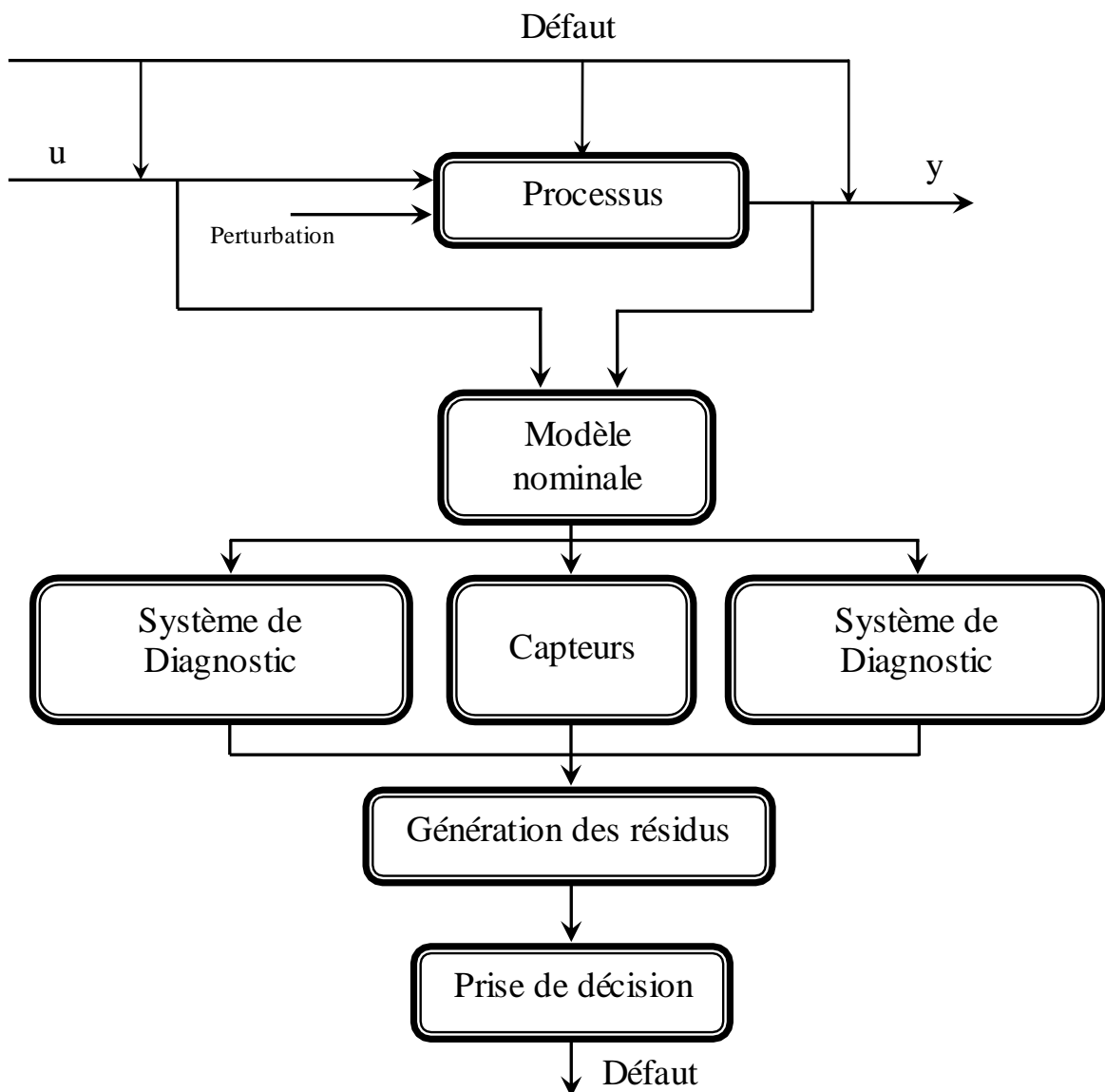
connaissance du comportement entrée/sortie d'un procédé ou des processus internes qui en gouvernent l'évolution.

Selon la méthode, différents types de modèle sont utilisés. Par exemple, pour les approches utilisant l'estimation d'état ou l'estimation paramétrique, on utilisera des modèles analytiques alors que pour l'approche systèmes experts, on recourra à des modèles de type base de connaissance. Le modèle servant directement de référence pour la détection de défauts, la qualité du résultat dépend directement de la qualité des modèles. La mise en œuvre de ces méthodes nécessite donc une modélisation précise.

La détection de défaut basée sur l'utilisation de modèles peut être divisée en deux étapes principales : la génération de résidus et la prise de décision. Lors de la première étape, les signaux d'entrée et de sortie du système sont utilisés pour générer un résidu c'est-à-dire un signal mettant en évidence la présence d'un défaut. En général, en régime de fonctionnement normal, ce signal est statistiquement nul et s'écarte notablement de zéro en présence de défaut. La génération de résidus est propre à la méthode utilisée. Durant la seconde étape, les résidus sont analysés pour décider s'il y a ou non présence de défaut, sur quelle composante du système il est intervenu (localisation) et pour déterminer la nature du défaut et sa cause (identification). La décision peut s'effectuer à l'aide d'un simple test de dépassement de seuil sur les valeurs instantanées ou les moyennes de résidus, en utilisant des fonctions floues, en faisant appel également à la reconnaissance de formes ou en utilisant des seuils adaptatifs qui évoluent en fonction du point de fonctionnement du processus surveillé.

La Figure I.4 présente l'architecture générale de la détection de défauts basée sur l'utilisation de modèles [ISE 84]. La génération de résidus s'effectue sur la base d'estimations (états ou paramètres) ou à l'aide d'un raisonnement heuristique. L'étape de prise de décision permet ensuite de détecter la présence de changements, d'identifier la nature, l'amplitude du changement voire localiser les défauts éventuels.

Parmi les différentes méthodes de détection et de diagnostic utilisant des modèles mathématiques, nous trouvons principalement l'espace de parité, les observateurs et l'estimation paramétrique



**Figure I.4** Architecture générale de la détection de défauts à base de modèles

**A) Espace de parité** : Une relation de redondance analytique est une équation dans laquelle toutes les variables sont connues. La génération de telles relations permet d'engendrer des résidus. Pour la détection de défauts basée sur l'utilisation de modèles, un résidu est un signal temporel, fonction des entrées et des sorties du processus, indépendant (le plus possible) du point de fonctionnement de celui-ci. En l'absence de défauts, ce résidu est statistiquement nul. Lors de l'apparition d'un défaut, son amplitude évolue de manière significative. L'approche la plus classique est celle dite de l'espace de parité. Les relations de parité utilisent la redondance directe au moyen de relations algébriques statiques liant les différents signaux ou la redondance temporelle issue de l'utilisation de relations dynamiques. Le terme « parité »

a été emprunté au vocabulaire employé pour les systèmes logiques où la génération de bits de parité permet la détection d'erreur.

**B) Observateurs** : La génération de résidus à l'aide d'une estimation d'état consiste à reconstruire l'état ou, plus généralement, la sortie du processus à l'aide d'observateurs et à utiliser l'erreur d'estimation comme résidu. Cette méthode s'est beaucoup développée car elle donne lieu à la conception de générateurs de résidus flexibles.

**C) Estimation paramétrique** : L'approche d'estimation paramétrique considère que l'influence de défauts se reflète sur les paramètres et non pas uniquement, comme c'est le cas des observateurs, sur les variables du système physique. Le principe de cette méthode consiste à estimer en continu des paramètres du procédé en utilisant les mesures d'entrée/sortie et en évaluant la distance qui les sépare des valeurs de référence de l'état normal du procédé. L'estimation paramétrique possède l'avantage d'apporter de l'information sur l'importance des déviations. Toutefois, un des Inconvénients majeurs de la méthode réside dans la nécessité d'avoir un système physique excité en permanence. Ceci pose donc des problèmes d'ordre pratique dans le cas de procédés dangereux, coûteux ou fonctionnant en mode stationnaire. De plus, les relations entre paramètres mathématiques et physiques ne sont pas toujours inversibles de façon unitaire, ce qui complique la tâche du diagnostic basé sur les résidus.

Les principaux outils numériques d'analyse de résidus sont les méthodes de décision statistiques, visant à affirmer ou à infirmer qu'un résidu est nul, qui permettront d'engendrer les signatures expérimentales de défauts.

### **I.3.2 Diagnostic Qualitatif**

Parmi les différentes méthodes de détection et de diagnostic utilisant des modèles de raisonnement heuristique, nous trouvons principalement les suivantes :

**A) Modèles issus de l'intelligence artificielle (IA)** : De manière globale, trois types de modèle sont utilisés en IA dans le contexte de la surveillance :

**a.** Dans les modèles associatifs, les situations significatives de dysfonctionnement sont associées à des schémas d'alarmes par l'expression directe des liens entre les symptômes et la situation caractéristique. Ils reposent sur l'expertise existante, qu'elle soit issue des opérateurs du procédé ou des ingénieurs de maintenance. De tels modèles sont en particulier utilisés dans les systèmes experts et dans la reconnaissance de scénarii de défaillance.

b. Les modèles prédictifs permettent de simuler les comportements possibles d'un système dans ces différents modes (normal, dégradé ou défaillant). Par leur capacité à calculer pas à pas les comportements prédits, ces modèles sont directement utilisables pour la détection de défauts (par comparaison des comportements prédits avec les observations). Ils peuvent s'obtenir par abstraction de modèles numériques, par réseaux bayésiens ou réseaux probabilistes temporels.

c. Les modèles explicatifs, s'appuient sur l'expertise déjà explicitée sur l'unité industrielle sous forme de catalogue de dysfonctionnements ou de graphes.

B) **Réseaux de neurones artificiels (RNA)**: Un RNA est un système informatique constitué d'un nombre de processeurs élémentaires (ou noeuds) interconnectés entre eux qui traite – de façon dynamique – l'information qui lui arrive à partir des signaux extérieurs. De manière générale, l'utilisation des RNA se fait en deux phases. Tout d'abord, la synthèse du réseau est réalisée et comprend plusieurs étapes : le choix du type de réseau, du type de neurones, du nombre de couches, des méthodes d'apprentissage. L'apprentissage permet alors, sur la base de l'optimisation d'un critère, de reproduire le comportement du système à modéliser. Il consiste dans la recherche d'un jeu de paramètres (poids) et peut s'effectuer de deux manières : supervisé (le réseau utilise les données d'entrée et de sortie du système à modéliser) et non supervisé (seules les données d'entrée du système sont fournies et l'apprentissage s'effectue par comparaison entre exemples). Quand les résultats d'apprentissage obtenus par le RNA sont satisfaisants, il peut être utilisé pour la généralisation. Il s'agit ici de la deuxième phase où de nouveaux exemples – qui n'ont pas été utilisés pendant l'apprentissage – sont présentés au RNA pour juger de sa capacité à prédire les comportements du système ainsi modélisé.

Leur faible sensibilité aux bruits de mesure, leur capacité à résoudre des problèmes non linéaires et multi-variables, à stocker les connaissances de manière compacte, à « **apprendre** » en ligne et en temps réel, sont des propriétés qui rendent l'utilisation des RNA attrayante. Leur emploi peut alors se faire à trois niveaux :

- comme modèle du système à surveiller en état normal et générer un résidu d'erreur entre les observations et les prédictions,
- comme système d'évaluation de résidus pour le diagnostic,
- ou comme système de détection en une seule étape (en tant que classificateur), ou en deux étapes (pour la génération de résidus et le diagnostic).

C) **Systèmes d'inférence flous (SIF)** : La structure de base d'un SIF est constituée :

a. d'un univers de discours qui contient les fonctions d'appartenance des variables d'entrée et de sortie à des classes. Ces fonctions peuvent avoir différentes formes, les plus usuelles étant les formes triangulaires, trapézoïdales et gaussiennes

b. d'une base de connaissance qui regroupe les règles liant les variables d'entrée et de sortie sous la forme « *SI...ALORS* ».

c. d'un mécanisme de raisonnement qui base son fonctionnement sur la logique du *modus ponens* généralisé.

Les formalismes les plus utilisés pour les *SIF* sont ceux de Mamdani [MAM 77] et de Takagi-Sugeno [TAK 85].

**D) Reconnaissance de formes :** La reconnaissance des formes regroupe l'ensemble des méthodes permettant la classification automatique d'objets, suivant sa ressemblance par rapport à un objet de référence. L'objectif est de décider, après avoir observé un objet, à quelle forme type celui-ci ressemble le plus. Une forme est définie à l'aide de  $p$  paramètres, appelés descripteurs, qui sont les composantes d'un vecteur forme que nous noterons  $\mathbf{X}$ .

La résolution d'un problème de reconnaissance de formes nécessite :

- La définition précise des  $l$  classes entre lesquelles va s'opérer la décision. Cette étape est liée à la nature des objets à classer et est donc spécifique au problème posé.

- Le choix d'un jeu de caractères pertinents pour la discrimination des vecteurs de formes. Malheureusement, il n'existe pas de méthodes systématiques permettant de choisir les paramètres les plus appropriés à la résolution d'un problème donné. Par conséquent, seule une bonne connaissance du problème à résoudre peut permettre un choix adéquat des paramètres de forme (ou descripteurs). Notons enfin que le nombre de caractères fixe la dimension de l'espace de représentation et, par conséquent, la quantité des calculs à mener. Ceci peut représenter une contrainte sévère dans un contexte de traitement en temps réel des objets à classer.

- L'élaboration d'un classificateur permettant l'affectation d'une forme observée à l'une des classes. Le classificateur est généralement établi à l'aide d'un ensemble d'apprentissage constitué de formes pour lesquelles on connaît l'appartenance aux différentes classes.

Les méthodes présentées précédemment sont complémentaires entre elles. Il n'est donc pas étonnant de trouver des applications où les différents niveaux de traitement (signaux, modèles et connaissance experte) soient intégrés dans une même structure.

D'autres articles traitent du domaine du diagnostic de défauts basé sur des modèles [FRA 00], sur le diagnostic de défauts par extraction de caractéristiques et classification de problèmes

(défauts). La classification des défauts est examinée par Kramer et Mah [KRA 93] selon trois catégories principales :

- i) Reconnaissance de formes : ici sont traitées les méthodes basées sur l'historique du processus.
- ii) Modèles basés sur le raisonnement : ils utilisent les techniques basées sur des modèles qualitatifs.
- iii) Les modèles causaux : dans ce cas, les techniques de recherche de symptômes en utilisant différents modèles sont traités.

Une autre méthodologie de diagnostic de défaillances à prendre en compte est celle basée sur des modèles à événements discrets (SED) [LAF 91, GAR 98] et sur le concept de diagnostiqueur [SAM 94], [SAM 95; GAR 99], lesquels permettent tant l'analyse **hors ligne**, qu'**en ligne** de la diagnosabilité des systèmes, relative à des défaillances qui peuvent se produire dans les processus. Le diagnostiqueur a comme objectif de détecter la cause de quelques situations (états fonctionnels) du procédé, en un temps fini, comme la conséquence d'une défaillance. Tant les SED que le diagnostiqueur sont des machines à états finis (MEF). La méthodologie du diagnostic à base de modèles à événements discrets [RAM 87] est applicable sur beaucoup de systèmes industriels. Toutefois, malgré son intérêt notable, elle présente l'inconvénient des systèmes partiellement observables, dans le sens de l'inévitable augmentation exponentielle des états produits au fur et à mesure qu'on incorpore de nouveaux modèles de dispositifs qui forment des parties du système ou que s'ajoutent de nouvelles situations de défaillance. Pour éviter l'augmentation des états, une conception modulaire du système a été adoptée, en divisant le système en sous-systèmes. La complexité des systèmes peut être considérée en tenant en compte du nombre de sous-systèmes liés entre eux. L'application de cette méthodologie est restreinte par la complexité du procédé.

## I.4 Méthodes de classification

Différentes approches pour la classification non supervisée « *clustering* » de données peuvent être décrites avec l'aide de la structure hiérarchisée de la Figure I.5 [JAI 88]. Sur ce graphe, il y a distinction entre les approches par hiérarchisation et par partitionnement. Les méthodes par hiérarchisation produisent une série de partitions liées, alors que les méthodes de partitionnement n'en produisent qu'une seule.

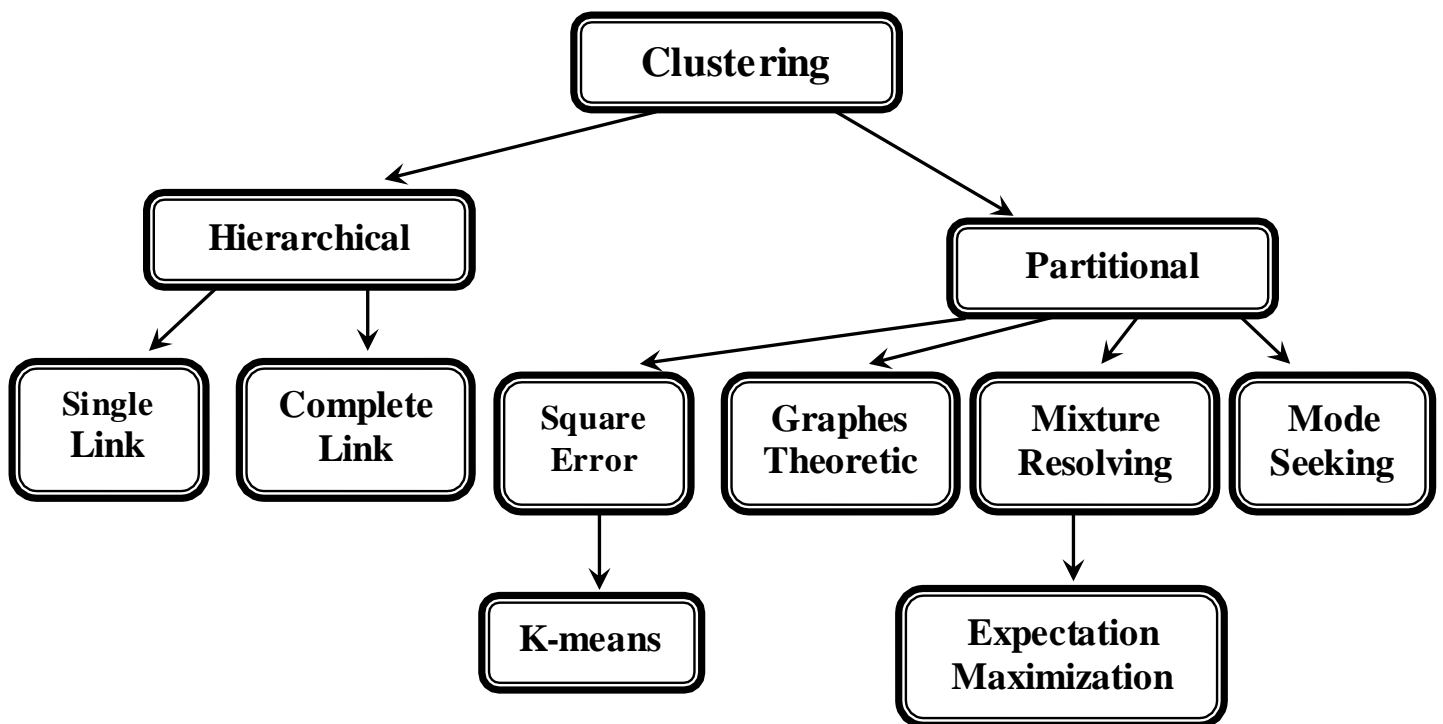


Figure I.5 Différentes approches de classification non supervisée.

Nous donnons, maintenant, une brève présentation des principales approches de classification non supervisée que nous considérons appropriées pour la mise en place de la stratégie proposée.

#### I.4.1 Algorithmes par hiérarchisation

L'algorithme par hiérarchisation crée une décomposition de données (ou d'éléments) en utilisant un critère. Il produit un dendrogramme qui représente le regroupement des éléments et les niveaux de similarité, permettant de faire figurer les changements de regroupement.

Les autres algorithmes de classification similaires à celui développé par [SNE 73] sont : l'algorithme *complete-link* par [KIN 67], et le *minimum de variance* [WAR 63; MUR 84]. Les deux algorithmes les plus populaires sont le *single-link* et le *complete-link*. Ces deux algorithmes se différencient dans la façon de caractériser la similarité entre une paire de classes. Dans la méthode *single-link*, la distance entre deux classes est le minimum des distances entre toutes les paires d'éléments appartenant à chaque classe (un élément dans la première classe, l'autre dans la seconde). Dans l'algorithme *complete-link*, la distance entre deux classes est le maximum des distances entre toutes les paires d'éléments sur les deux classes.

### 1.4.2 Algorithmes par partitionnement

Un algorithme de classification par partitionnement obtient une unique partition des données au lieu d'une structure hiérarchisée comme dans le cas précédent. Les méthodes de partitionnement ont l'avantage de manipuler une grande quantité d'ensembles de données, pour lesquelles, la construction d'un graphe hiérarchisé permettant de tout couvrir serait difficile. Un des problèmes relatifs aux algorithmes de partitionnement est le choix du nombre de classes désirées en sortie. Dans [DUB 87] un guide pour choisir ce paramètre clé pour la conception est donné. Cette technique produit des classes en optimisant une fonction critère définie soit localement (sur un sous-ensemble d'éléments) soit globalement (définie sur l'ensemble des éléments). L'algorithme est réitéré de multiple fois avec différents nombres de classes de sortie, et la meilleure configuration obtenue est utilisée pour le choix de la classe de sortie.

#### - Algorithmes de minimisation de l'erreur quadratique

Cet algorithme est basé sur la minimisation d'un critère quadratique, critère qui est le plus utilisé dans les algorithmes de partitionnement, et qui est bien adapté dans le cas de classes séparées et compactes. L'erreur quadratique pour une classe  $L$  parmi  $K$  classes d'un ensemble d'éléments  $H$  est définie par :

$$e^2(H, L) = \sum_{j=1}^K \sum_{i=1}^{n_j} \|x_i^{(j)} - c_j\|^2 \quad (1.1)$$

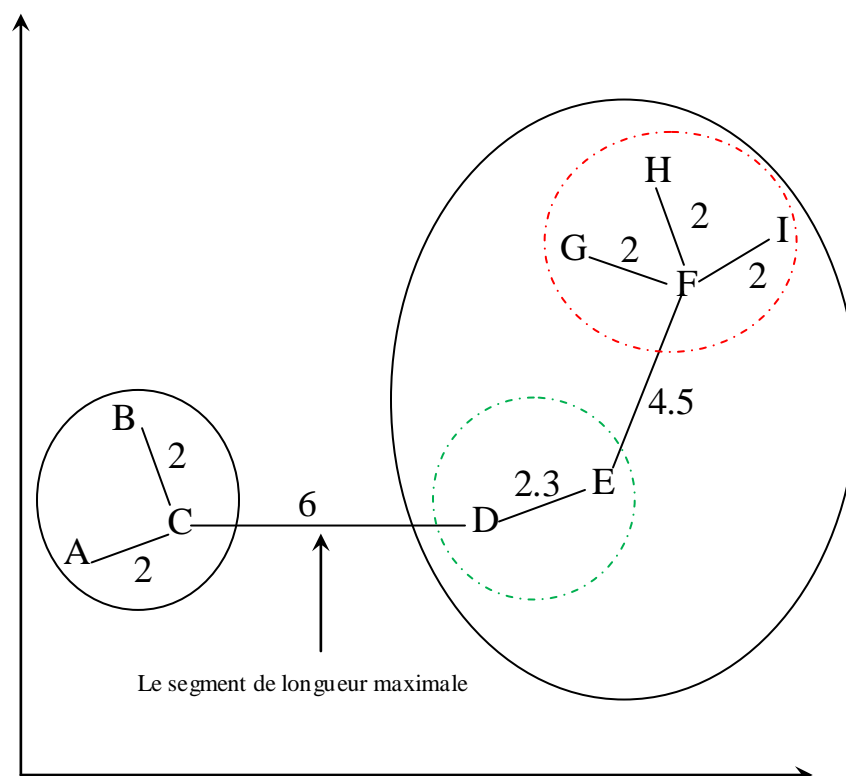
Où  $x_i^{(j)}$  est le  $i^{\text{ème}}$  élément qui appartient à la  $j^{\text{ème}}$  classe et  $c_j$  est le centre de la  $j^{\text{ème}}$  classe.

L'algorithme *k-means* est le plus simple et le plus connu des algorithmes de minimisation d'un critère quadratique [MCQ 67]. Il débute avec une partition initiale aléatoire et il déplace les éléments dans les classes sur la base de la similarité entre l'élément et le centre de la classe par minimisation du critère ((1.1) jusqu'à convergence (quand il n'y a pas de réaffectation d'élément d'une classe à une autre, ou lorsque l'erreur quadratique ne décroît plus significativement après un certain nombre d'itérations)). L'algorithme *k-means* est populaire parce qu'il est facile à implanter. Un des problèmes majeurs de cet algorithme est qu'il est sensible au choix de la partition initiale et qu'il peut converger vers un minimum local de la fonction critère si la partition initiale n'est pas choisie de manière adéquate.

### I.4.3 Algorithmes issus de la théorie des graphes

Ce type d'algorithmes utilise le principe de l'algorithme STA (« *Spanning Tree Algorithm* »). L'algorithme STA est un protocole de dialogue entre tous les ponts permettant de supprimer des liens redondants et de ne garder que les liens les plus intéressants (lignes de communications les plus rapides, les moins coûteuses, etc.).

Le plus connu de ces algorithmes est basé sur la construction du « MST » (*minimal spanning tree*) des données [ZAN 71]. Les segments du MST avec les plus grandes distances sont ensuite supprimés pour produire les classes. La Figure I.6 représente le MST obtenu à partir de neuf points en deux dimensions. En supprimant le segment étiqueté CD avec une longueur de 6 unités (le segment avec la distance Euclidienne maximale), deux classes ( $\{A, B, C\}$  et  $\{D, E, F, G, H, I\}$ ) sont obtenues. La deuxième classe peut être ultérieurement divisée en deux classes en supprimant la branche EF, laquelle a une longueur de 4.5 unités.



**Figure I.6** Classes formées en utilisant le MST (« *Minimal Spanning Tree* »).

#### **I.4.4 Algorithme de résolution mixte « Mixture-Resolving »**

Le fondement de l'approche « *Mixture-Resolving* » est que les éléments sont obtenus à partir de plusieurs distributions (par exemple, gaussiennes), et le but est d'identifier les paramètres ainsi que, leur nombre. Il consiste à obtenir (itérativement) une estimation de la probabilité maximale (Maximisation de l'espérance mathématique) du vecteur de paramètres des composants de densités [JAD 88].

#### **I.4.5 Classification par le plus proche voisin**

Un procédure itérative a été proposée par [LU 78]. Elle assigne chaque élément non étiqueté à la classe de son plus proche élément voisin étiqueté, si la distance au voisin étiqueté est au-dessous d'un seuil. Le processus continue jusqu'à ce que tous les éléments soient étiquetés ou qu'il ne se produise plus d'étiquetage.

#### **I.4.6 Classification Floue**

Les approches traditionnelles de classification génèrent des partitions; dans une partition, chaque élément appartient à une seule classe. La classification floue amplifie cette notion en associant chaque élément à chacune des classes en utilisant des fonctions d'appartenance [ZAD 65].

### **I.5 La méthode LAMDA**

LAMDA (en anglais : « *Learning Algorithm for Multivariate Data Analysis* ») [AGU 80 ; KOO 89] est une technique de classification qui a été développée au Laboratoire d'Analyse et Architecture des Systèmes (LAAS). C'est un Algorithme d'Analyse de Données Multidimensionnelles par Apprentissage et Reconnaissance de Formes. La formation et la reconnaissance de classes dans cette méthode sont basées sur l'attribution d'un élément à une classe à partir de la règle heuristique appelée *adéquation maximale*. Cette règle d'adéquation a été proposée par Aguilar-Martin [AGU 80]. Chaque élément est représenté par des descripteurs. Le degré d'adéquation maximale d'un élément à une classe est déterminé à partir des degrés d'adéquation marginale de chaque descripteur dans les différentes classes.

## I.6 Conclusions

L'objectif du diagnostic est de détecter rapidement les divers défauts existant sur un procédé (défaut d'instrumentation, de paramètres, structurel) pour éviter la dégradation de ses performances et augmenter la sécurité des opérateurs et de l'environnement.

Dans ce chapitre nous avons fourni une présentation succincte des différentes techniques de diagnostic classiquement utilisées.

Parmi les méthodes de diagnostic ne nécessitant pas de modèle mathématique explicite, se situent les techniques de classification. Un système de diagnostic peut être considéré en effet comme un classificateur, puisque la tâche de diagnostic peut être vue comme une classification de problèmes.

Deux techniques de classification ont suscité notre attention : La première, est une méthode de diagnostic quantitative. Elle est destinée à détecter et localiser les défauts par Analyse en Composante Principale (ACP). La seconde, basée

sur l'intelligence artificielle, est une méthode de diagnostic qualitatif. Elle est destinée à détecter et localiser les défauts par utilisation de carte auto adaptative ou auto organisatrice.

Ce type de carte correspond à une classe de réseau de neurones artificiels fondée sur des méthodes d'apprentissage non supervisée. On la désigne souvent par le terme anglais *self organizing map* (SOM), on encore carte de Kohonen. Les mécanismes de détection et de localisation de défauts qui sous tendent notre approche seront développés de manière plus détaillée aux chapitres suivants.

# CHAPITRE II : DETECTION ET LOCALISATION DES DEFAUTS PAR ACP

## II.1 Principe

L'analyse en composantes principales (ACP) est une technique principale utilisée pour la compression de données et d'extraction d'information. La philosophie de cette approche est de réduire la dimensionnalité des données d'origine en formant un nouvel ensemble de variables latentes qui sont une combinaison linéaire des données d'origine, sans perte d'information essentielle.

Pour le diagnostic des défauts, le but de cette méthode est de détecter en temps réel des défauts issus de capteurs par exemple et de déterminer leur origine. Elle permet, pour chaque nouvelle mesure, de décider de sa validité par comparaison à un modèle mathématique représentant le fonctionnement normal du processus. La première phase consiste donc à établir ce modèle. Il est obtenu à l'aide d'une Analyse en Composantes Principales d'un tableau de données sans défaut. Cette démarche permet d'obtenir une représentation globale du comportement des capteurs dans un espace de dimension réduite. Cette première étape est commune aux méthodes présentées. La seconde phase est celle de détection des défauts proprement dite. A chaque nouvelle mesure, on effectue un test de cohérence entre celle-ci et le modèle. Si ce test est positif, la mesure est considérée comme valide. Si ce n'est pas le cas, on peut conclure à l'existence d'un défaut. Cependant, on ne peut pas dire quelle variable est incriminée. Il faut alors passer par une troisième étape : la localisation. Pour ce faire, on remplace la première variable par une valeur de substitution. Cette valeur, cohérente avec le modèle, est par une technique de reconstruction des variables à partir du modèle et des valeurs des autres variables. On effectue un nouveau test de cohérence entre ce nouveau vecteur de mesures et le modèle. Si ce test devient positif, la première variable est soupçonnée. On procède de la même manière avec la deuxième, puis la troisième variable, et ainsi de suite jusqu'à la dernière.

## II.2 Modélisation par ACP

Soit  $X \in \mathcal{R}^m$  un vecteur d'observation de  $m$  capteurs. On suppose qu'il y a  $n$  mesures pour chaque capteur. La matrice des données  $X$  est donc de taille  $n \times m$ , chaque ligne représentant une observation et chaque colonne une variable. Les données contenues dans cette matrice sont centrées réduites (voir ANNEXE), c'est à dire que chacune des colonnes est de moyenne nulle et de variance égale à 1.

L'analyse en composantes principales permet de décomposer la matrice  $X$  de la manière suivante [STE 97]:

$$X = TP^T \quad (2.1)$$

où les colonnes de la matrice  $T$  sont les composantes principales de  $X$  et celles de  $P$  les vecteurs propres normés de la matrice de corrélation des données  $R$ .

On a alors, pour tout  $\ell$  compris entre 1 et  $m - 1$  :

$$\begin{aligned} X &= \sum_{i=1}^m t_i p_i^T = \sum_{i=1}^{\ell} t_i p_i^T + \sum_{i=\ell+1}^m t_i p_i^T \\ &= \hat{T} \hat{P}^T + \tilde{T} \tilde{P}^T \\ &= \hat{X} + \tilde{X} \end{aligned} \quad (2.2)$$

où  $t_i$  et  $p_i$  sont, respectivement, les colonnes de  $T$  et de  $P$ .

Si

$$C = \hat{P} \hat{P}^T \quad (2.3)$$

on a de plus

$$XC = TP^T \hat{P} \hat{P}^T = T \begin{pmatrix} I_{\ell} \\ (0) \end{pmatrix} \hat{P}^T = \hat{T} \hat{P}^T = \hat{X} \quad (2.4)$$

où  $I_{\ell}$  désigne la matrice identité d'ordre  $\ell$  et  $(0)$  la matrice nulle de taille  $m - \ell \times \ell$ .

De même, si

$$\tilde{C} = \tilde{P} \tilde{P}^T = I - C \quad (2.5)$$

alors

$$X\tilde{C} = \tilde{X} \quad (2.6)$$

Les vecteurs de  $P$  forment une base orthonormée de  $\mathfrak{R}^m$ . Si on ne conserve que les  $\ell$  premières composantes principales, on peut décomposer  $\mathfrak{R}^m$  en deux sous espaces orthogonaux : le sous espace principal (PCS, Principal Component Subspace), engendré par les vecteurs  $p_1, \dots, p_{\ell}$  et le sous-espace résiduel (RS, Residual Subspace), engendré par les vecteurs  $p_{\ell+1}, \dots, p_m$ .

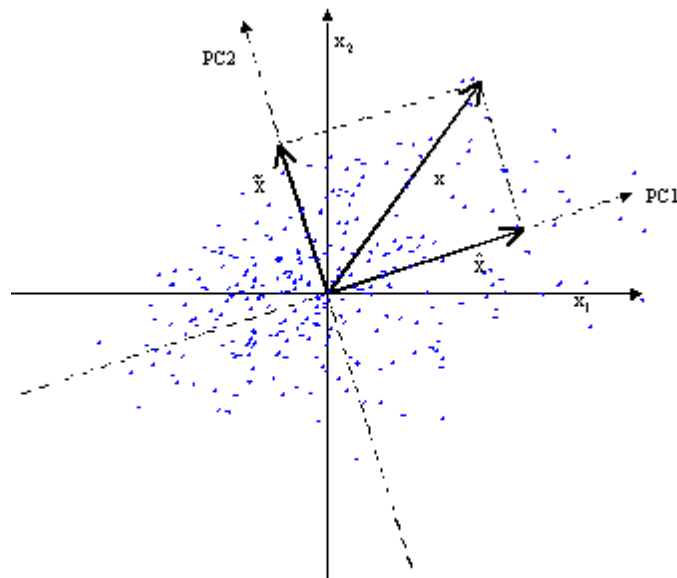
Les matrices  $C$  et  $\tilde{C}$  sont, en fait, les matrices de projections sur ces deux sous espaces. On a alors :

$\hat{X}$  et  $\tilde{X}$  sont respectivement les parties modélisée et résiduelle.

$$X = \hat{X} + \tilde{X} \quad (2.7)$$

Avec

$$\begin{aligned} \hat{X} &= \hat{C}X \text{ et } \tilde{X} = \tilde{C}X \\ \tilde{X}^T \hat{X} &= 0 \end{aligned} \quad (2.8)$$



**Fig. II-1** : représentation d'observations et des parties modélisée  $\hat{X}$  et résiduelle  $\tilde{X}$  de l'une d'entre elles  $X$  dans le cas de deux variables ( $m = 2$ )

Pour obtenir notre modèle, il nous reste à déterminer le nombre  $\ell$  de composantes principales conservées. Ce choix est une étape importante de la construction du modèle. En effet, si on retient trop de composantes, la dimension du processus sera peu réduite et le seul gain sera l'orthogonalité des nouvelles variables. De plus, le modèle risque de contenir des bruits de mesure. Si, au contraire, on ne conserve que trop peu de composantes, il y aura une perte d'information importante et on obtiendra une représentation incomplète du procédé. L'essentiel de la méthode de localisation des défauts repose sur une technique de reconstruction des variables qui vise à retrouver la valeur « normale » qui aurait dû être mesurée par un capteur défectueux. Si le  $i^{\text{ème}}$  capteur est défectueux à l'instant  $k$ , on a :

$$x_k = x_k^* + f\xi_i \quad (2.9)$$

où  $x_k^*$  est un vecteur de mesures « normal » et  $i$  la  $i^{\text{ème}}$  colonne de la matrice identité.  $f$  est l'amplitude du défaut.

La reconstruction du  $i^{\text{ème}}$  capteur a pour but d'estimer  $x_k^*$  en fonction du modèle et des mesures des autres capteurs, par un vecteur  $x_k^i$ . On mesure la fiabilité de cette technique par la variance de l'erreur de reconstruction :

$$\text{Var}(x_k^* - x_k^i) \quad (2.10)$$

Pour les données supposées sans défaut (qui servent à établir le modèle) on a  $f = 0$  et  $x = x^*$ . La variance de l'erreur de reconstruction est donc, dans ce cas :

$$\text{Var}(x_k^* - x_k^i) \quad (2.11)$$

Cette quantité dépend de  $\ell$  car  $x_k^i$  est établi à partir du modèle et est, par conséquent, différent selon le nombre de composantes principales conservées. On détermine alors le nombre de composantes principales du modèle en minimisant cette erreur pour l'ensemble des capteurs. On cherche donc  $\ell$  tel que :

$$J(\ell) = \sum_{i=1}^m \text{Var}(x_k - x_k^i) \quad (2.12)$$

soit minimale.

## II.3 Détection

### II.3.1 : Première méthode : test sur les dernières composantes principales

L'idée de base de cette méthode est que les dernières composantes principales sont, en général, les plus affectées par les valeurs aberrantes. En effet, celles-ci étant de variance faible, l'étendue de leurs valeurs est plus restreinte. Si une valeur aberrante altère l'une des variables, les composantes principales, en tant que combinaison linéaire des variables, en seront affectées aussi. Cet écart est, en général, plus visible sur les dernières composantes en raison de la faible étendue de leurs valeurs.

On va alors chercher à détecter les valeurs aberrantes en observant les dernières composantes. Pour Cela, effectuons un test sur les sommes des carrés des dernières composantes

$$I_{nk} = \sum_{j=m-i+1}^m t_{kj}^2 \quad \forall i; 1 \leq i \leq m - \ell \quad (2.13)$$

$t_{kj}$  étant la  $k^{\text{ième}}$  coordonnée de la  $j^{\text{ième}}$  composante principale.

Ce test consiste à comparer ces sommes à des seuils déterminés sur les données qui ont servi à établir le modèle. En fonctionnement normal, les valeurs des dernières composantes sont faibles, on restera donc en dessous du seuil. Au contraire, si l'un des capteurs semble comporter un défaut, les carrés des dernières composantes en seront affectés et les  $In_i$  dépasseront alors la limite fixée. On commence par calculer ces quantités en l'absence de défaut, c'est à dire sur les données d'apprentissage du modèle. On définit alors le seuil de détection  $\delta_i$  pour chacun des  $In_i$  comme le maximum atteint sur les données sans défaut.

Le schéma de la **Figure II.2** illustre le principe de contrôle et de détection de défaut

Le nouveau vecteur de mesures  $x_k = [x_{k1} \dots \dots x_{km}]^T$  (à l'instant ) est considéré comme un individu illustratif :

1. on calcule le nouveau vecteur des composantes principales  $t_k = [t_{k1} \dots \dots t_{km}]^T$  correspondant à l'aide de la formule suivante :

$$t_k = P^T x_k, i = 1 \quad (2.14)$$

2. on calcule  $In_{ki}$  , si  $In_{ki} < \delta_i$  , il n'y a pas de défaut, on pose alors  $i = i + 1$  (on s'arrête lorsque  $i = m - \ell$  ) et on retourne en 2.

Sinon, on soupçonne un défaut, il faut alors procéder à une étape de localisation

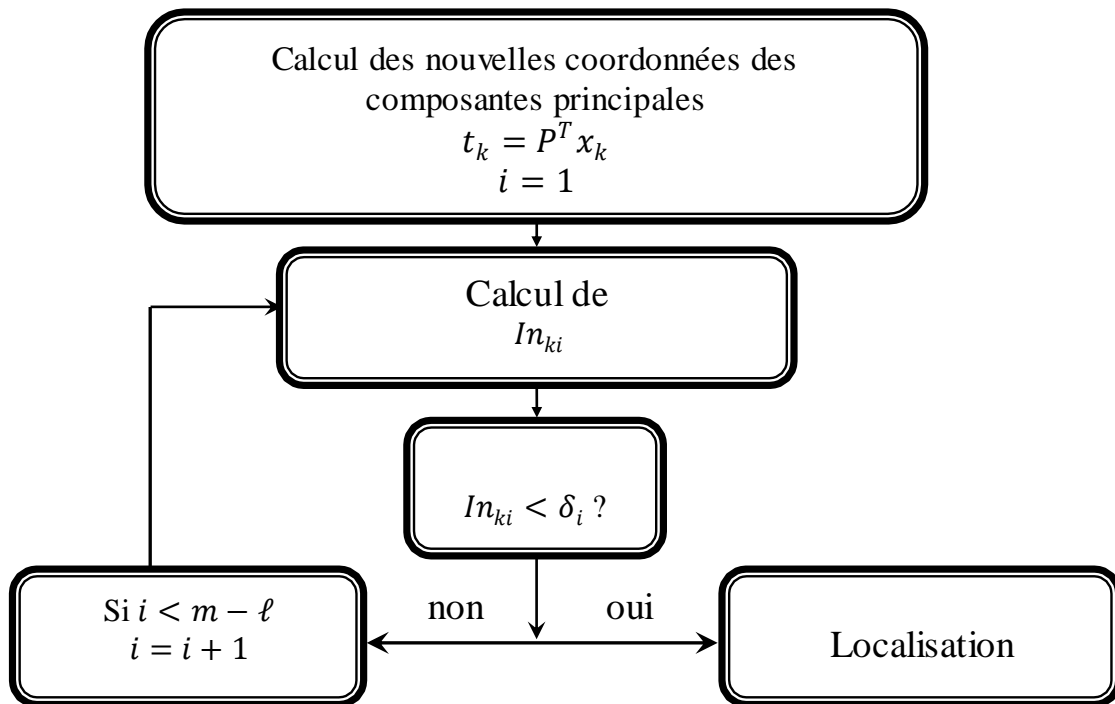
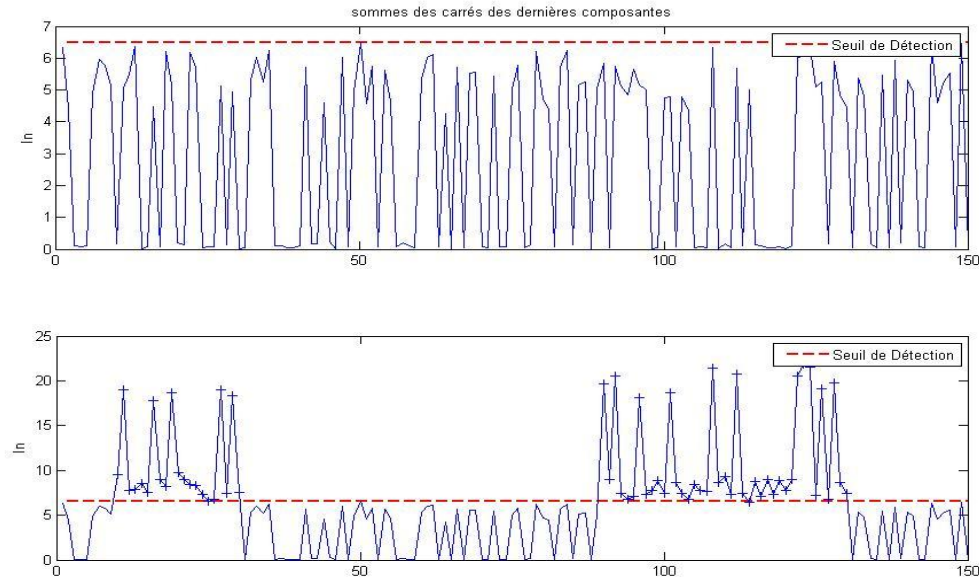


Figure. II-2 : méthode de détection

La figure II-3 représente le carré de la dernière composante principale  $In_1$  au cours du temps et le seuil  $\delta_1$ . Sur la première courbe, seules les données « normales » sont représentées. Les données « anormales » représenté par '+', a été ajoutée sur la seconde courbe. On détecte bien les anomalie au temps  $t \in [10,30]$  et  $[90,130]$ .

Exemple de données simulées



**Figure II.3 :**  $In_1$  sur les données sans défaut (1<sup>ère</sup> courbe) et (2<sup>ème</sup> courbe)  $In_1$  sur les données avec défaut

### II.3.2 Deuxième méthode : Test sur l'erreur de prédiction

L'erreur de prédiction (Square Prediction Error) caractérise la qualité de la représentation. Elle est donnée, à l'instant  $k$ , par :

$$SPE = \|\tilde{x}_k\| \quad (2.15)$$

Où  $\|\cdot\|$  désigne la norme euclidienne de  $\mathfrak{R}^m$ .

Comme lors du test sur la somme des carrés des dernières composantes, on définit un seuil de détection. Le processus est considéré normal à l'instant  $k$  si

$$SPE < \delta \quad (2.16)$$

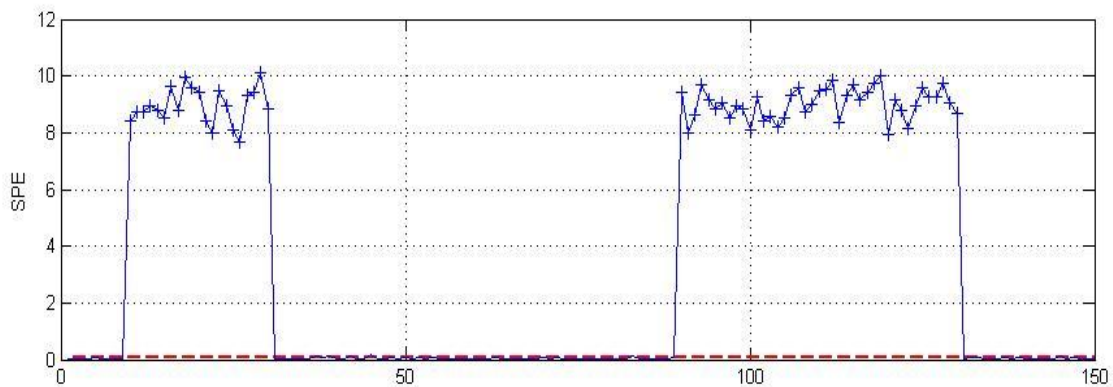
où  $\delta$  désigne une limite de confiance pour la quantité  $SPE$ .

Pour l'obtenir, on peut supposer que les résidus suivent une distribution particulière. Jackson [JAC 79] ont établi une expression de  $\delta$  dans le cas où  $x$  suit une loi normale

$$\delta = \theta_1 \left[ \frac{C_\alpha \sqrt{2\theta_2 h_0^2}}{\theta_1} + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2} \right]^{1/h_0} \quad (2.17)$$

Avec  $\theta_1 = \sum_{j=\ell+1}^m \lambda_j^i$  pour  $i = 1, 2, 3, \dots$ ,  $\lambda_j$  étant  $j^{\text{ème}}$  plus grande valeur propre de la matrice de corrélation,  $h_0 = 1 - \frac{2\theta_1\theta_3}{\theta_2^2}$  et  $C_\alpha$  est la limite de confiance à  $(1 - \alpha)$  dans le cas d'une distribution normale. Cette expression du seuil est fréquemment utilisée pour la détection de défauts.

Cette méthode permet de détecter le défaut sur les données simulées (figure II- 4)



**Figure II-4:** SPE avec anomalie ('+') à l'instant  $t \in [10,30]$  et  $[90,130]$ .

### II.3.3 Conditions de détectabilité

Nous allons énoncer ici des conditions de détectabilité valables pour les deux méthodes. Elles ne seront démontrées que dans le cas de la détection sur la quantité *SPE* mais s'étendent facilement à la détection sur la somme des carrés des dernières composantes.

Une observation présentant un défaut peut être vue comme une observation normale perturbée. Soit  $x$  l'observation courante et  $x^*$  l'observation normale dont elle provient. Si on suppose que le  $i^{\text{ème}}$  capteur est défaillant, on a :

$$x = x^* + f\xi_i \quad (2.18)$$

où  $\xi_i$  est la  $i^{\text{ème}}$  colonne de la matrice identité et  $f$  l'amplitude de la perturbation

En projetant cette relation sur RS, on obtient :

$$\tilde{x} = \tilde{x}^* + f\tilde{\xi}_i \quad (2.19)$$

Si  $\tilde{\xi}_i = 0$ .

on a :

$$\tilde{x} = \tilde{x}^* \quad (2.20)$$

donc  $SPE = SPE^*$ .

Par conséquent, il est impossible de détecter le défaut

On suppose désormais que  $\tilde{\xi}_i = 0$  Pour être sûr que le défaut soit détecté, il faut que

$$SPE = \|\tilde{x}\|^2 > \delta \quad (2.21)$$

Or d'après l'équation précédente on a :

$$\tilde{x} = \tilde{x}^* + \tilde{f}\tilde{\xi}_i^0 \quad (2.22)$$

Avec

$$\tilde{f} = f\|\tilde{\xi}_i\| \text{ et } \tilde{\xi}_i^0 = \tilde{\xi}_i/\|\tilde{\xi}_i\| \quad (2.23)$$

Par suite

$$|\tilde{f}| = \|\tilde{x} - \tilde{x}^*\| \text{ et } \tilde{\xi}_i^0 = \tilde{\xi}_i/\|\tilde{\xi}_i\| \quad (2.24)$$

Donc

$$|\tilde{f}| \leq \|\tilde{x}\| + \|\tilde{x}^*\| \quad (2.25)$$

Comme  $\|\tilde{x}^*\| = \sqrt{SPE^*} \leq \sqrt{\delta}$

$$|\tilde{f}| > 2\sqrt{\delta} \quad (2.26)$$

Cette contrainte garantie la détection du défaut, cependant, il est possible qu'un défaut soit détectable sans que celle-ci ne soit satisfaite.

Une observation est considérée « normale » si elle se situe à une distance inférieure à  $\sqrt{\delta}$  de l'espace des composantes principales PCS. On obtient ainsi une « zone de confiance » autour de PCS. Si on se place dans l'espace résiduel RS. On peut voir cette zone comme un disque centré en l'origine. Par définition,  $x^*$  se situe dans cette zone. Si  $\tilde{x}^*$  se situe sur le bord du disque, il faut que la distance qui le sépare de  $\tilde{x}$  soit supérieur au diamètre du disque pour être certain que ce défaut soit détecté.

### III. 3 Localisation

Cette étape permet de déterminer quel capteur est à incriminer en cas de défaut. Elle repose sur une technique de reconstruction des variables et suppose que chaque capteur puisse

être suspecté. La reconstruction du capteur en défaut effectuée en se base sur le modèle et les mesures des autres capteurs.

### III.3.1 Reconstruction des variables

Le but de la reconstruction est de retrouver un vecteur de mesures « normal » à partir de mesures défectueuses, et connaissant le modèle ACP. Une mesure anormale peut être vue comme une mesure normale perturbée [DUN 96, QIN 00 , HAR 03].

Soit  $x^*$  une observation sans défaut. On considère :

$$x = x^* + f\xi_i \quad (2.27)$$

Ceci revient à ajouter une perturbation de direction  $\xi_i \in \mathfrak{R}^m$  ( $\|\xi_i\| = 1$ ) et d'amplitude  $f$  à  $x^*$ .

Si cette perturbation correspond à un défaut sur le  $i^{\text{ème}}$  capteur ,  $\xi_i$  est la  $i^{\text{ème}}$  colonne de la matrice identité.

L'objectif est de corriger au mieux l'effet de cette perturbation, c'est-à-dire d'estimer  $x^*$  dans la direction  $\xi_i$ . En d'autre termes on recherche  $f_i$  tel que :

$$x^i = x - f_i \xi_i \quad (2.28)$$

Soit le plus cohérent possible avec le modèle ACP, c'est-à-dire le plus proche possible de sa projection sur l'espace PCS.

On recherche donc  $f_i$  tel que :

$$\begin{aligned} f_i &= \arg \min_{f_i} \|\tilde{x}^i\|^2 \\ &= \arg \min_{f_i} \|\tilde{x} - f_i \xi_i\|^2 \end{aligned} \quad (2.29)$$

Avec  $\tilde{\xi}_i = \tilde{C} \xi_i$ .

Or  $g(f_i) = \|\tilde{x} - f_i \tilde{\xi}_i\|^2$  est convexe et dérivable en  $f_i$  donc l'annulation de la dérivée est une condition de minimalité. Le  $f_i$  donc cherché vérifie alors :

$$g'(f_i) = 2f_i \|\tilde{\xi}_i\|^2 - 2\tilde{x}^T \tilde{\xi}_i = 0 \quad (2.30)$$

Où

$$f_i = \frac{\tilde{\xi}_i^T \tilde{x}}{\tilde{\xi}_i^T \tilde{\xi}_i} = \frac{\xi_i^T x}{\xi_i^T \xi_i} \quad (2.31)$$

La meilleure approximation possible de  $x^*$  est par conséquent :

$$\begin{aligned}
 x^i &= x - \frac{\tilde{\xi}_i^T x}{\tilde{\xi}_i^T \tilde{\xi}_i} \tilde{\xi}_i \\
 &= x - \frac{\tilde{\xi}_i^{0T} x}{\|\tilde{\xi}_i\|} \tilde{\xi}_i
 \end{aligned}
 \tag{2.32}$$

Avec  $\tilde{\xi}_i^0 = \frac{\tilde{\xi}_i}{\|\tilde{\xi}_i\|}$ .

Pour pouvoir reconstruire la variable  $i$  il faut que  $f_i$  existe. L'existence de  $f_i$  assurée dans le cas où  $\tilde{\xi}_i$  est non nul. Dans le cas contraire le défaut n'est pas détecté.

La figure II.6 illustre la reconstruction de la variable  $x_1$  dans le cas de l'exemple des variables simulées. Le vecteur  $x^1$  est la somme des vecteurs  $x$  et  $-f_1 \tilde{\xi}_1$ . On commence par déterminer  $\tilde{f}_1 = \tilde{\xi}_1^{0T} x$  en projetant  $x$  sur la droite dirigée par  $\tilde{\xi}_1^0$ , puisque  $\|\tilde{\xi}_1^0\| = 1$ . On cherche maintenant  $f_1 = \tilde{f}_1 / \|\tilde{\xi}_1\|$ . On a :

$$\tilde{\xi}_1^T \tilde{\xi}_1^0 = \frac{\tilde{\xi}_1^T C \tilde{\xi}_1}{\|\tilde{\xi}_1\|} = \|\tilde{\xi}_1\|
 \tag{2.33}$$

Et

$$\tilde{\xi}_1^T \tilde{\xi}_1^0 = \|\tilde{\xi}_1\| \|\tilde{\xi}_1^0\| \cos(\tilde{\xi}_1, \tilde{\xi}_1^0) = \cos(\alpha)
 \tag{2.34}$$

Donc

$$\|\tilde{\xi}_1\| = \cos(\alpha) \text{ et } f_1 = \tilde{f}_1 / \cos(\alpha)
 \tag{2.35}$$

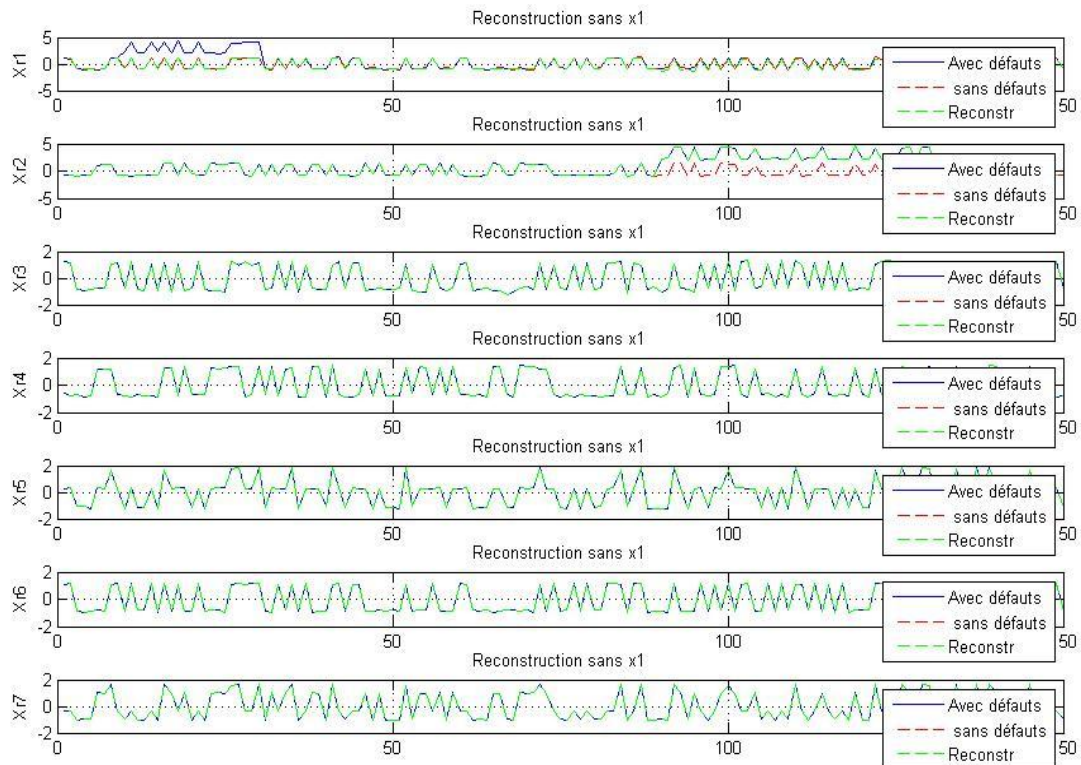
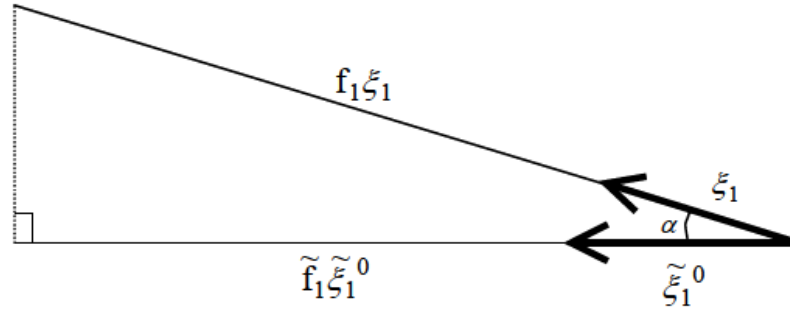


Figure II-6 Reconstruction des variables sans  $x_1$

On obtient donc le vecteur  $f_1 \xi_1$  de la manière suivante :



**Figure II.7:** détermination de  $f_i \xi_i$

Dans le cas d'un défaut sur le  $i^{\text{ème}}$  capteur.  $\xi_i$  est la  $i^{\text{ème}}$  colonne de la matrice identité. Ainsi seule la  $i^{\text{ème}}$  coordonnée de  $x$  est modifiée. On a alors :

$$x^i = [x_{-i}^T \quad z_i \quad x_{+i}^T] \quad (2.36)$$

$$\text{Avec } x_{-i} = [x_1 \quad \dots \quad x_{i-1}] \text{ et } x_{+i} = [x_{i+1} \quad \dots \quad x_m] \quad (2.37)$$

Par conséquent :

$$z_i = \xi_i^T x^i \quad (2.38)$$

$$\begin{aligned} &= x_i - \frac{\xi_i^T x}{\xi_i^T \xi_i} \\ &= \frac{\xi_i^T \xi_i x_i - \xi_i^T x}{\xi_i^T \xi_i} \\ &= \frac{\xi_i^T \xi_i x_i - \xi_i^T C x}{\xi_i^T \xi_i} \end{aligned} \quad (2.39)$$

Or  $\xi_i^T \xi_i = \xi_i^T C C \xi_i = \xi_i^T C \xi_i$  car  $C$  est une matrice de projection. De plus,  $C \xi_i$  est la  $i^{\text{ème}}$  colonne de  $C$  et  $\xi_i^T v$  la  $i^{\text{ème}}$  ligne de  $v$ . Ainsi,  $\xi_i^T \xi_i$  est le  $i^{\text{ème}}$  terme diagonal de  $C$  c'est à dire  $(1 - c_{ii})$ . D'où :

$$z_i = \frac{(1 - c_{ii}) \xi_i^T x - \xi_i^T C x}{(1 - c_{ii})} \quad (2.40)$$

$$z_i = \frac{((1 - c_{ii}) \xi_i^T - \xi_i^T C) x}{(1 - c_{ii})} \quad (2.41)$$

Avec  $\xi_i^T C = [-c_{-i} \quad (1 - c_{ii}) \quad +c_{+i}]$  (c'est le  $i^{\text{ème}}$  ligne de  $C$ ).

Finalement

$$z_i = \frac{[c_{-i}^T \quad 0 \quad c_{+i}^T] x}{(1 - c_{ii})}, \quad (c_{ii} \neq 1) \quad (2.42)$$

On remarque que cette formule n'utilise que les « paramètres » du modèle et les mesures des autres capteurs. La valeur mesurée par le capteur suspecté n'intervient pas, ainsi, on ne prendra pas en compte une mesure supposée incorrecte pour estimer la valeur de substitution.

## II.3.2 Méthodes de localisation des défauts

### II.3.2.1 Première méthode : test sur les dernières composantes.

Si la phase de détection a mis en évidence un problème à l'instant  $k$ , on procède de la manière suivante :

Supposons, par exemple, que  $In_{k1} > \delta_1$ . On commence par reconstruire la première variable, puis, on calcule la valeur de la dernière composante  $t_{km}(1)$  correspondant au nouveau vecteur d'observations  $x_k^1$  obtenu à l'aide de la formule (2.14). On calcule alors la nouvelle valeur  $In_{k1}(1)$  associée : si  $In_{k1}(1) < \delta_1$ , le vecteur  $x_k^1$  est conforme au modèle, la première variable est alors incriminée. Sinon, elle est « innocentée ». On effectue les mêmes calculs pour toutes les variables.

Reprenons exemple 1 chapitre IV. L'étape précédente a permis de détecter le défaut à  $t \in [10,30]$  et  $t \in [90,130]$ . On procède maintenant à l'étape de localisation (sur les  $m - \ell$  dernières composantes). On reconstruit la première variable,  $x_1$ , puis on calcule les sommes des carrés des dernière composante correspondant  $In(1)$ . On fait la même chose pour les variables  $x_2(In(2))$ ,  $x_3(In(3))$ ,  $x_4(In(4))$ ,  $x_5(In(5))$ ,  $x_6(In(6))$  et  $x_7(In(7))$ . On peut voir sur les courbes ci-dessous (Fig. II-8) que les  $In(i)$ ,  $i = 1, \dots, 7$  passe en dessus du seuil  $\delta_1$  de différents manières avec  $In(3)$ ,  $In(4)$ ,  $In(5)$ ,  $In(6)$  et  $In(7)$  ont la même courbe qui traduit l'admission des données des variables  $x_3, x_4, x_5, x_6$  et  $x_7$ . D'autre part pour  $In(1)$  et  $In(2)$ , on remarque bien l'absence de dépassement pour que  $t \in [10,30]$  et  $t \in [90,130]$  respectivement, qui interprète une localisation des données anormales pour  $x_1(t \in [10,30])$  et  $x_2(t \in [90,130])$ . Cette méthode permet donc bien de localiser le défaut.

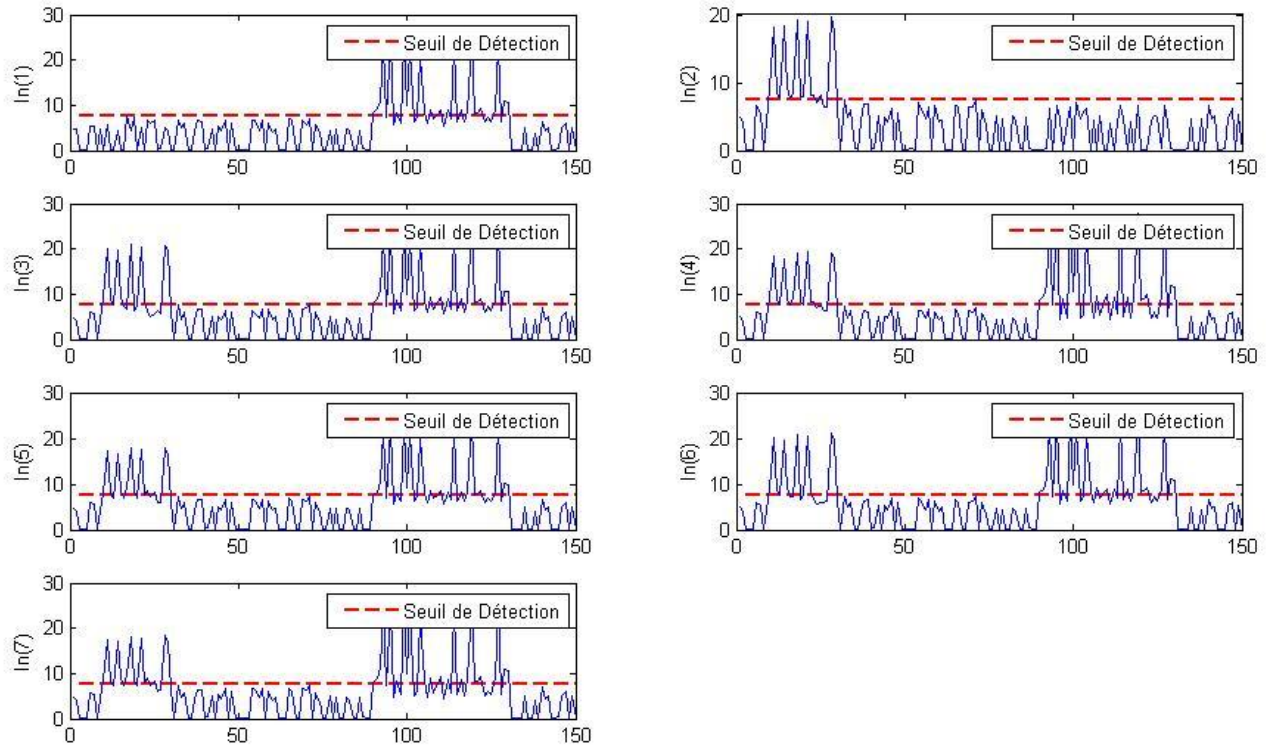


Fig. II-8 : carré de la dernière composante principale

$ln(1)$  : après reconstruction de  $x_1$ ,  $ln(2)$  : après reconstruction de  $x_2$ ,

$ln(3)$  : après reconstruction de  $x_3$ ,  $ln(4)$  : après reconstruction de  $x_4$ ,

$ln(5)$  : après reconstruction de  $x_5$ ,  $ln(6)$  : après reconstruction de  $x_6$ ,

$ln(7)$  : après reconstruction de  $x_7$ .

### II.3.2.2 Deuxième méthode : test sur la quantité SPE.

Afin de localiser le capteur défaillant on introduit une nouvelle quantité  $\eta_{ki}$  : c'est un indice de validité du  $i^{ème}$  capteur à l'instant  $k$ . Il est défini par :

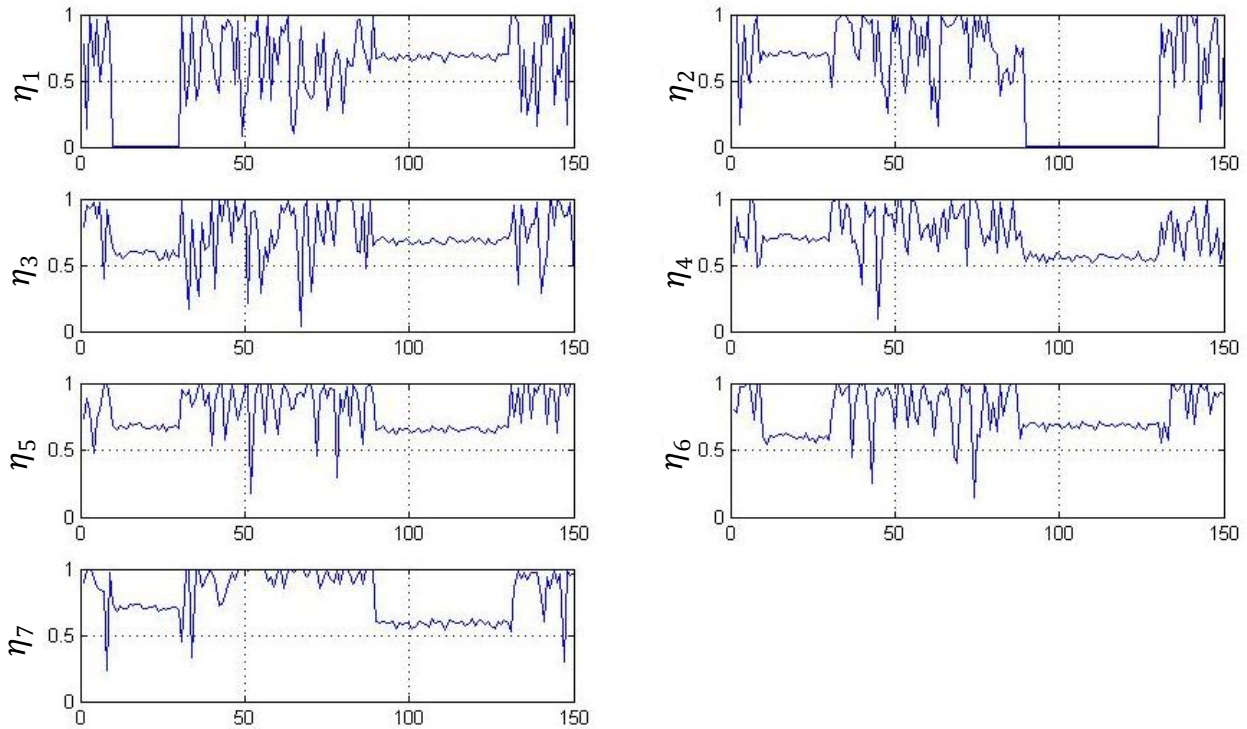
$$\eta_{ki} = SPE_k^i / SPE_k \quad (2.43)$$

où  $SPE_k$  et  $SPE_k^i$  sont, respectivement, le  $SPE$  calculé avant et après reconstruction du  $i^{ème}$  capteur à l'instant  $k$ .

Lorsque le capteur reconstruit ( $i^{ème}$ ) n'est pas le capteur défectueux, la reconstruction modifiera peu la valeur du capteur et donc le  $SPE$ . Par conséquent,  $\eta_{ki}$  sera proche de 1. Au contraire, lorsque le capteur défectueux est reconstruit,  $\eta_{ki}$  est proche de 0. En effet, par définition de la reconstruction, la partie non modélisée du nouveau vecteur est minimale donc

la norme  $SPE_k^i$  du résidu associé sera faible. De plus, puisqu'un défaut a été suspecté à cet instant,  $SPE_k^i$  dépasse le seuil de détection ce qui rend  $\eta_{ki}$  encore plus petit.

Dans le cas de notre exemple, la figure suivante illustre les changements de l'indice de validité :



**Figure II.9 :** Indice de validité des variables .

On retrouve bien le fait que les défauts affectent les variables  $x_1$  et  $x_2$ . Toutefois, ce critère semble assez « flou » à cause de la notion de proximité.

Puisque  $SPE_k = In_{k,m-\ell}$ , on peut aussi utiliser le même type de test que celui effectué sur les  $In_i$ . Après reconstruction de chacune des variables, on compare la quantité  $SPE_k^i$  au seuil établi précédemment. Si  $SPE_k^i > \delta$  la  $i^{\text{ème}}$  variable est innocentée, sinon elle est suspectée.

En appliquant cette deuxième méthode à l'exemple 1 chapitre IV, nous avons bien localisé le défaut sur la variable  $x$ .

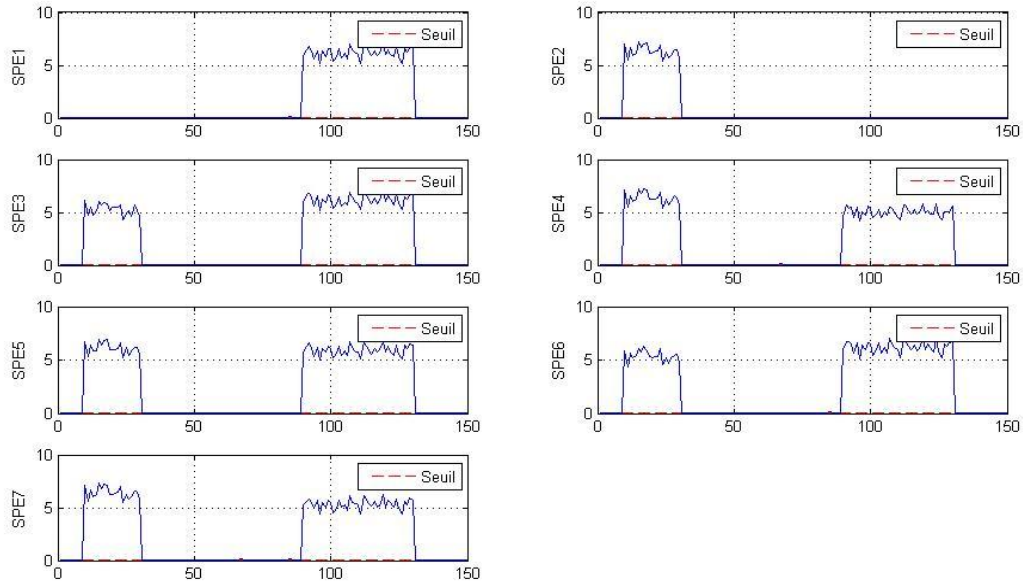


Figure II.10: SPE après reconstruction de chacune des variables

### II.3.2.3. La contribution classique au critère SPE

Avec l'ACP, le  $SPE$  est utilisées pour la surveillance d'un processus. Par conséquent, chaque variable aura une contribution.

Nous déduisons que le critère  $SPE$  n'est autre que l'erreur quadratique de prédiction qui est égale à la somme des carrés des résidus [BAL09, MNA 08] :

$$SPE = \|\tilde{X}\|^2 = \tilde{X}^T(k)\tilde{X}(k) = \sum_{j=1}^m \tilde{X}_j^2 \quad (4.19)$$

où  $\tilde{X}_j$  est le résidu de la  $j^{ième}$  variable. Cette approche propose que la contribution d'une variable au  $SPE$  s'exprime par le carré de son propre résidu [KOU 96, KOU 05].

avec

$$Cont_j^{SPE} = \tilde{X}_j^2 \quad (4.20)$$

## **II.4.Conclusion**

Les méthodes statistiques basées sur l'analyse en composantes principales (ACP) sont très intéressantes pour la mise en évidence des corrélations significatives entre les différentes variables sans formuler de manière explicite le modèle du système. L'ACP est utilisée pour modéliser le comportement d'ensemble du processus en fonctionnement normal puis les défauts sont détectés en comparant le comportement observé à celui donné par le modèle. Ce procédé indique s'il existe un défaut au niveau global, c'est à dire pour l'ensemble des capteurs, sans information sur l'origine de ce défaut.

Lorsque le nombre de dimension de la matrice d'observation augmente, l'utilisation de l'ACP devient lourde du point de vue temps de calcul intervenant dans les procédures de diagonalisation de la matrice d'observation des données. Pour éviter la diagonalisation de la matrice deux réseaux sont utilisés pour faire de l'ACP : Les réseaux de Soja et de Sanjer. Ces réseaux s'inspirent de la modélisation des données par les modèles connexionnistes ou réseaux de neurones, et permettent de réaliser l'analyse en S composantes principales. Nous abordons dans la suite de notre travail, les méthodes de détection et la localisation des défauts utilisant le modèle connexionniste : le modèle des cartes topologiques de Kohonen.

# **CHAPITRE III : DETECTION ET LOCALISATION DES DEFAUTS PAR CARTE TOPOLOGIQUE DE KOHENEN (SOM)**

## **III.1. Principe**

L'idée fondamentale de notre travail est de détecter et localiser les défauts à travers les mêmes démarches que l'analyse en composant principale (ACP), mais en se base sur une architecture basé sur les réseaux de neurones. L'architecture de base utilisé est la carte auto-organisatrice de Kohonen (SOM).

La détection et la localisation de défaut par la carte de Kohonen passe par les étapes suivantes :

- Une étape de modélisation des données (fondée sur une phase d'apprentissage)
- Une étape détection et localisation proprement dite.

## **III.2.Phase d'apprentissage (Modélisation) :**

### **III.2.1.Principe**

L'approche retenue est fondée sur :

1. La mise en correspondance entre les vecteurs d'entrée et les vecteur de références (prototypes ou classes d'objets) en se basant sur le choix d'une distance appropriée entre classes d'objet.
2. L'utilisation de l'algorithme de Kohonen pour le choix des prototype en phase d'apprentissage (voir Annexe carte SOM).

### **III.2.2.Modélisation par carte de Kohonen**

Les cartes auto-organisatrices présentent un exemple des méthodes à apprentissage non supervisé. Elles analysent les données et extraient l'information à partir de ces dernières en utilisant un algorithme de classification sophistiqué [KOH 91,VER 90]. Cette technique utilise uniquement des données ou des attributs dérivés de ces données, et n'exige pas une estimation a priori sur les données d'entrée.

Les cartes auto-organisatrices de Kohonen devaient pouvoir concerner de très grands ensembles de données, pour lesquelles les observations traitées pouvaient atteindre de grandes dimensions. Le modèle issu par cartes topologiques proposées par Kohonen cherche, par apprentissage à partir des données, à partitionner l'ensemble des observations disponibles en groupement similaires. Les groupements proposés possèdent la particularité caractéristique

d'avoir une structure de voisinage qui peut être matérialisée à l'aide d'un espace discret que l'on appelle carte topologique.

### III.2.3. Architecture

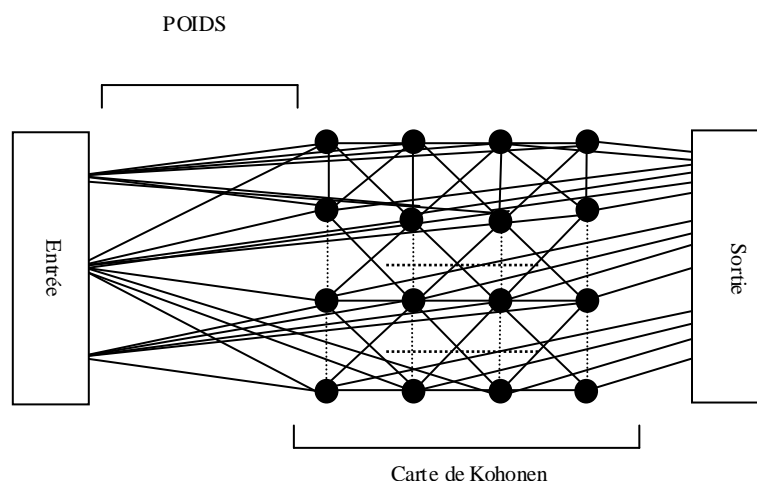
La carte est constituée de deux couches [KOH 81, MAR 90, VIL97]:

La couche d'entrée sert à la présentation des observations (données) à classer. Cette couche contient alors exactement  $N$  neurones ( $N$  étant la dimension de l'espace des observations)

La couche d'adaptation est formée du treillis des neurones qui forme la carte. La structure du réseau employé peut être soit fixée a priori, soit évoluer lors de l'apprentissage.

La couche de sortie en compétition où les neurones sont reliés en connexion complète avec les neurones de la couche d'entrée, chaque neurone possède  $N$  connexion (poids).

Le schéma de la carte est illustré par la figure suivante



**Figure III.1.** Schéma d'une carte de Kohonen connectée à un stimulus. Les connexions sont complètes

### III.2.4. Apprentissage par la carte

L'algorithme d'apprentissage de la carte topologique de Kohonen comprend principalement les phases suivantes [RIT 92] :

#### Etape 1 :

- **Phase d'Initialisation :** initialisation des données, choix de la structure de la carte, initialisation des poids de la carte.

**Etape 2 :**

- **Phase de compétition :** Sélection d'un individu, Détermination du neurone vainqueur. en prenant en considération le voisinage de ce neurone.

**Etape 3 :**

- **Phase d'adaptation des poids :** Modification des poids des prototypes ou phase de mise à jour.

**Etape 4 :**

- **Phase de poursuite de la phase de compétition :**

Retour à l'étape 2 si le changement des poids demeure non négligeable.

Les détails de la phase d'apprentissage sont expliqués en Annexe B.

**III.3.Détection et Localisation des défauts :****III.3.1.Principe**

Le principe de détection et localisation des défauts par carte de Kohonen repose sur les principales étapes suivantes [MIN 04, FAD10]:

**Etape 1 : Elaboration du modèle en fonctionnement normal** (partie traitée précédemment).

**Etape 2 : Détection et Localisation des défauts.** Dans cette étape nous commençons par :

- 1- Présenter un nouveau échantillon,
- 2- De faire la classification de cet échantillon par carte de Kohonen (détermination du neurone gagnant).
- 3- Elaboration d'un critère de détection qui prend en considération
  - a) Le choix d'un seuil  $\alpha$  et de la fonction d'activation.
  - b) Test effectué sur le calcul de la distance entre l'échantillon présenté et le neurone gagnant, en fixant un seuil  $s_d$

**Etape 3 :** décision.

Nous explicitons les détails de cet algorithme dans ce qui suit :

1. La classification du nouveaux échantillon présenté à la carte, consiste à rechercher l'indice du neurone gagnant appelé BMU (Best Matching Unit), par la relation suivante :

$$j(x_i) = \operatorname{argmin}_j \{\|x_i - w_j\|\} \quad (3.1)$$

où  $x_i$  est la nouvelle donnée et  $j$  est l'indice de classe gagnante dans modèle en bon fonctionnement.

2. Choix du seuil  $\alpha$ . Le choix de ce seuil est fixé par la méthode N.VALENTIN [MCL 88, ZRE92, VAL00 ]. Celui-ci, suppose disposer au départ d'un ensemble de vecteurs  $x$  de l'ensemble d'apprentissage, et considère que 1% de ces données sont invalides ou aberrantes. Il calcule l'activation  $K_i(x)$  du plus proche prototype pour ces  $n$  vecteurs. Par suite, il suppose que 1% des données de l'ensemble d'apprentissage ont une activation trop faible pour être considérées comme valides. Il classe les données par ordre croissant d'activation. Le seuil  $\alpha$  choisis correspond à l'activation  $K_i(x)$  de rang  $n/100$ .

Le calcul de la fonction d'activation est donné par l'expression suivante [VAL00 ] :

$$K_i(x) = \exp \left[ -\frac{\|x - w_i\|^2}{2\sigma_i^2} \right]$$

Où  $\sigma_i^2$  est un paramètre qui définit la région d'influence du prototype  $i$ .  $\sigma_i^2$  peut être défini comme la moyenne empirique des variances des  $n$  vecteurs d'entrées activant le prototype  $i$ . Plus  $\sigma_i^2$  est grand, plus la zone d'influence de  $w_i$  est grande et donc, plus l'activation  $K_i(x)$  est proche de 1.

3. Dans notre approche le seuil de détection  $s_d(i)$  définit la région d'influence du prototype  $i$ . Le seuil  $s_d(i)$  peut être défini par la distance euclidienne maximale calculée sur l'ensemble des échantillons d'une classe  $i$  exprimée par rapport à son centre de gravité . Il est défini par [FAD10] :

$$s_d(i) = \max_i \{\|x - w_i\|\} \quad (3.3)$$

### III.3.2. Décision :

Une donnée est valide si :

- sa fonction d'activation est supérieure au seuil  $\alpha$ , et si la distance euclidienne  $\|x - w_i\|$  est inférieure au seuil  $s_d(i)$ . Sinon il y a défaut.

Remarque :

Les échantillons de chaque classe sont supposés suivre une distributions gaussienne de moyenne  $w_i$  et de variance  $\sigma_i^2 I$  ( $I$  représente la matrice identité). L'activation du plus proche

prototype  $i$  par les vecteurs d'entrée  $x$  peut être considérée comme une approximation de la densité de probabilité conditionnelle de la classe  $i$ , et le seuil de détection  $s_d(i)$  voisin de  $4\sigma_i$  pour avoir 95% d'informations.

### III.4.Reconstruction des données

#### III.4.1.Méthodologie

Pour l'estimation ou la reconstruction des données en défauts, nous avons envisagé deux méthodes.

Dans un premier temps, l'estimation d'une observation consiste à utiliser une méthode simple qui estime les données par la valeur de la composante correspondante du prototype gagnant  $w_j$  :

$$\hat{x} = w_j \quad (3.4)$$

Cette méthode est très sensible au changement de prototype entre deux vecteurs successifs. Pour remédier à ce problème, nous avons envisagé une autre méthode, assez similaire, qui prend en compte l'influence des  $k$  plus proches prototypes. Chaque composante anormale est estimée par une combinaison des composantes correspondantes des  $k$  plus proches prototypes [VAL00 , PET98] :

$$\hat{x} = \frac{\sum_{j=1}^k K_j(x) w_{j,\hat{x}}}{\sum_{j=1}^k K_j(x)} \quad (3.5)$$

Ainsi, le vecteur de données  $x$  peut être exprimé sous la forme :

$$x = \hat{x} + e \quad (3.6)$$

où  $e$  est l'erreur d'estimation. :

$$e = x - \hat{x}$$

### **III.5. Conclusion**

Nous avons proposé dans ce chapitre un système basé sur l'application des cartes topologiques de Kohonen pour la détection et la localisation des défauts ainsi qu'à la reconstruction des données. La méthodologie adoptée est paramétrique et permet de contourner certaines difficultés rencontrées du point de vue temps de calcul et performances. Nous examinons dans la suite de ce travail, les résultats expérimentaux qui en découlent.

## CHAPITRE IV : RESULTATS EXPERIMENTAUX

Dans cette partie nous allons donner quelques exemples sur des données simulées pour mettre en œuvre pratique les approches étudiées précédemment concernant la détection et la localisation des défauts par de carte Kohonon. Les résultats obtenus par ACP ont été présentés (voir chapitre II ).

### IV.1. Les données

$$\begin{aligned}x_1(t) &= u_1(t) + e_1(t) \\x_2(t) &= u_2(t) + e_2(t) \\x_3(t) &= u_1(t) + e_3(t) \\x_4(t) &= u_2(t) + e_4(t) \\x_5(t) &= u_1(t) + u_2(t) + e_5(t) \\x_6(t) &= 2u_1(t) + e_6(t) \\x_7(t) &= u_1(t) + 3u_2(t) + e_7(t)\end{aligned}\tag{4.1}$$

Avec  $u_1$  et  $u_2 \in [-1,1]$  sont des variables aléatoires uniformes et  $e_i$ , un bruit aléatoire uniformément distribué dans l'intervalle  $[-0.1,0.1]$ .  $i = 1, \dots, 7$ , et des défauts d'amplitude égale à 3 définis de la façon suivante :

- Défauts sur  $x_1$  pour  $t \in [10,30]$ .
- Défauts sur  $x_2$  pour  $t \in [60,100]$ .

La figure suivante (IV.1) illustre la représentation graphique de ces signaux

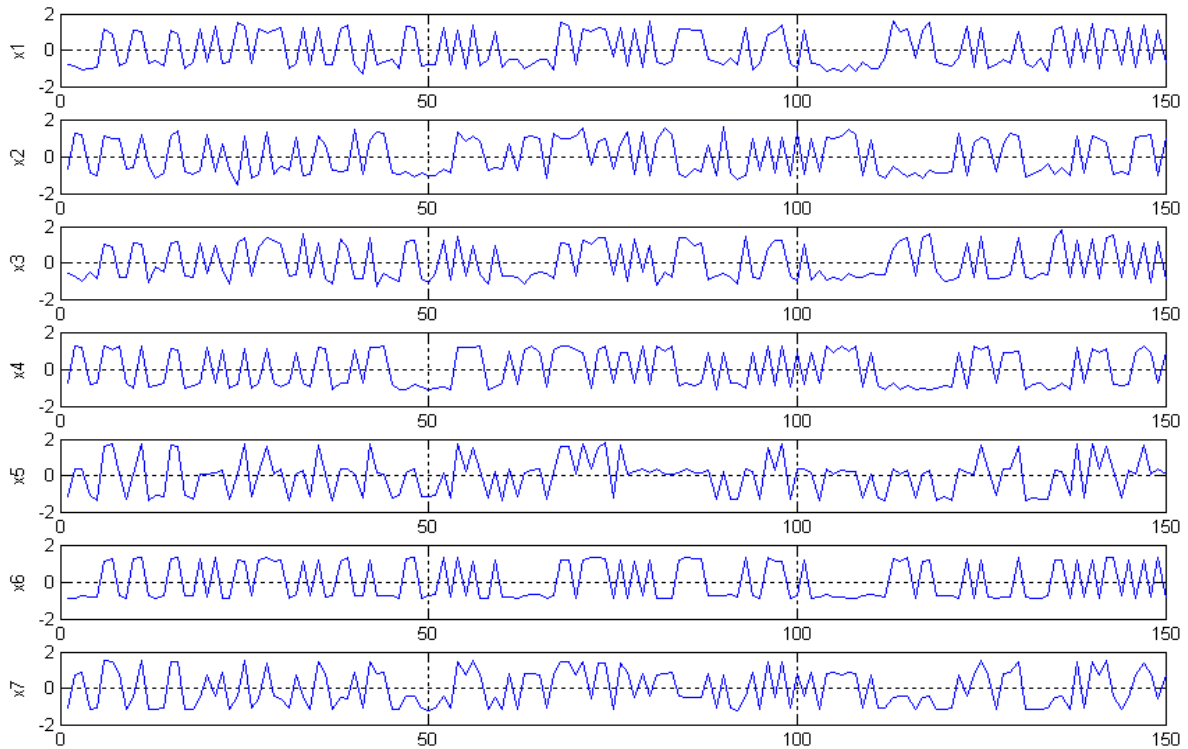


Figure IV.1 : signaux retenus

## IV.2.Modélisation :

Avant d'aborder la phase de modélisation proprement dite, nous examinons d'abord le choix adopté pour dimensionner la carte [KOH 94].

### IV.2.1.Choix des dimensions de la carte

Le choix des dimensions de la carte repose sur l'estimation de l'erreur de discrimination qui est définie par [BRA 89, MAR 90, KOH 90-1] :

$$q(x) = d^v(x, w) \quad (4.2)$$

Celle ci définit une distance entre  $x$  (réalisation de  $X$ ) et le vecteur prototype de la BMU. L'ensemble des vecteurs prototypes minimise l'erreur de discrétisation totale [FRI93]. lorsque l'espérance  $Q$  des  $q(x)$  est minimale pour une densité de probabilité donnée. Ce qui nous donne pour la fonction de densité de probabilité  $f_X(x)$  :

$$Q = \int_V q(x) f_X(x) dx \quad (4.3)$$

Cette erreur est estimée par l'expression suivante :

$$\hat{Q} = \frac{1}{\text{card}(E_t)} \sum_{x \in E_t} q(x) \quad (4.4)$$

Avec  $E_t$  = ensemble de test  $\subset$  dans  $V$  (espace d'entrée).

Nous avons retenu pour la suite une carte constituée de  $k_1 \times k_2$  cellules, avec un voisinage carrée. La sélection de la taille de la carte s'effectue à l'aide de l'erreur de discrétisation. Cette dernière définit l'aptitude à reconstruire l'ensemble d'apprentissage. Comme les cartes de Kohonen nous servent principalement pour la reconstruction de données, nous avons décidé de sélectionner la carte en minimisant ce critère afin d'avoir les meilleures performances possibles en reconstruction.

Le Tableau 4-1 suivant montre la valeur de l'erreur de discrétisation obtenue pour différentes tailles de carte, à la fin de l'apprentissage.

**Tableau 4-1: Valeurs de l'erreur de discrétisation pour la carte sélectionnée à la fin de l'apprentissage.**

Nombre d'itération	Nombre de Cellules	Erreur de discrétisation
300	$2 \times 2$	1.4432
300	$2 \times 3$	0.1476
300	$2 \times 4$	0.1354
300	$3 \times 3$	0.1267
300	$3 \times 4$	0.0926
300	$4 \times 4$	0.0864

Les cartes des tailles  $4 \times 4$  et  $3 \times 4$  donnent des erreurs de discrétisation légèrement meilleure que les autres architectures, mais celle-ci comporte beaucoup de prototypes qui ne sont activés par aucun vecteur de l'ensemble d'apprentissage. Nous avons donc sélectionné une carte de taille  $3 \times 3$ .

#### IV.2.2. Phase d'apprentissage

Nous appliquons en ce sens l'algorithme d'apprentissage de la carte pour élaborer la mise en correspondance entre les signaux d'entrée et les neurones de la carte. Nous obtenons les résultats suivants (Figure IV.2):

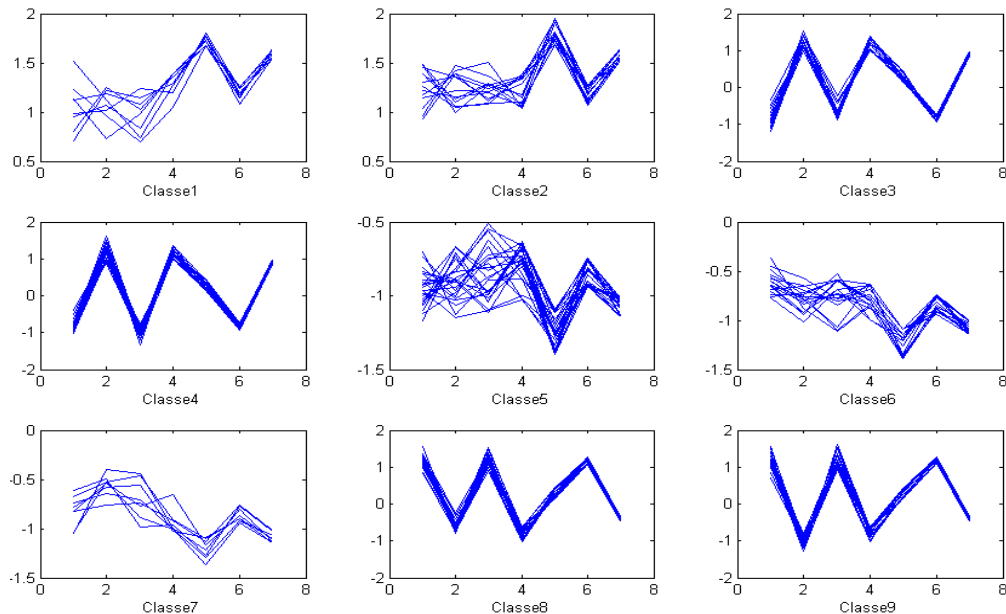
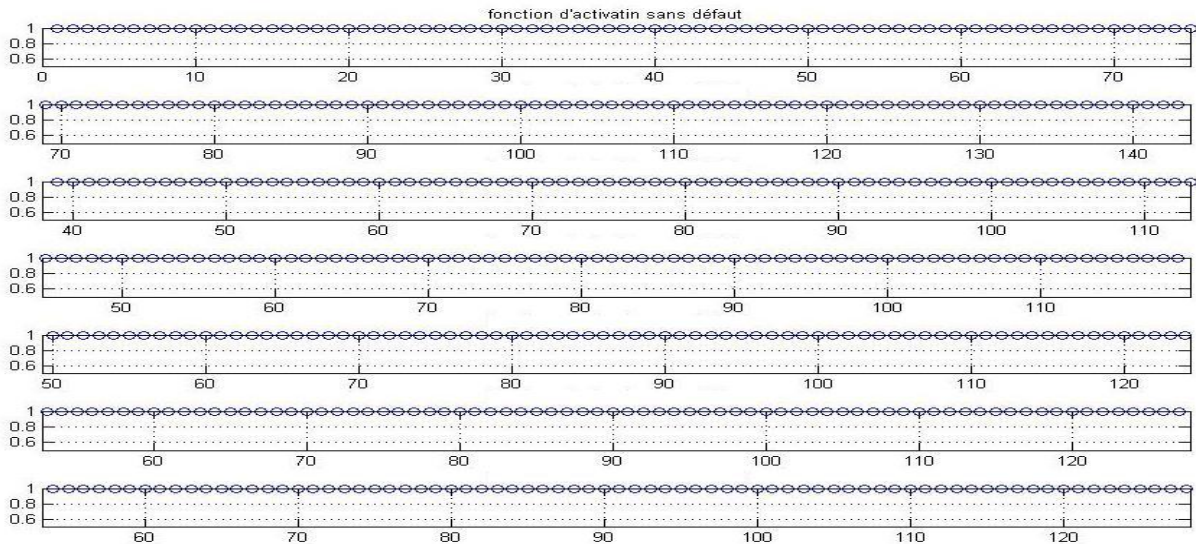


Figure IV.2 : résultats de la phase d'apprentissage.

*Chaque neurone est associé à une classe et un ensemble d'échantillons.  
Taux d'apprentissage initial  $\eta_0=0.9$ , Taux d'apprentissage final  $\eta_f=0.2$ ,  
Largeur de voisinage initiale  $\sigma_0=3$ , Largeur de voisinage finale  $\sigma_f=0.01$ .*

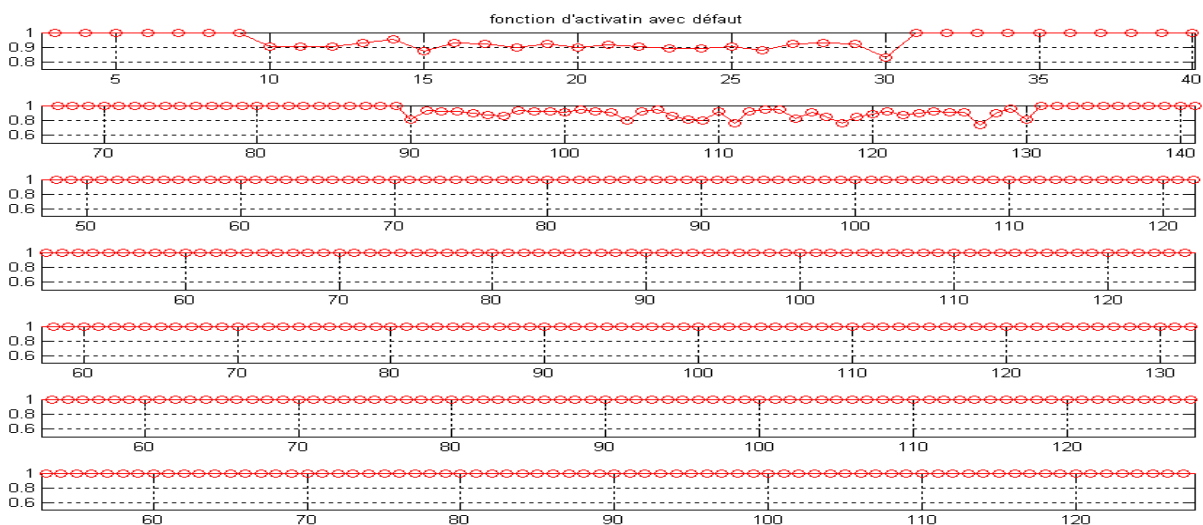
### IV.3. Détection et localisation des défauts par la carte

Avant d'entamer cette phase, il y a lieu d'abord de représenter la fonction d'activation dans le cas de fonctionnement normal est défectueux. Ce ci est illustré par le schéma des figures suivantes.(figure IV.3.a, figure IV.3.b)



**Figure IV.3.a :** fonction d'activation des données  
(cas d'un fonctionnement normal appliqué aux signaux de l'exemple 1).

La figure IV.3.a montre que la fonction d'activation de la BMU dans le cas de bon fonctionnement, tend vers 1. Ce qui signifie que la zone d'influence des prototypes est grande. Contrairement, dans le cas de fonctionnement anormal (figure IV.3.b), celle-ci tend vers 0.



**Figure IV.3.b:** fonction d'activation des données  
(des défauts affectant  $x_1$  à l'instant (10-30) et  
 $x_2$  à l'instant (90-130) appliqué aux signaux de l'exemple 1)

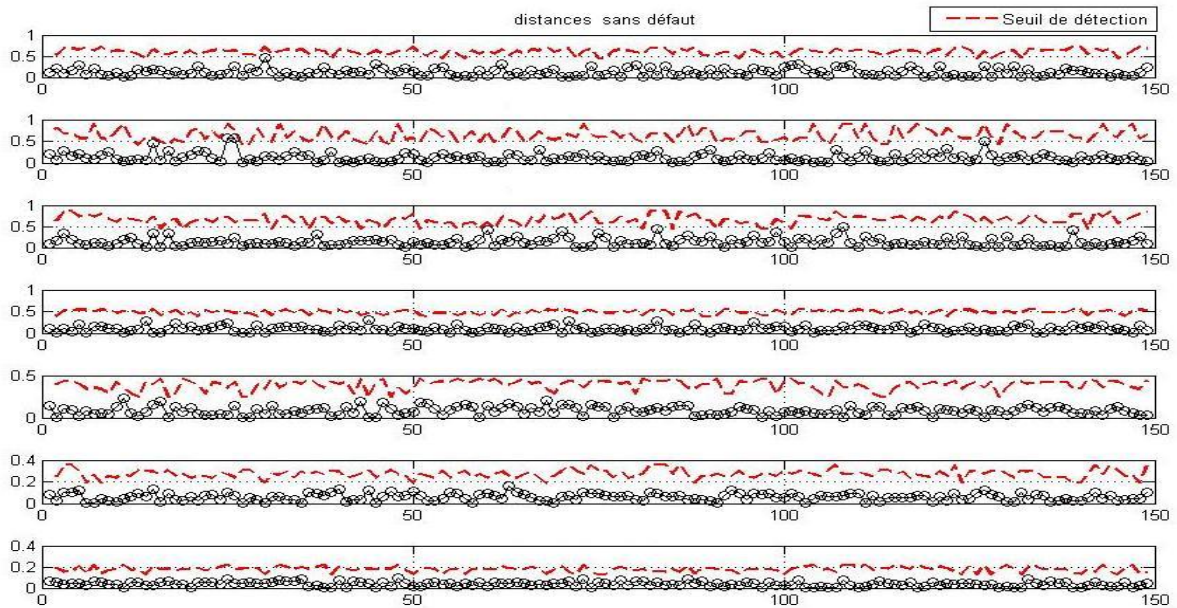
La détection des défauts est obtenue par le calcul du seuil de détection  $s_d(i)$

Ce seuil  $s_d(i)$  a été défini précédemment ( formule (3.3) ).

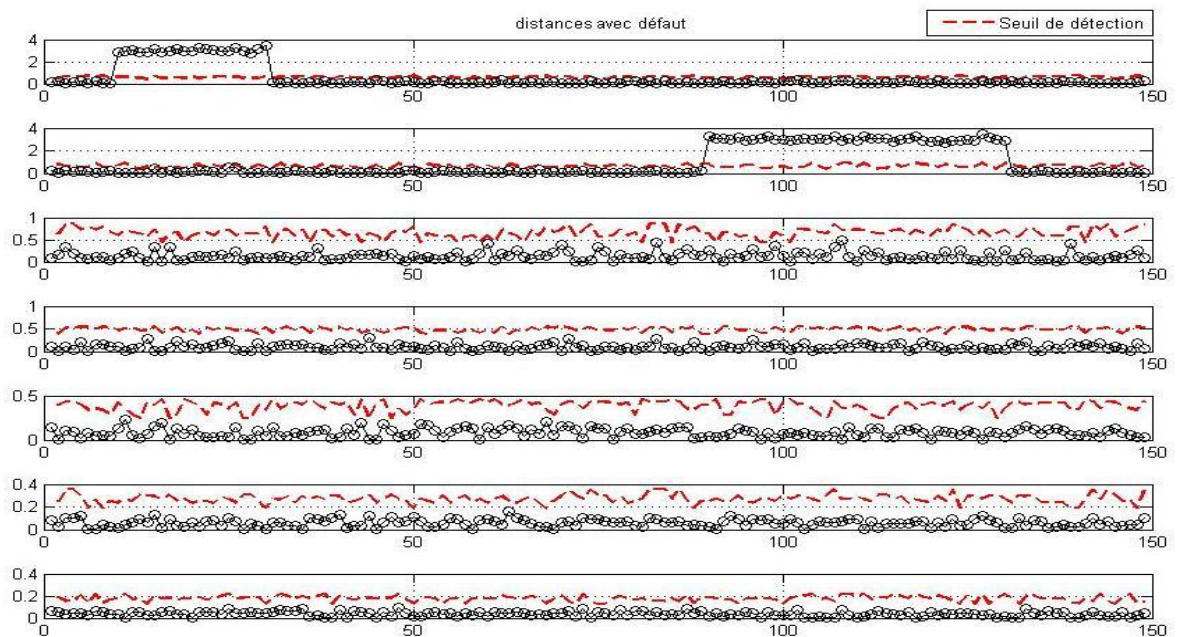
Une donnée est valide si la fonction d'activation est supérieure au seuil  $\alpha$  (dans notre cas

$\alpha = 1$ ), et si la distance euclidienne  $\|x - w_i\|$  est inférieure au seuil  $s_d(i)$ . Sinon il y a défaut.

Les schémas des figures suivantes (figure IV.4.a, et figure IV.4.b) illustrent le principe de la démarche adoptée.



**Figure IV.4.a :** Distances entre les données et les neurones gagnants (cas d'un fonctionnement normal appliqué aux signaux de l'exemple 1)

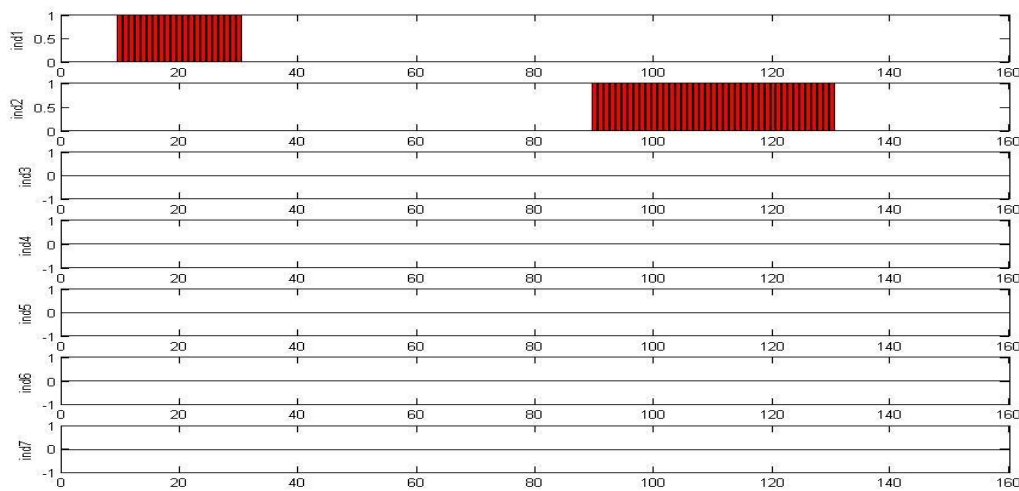


**Figure IV.4.b :** Distances entre les données et les neurones gagnants appliqué aux signaux de l'exemple 1 avec apparition de défauts

Pour mieux appréhender la détection et la localisation des défauts nous avons défini une fonction logique comme suit :

$$Ind_i = \begin{cases} 1 & \text{Si } \|x_i(k) - W\| > s_d \text{ et } x_i(k) \text{ est considéré invalde} \\ 0 & \text{Sinon et } x_i(k) \text{ est considéré valide} \end{cases} \quad (4.5)$$

La figure IV.5 montre le principe de cette fonction sur des données avec présence de défauts.



**Figure IV.5 :** Fonction logique pour localisation et identification des défauts

#### IV.4.Reconstruction des données

La reconstruction des données est fondée sur l'utilisation de la formule (3.5)

Afin d'évaluer les performances de cette méthode pour la validation et la reconstruction de données, une étude de simulation a été réalisée. Différents défauts de variables ont été introduits artificiellement dans les données (cas de l'exemple 1).

La figure IV.6 illustre la reconstruction des données utilisant l'approche de la carte de Kohonen (SOM), les performances de cette approche sont garantit l'estimation des données et aussi l'élimination des défauts qui intervient sur les variables.

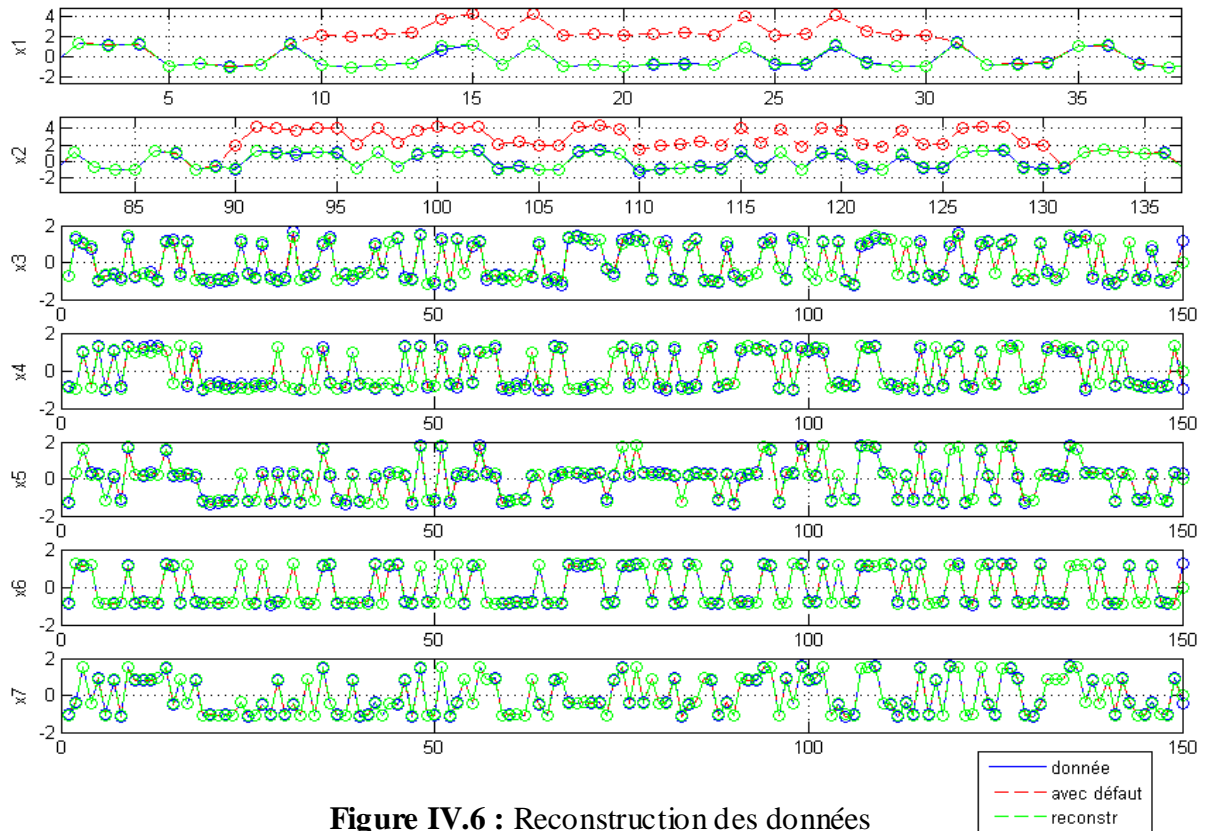


Figure IV.6 : Reconstruction des données

#### IV.5. Cas de deux défauts simultanés (Multiple):

Dans l'exemple suivant, nous présentons la réaction des approches de détection et localisation de défaut étudiée dans ce mémoire, lors de l'introduction de deux défauts simultanément sur les variables [THA 08].

##### Exemple 2

$$x_1(t) = 1 + v(t)^2 + \sin\left(\frac{t}{10}\right), v(t) \sim \mathcal{N}(0,1)$$

$$x_2(t) = 2 \sin\left(\frac{t}{6}\right) \cos\left(\frac{t}{4}\right) \exp\left(-\frac{t}{n}\right)$$

$$x_3(t) = \log\left(\frac{x_2(t)^2}{6}\right)$$

$$x_4(t) = x_1(t) + x_2(t)$$

$$x_5(t) = x_1(t) - x_2(t)$$

$$x_6(t) = 2x_1(t) + x_2(t)$$

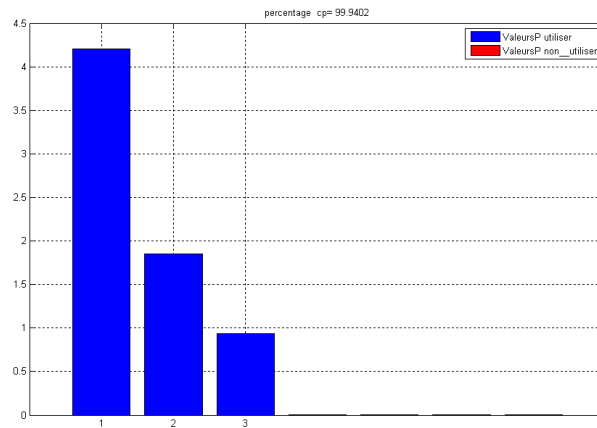
$$x_7(t) = x_1(t) + x_3(t)$$

(4.6)

Ainsi que des défauts représentés par des biais d'amplitude égale à 3 définis de la façon suivante :

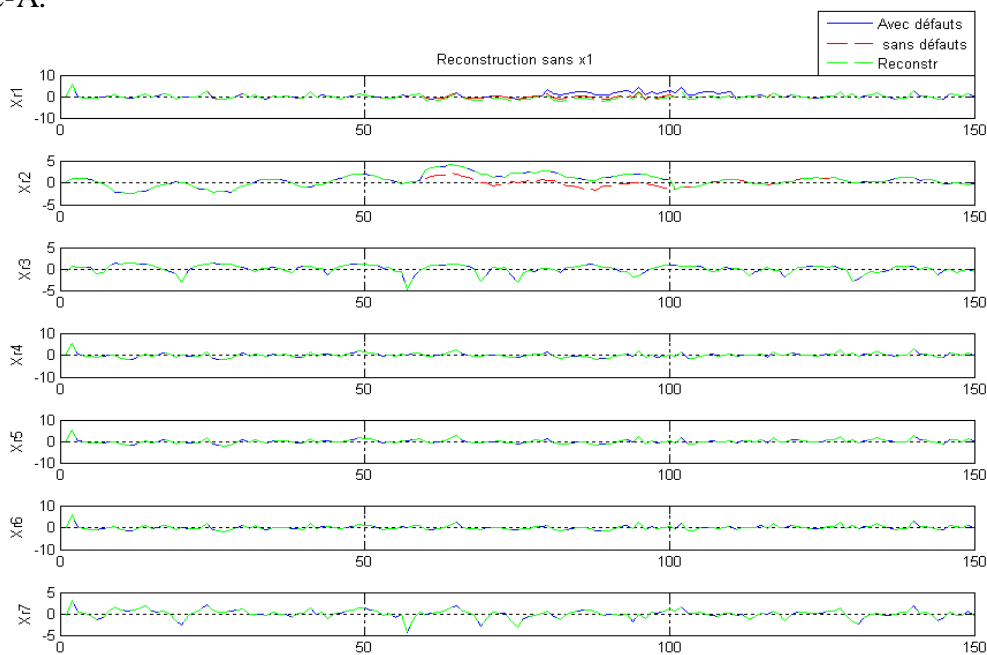
- Défauts sur  $x_1$  pour  $\in [80,130]$  .
- Défauts sur  $x_2$  pour  $\in [60,100]$  .

**IV.5.1.Résultats de l'ACP**



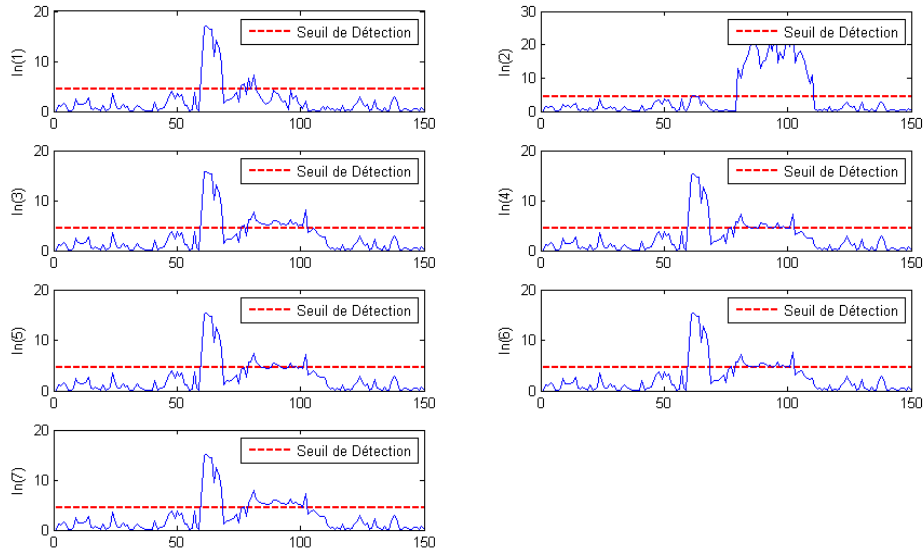
**Figure IV.7** : valeurs propres utilisées pour la reconstruction du modèle ACP de l'exemple 2

La figure IV.7 montre les valeurs propres utilisée pour la détermination du modèle ACP. Les différents approches du choix de nombre de composantes principales sont présentées aux Annexe-A.



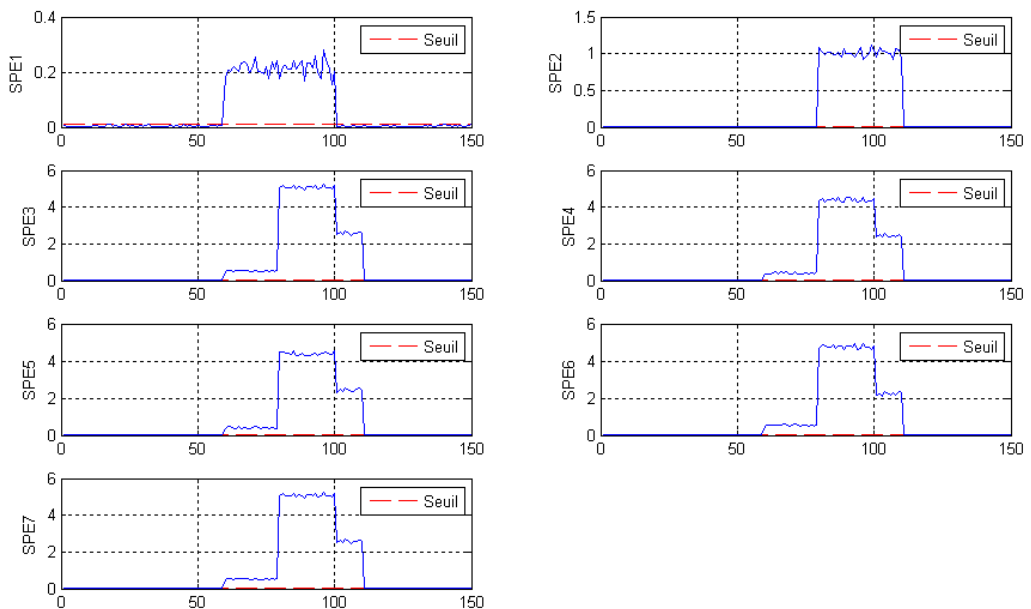
**Figure IV.8** : reconstruction des variables sans  $x_1$  de l'exemple 2

La figure IV.8 présente la reconstruction des données de l'exemple 2 sans utiliser la variable  $x_1$ , cette figure confirme l'effet de la reconstruction des données par l'élimination les effets des défauts qui intervient sur la variable  $x_1$ .



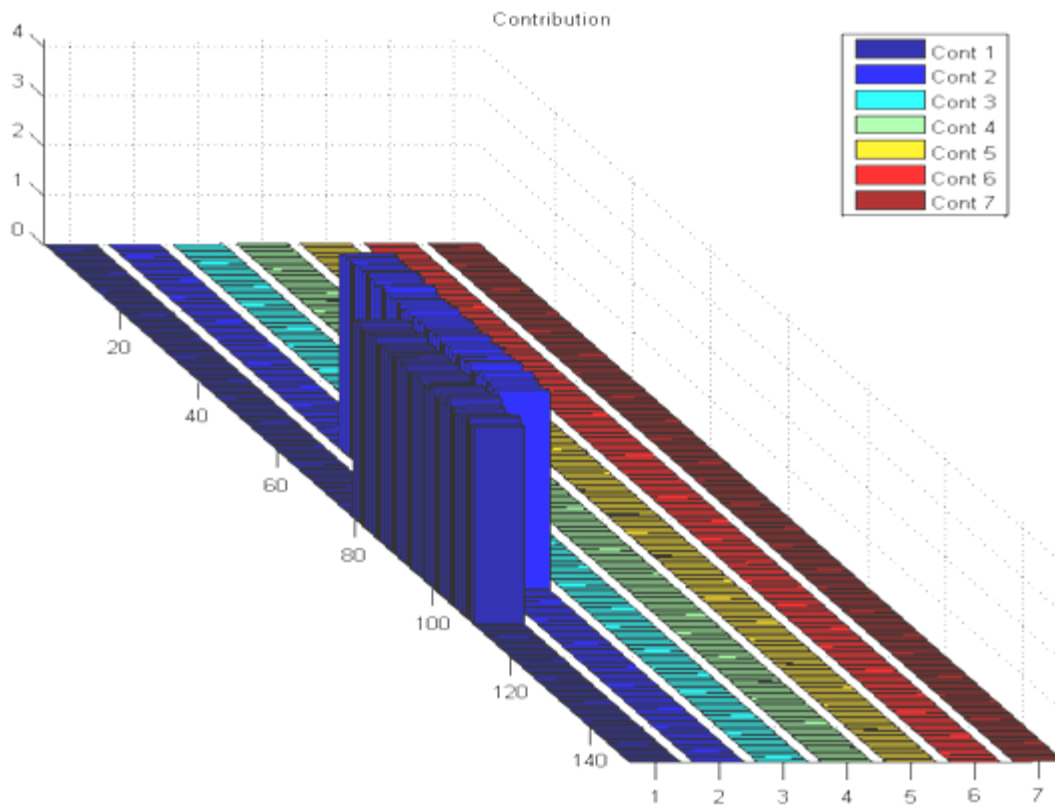
**Figure IV.9 :**  $In_i$  test sur les dernières composantes après reconstruction

La figure IV.9 montre l'approche de détection et localisation des défauts par le test que  $In_i$  des dernières composantes principales après la reconstruction des données sans la  $i^{i\grave{e}me}$  variable de l'exemple 2, Alors, à cause de la nature des données, les défauts qui intervient sur  $x_1$  et  $x_2$  et le modèle de l'ACP utilisé, on remarque bien que  $In_i$  avec  $(i = 1, \dots, 7)$  semble assez irrégulier à l'instant de l'intervention des défaut sauf  $In_2$ , pour cette irrégularité sur  $In_i$  la détection et la localisation des défauts est très difficile.



**Figure IV.10 :**  $SPE_i$  après reconstruction des données sans la  $i^{i\grave{e}me}$  variable de l'exemple 2

La figure IV.10 représente l'évolution des différents  $SPE_i$  après reconstruction des données sans la  $i^{i\text{ème}}$  variable de l'exemple 2, où les  $SPE_i$  pour  $i = 3, \dots, 7$  semblent similaires, ce qui signifie inutilité de l'effet de la reconstruction pour l'élimination des défauts sur les variables qui correspondent. Alors, on peut dire que les défauts n'intervient pas sur les variables  $x_i$  ( $i = 3, \dots, 7$ ). Mais pour  $SPE_1$  et  $SPE_2$  besoin d'une analyse pour détecter et localiser les défauts. Pour le  $SPE_1$  est dessus du seuil à l'instant  $t \in [60, 100]$ , et  $t \in [80, 130]$  pour  $SPE_2$ , ce qui indique directement l'intervention des défauts sur  $x_1$  à l'instant  $t \in [100, 130]$ , et sur  $x_2$  à l'instant  $t \in [100, 130]$ . Mais pour  $t \in [80, 100]$  il est difficile de décider la variable en cause, pour cela, cette méthode approche est besoin de reconstruction des données sans utiliser plus deux variables pour assurer la localisation des défauts.

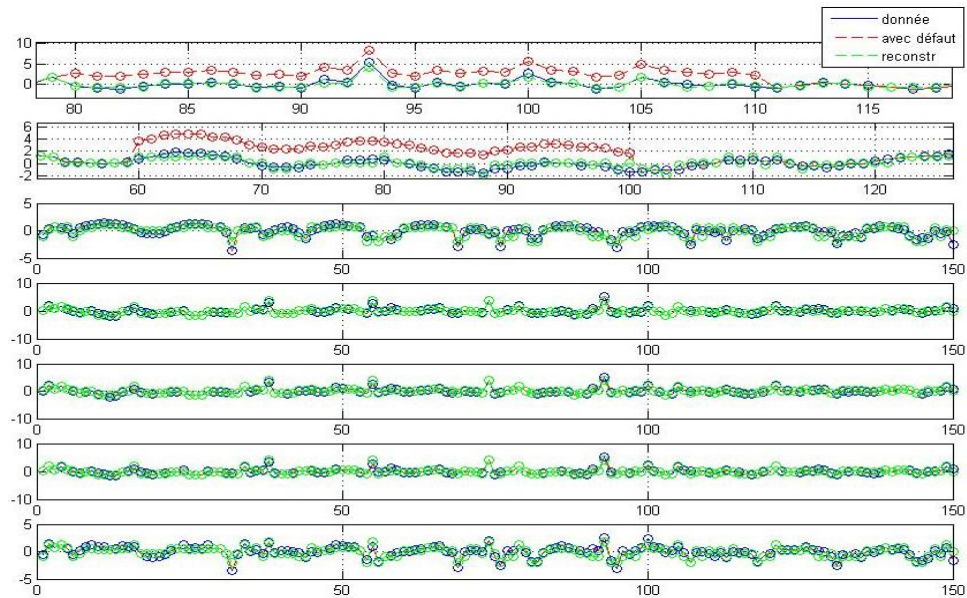


**Figure IV.11** : contribution  $cont_i^{SPE}$  de l'exemple 2

La figure IV.11 montre l'évolution des différentes contributions  $cont_i^{SPE}$  au  $SPE_i$  après reconstruction des données sans la  $i^{i\text{ème}}$  variable de l'exemple 2, pour donner une bonne visualisation et mieux localiser les variables en cause nous avons utilisé une présentation en 3D. Ce qui traduit l'efficacité de cette méthode dans la localisation des défauts.

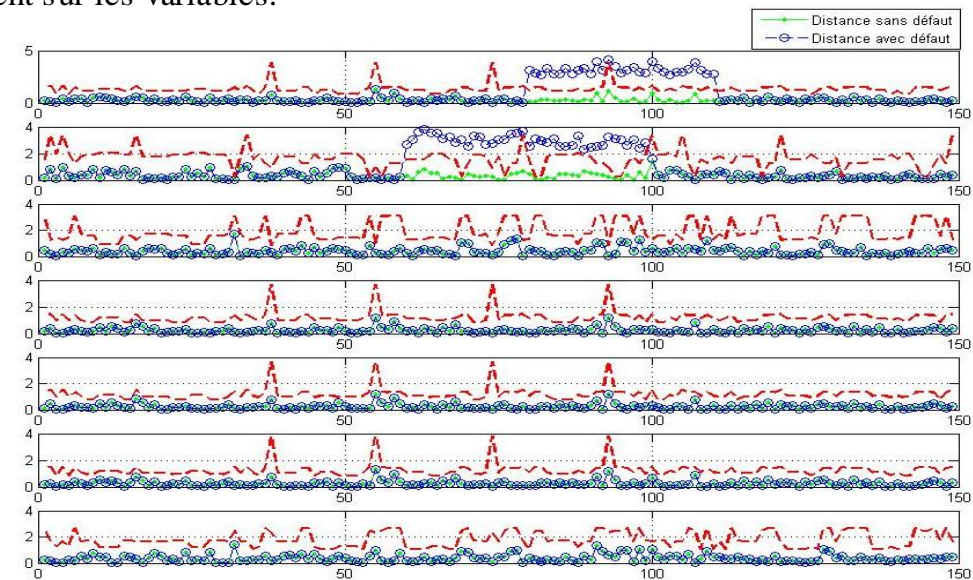
### IV.5.2. Résultats des cartes de Kohonen

La carte de Kohonen utilisée est une carte de taille  $4 \times 4$ .



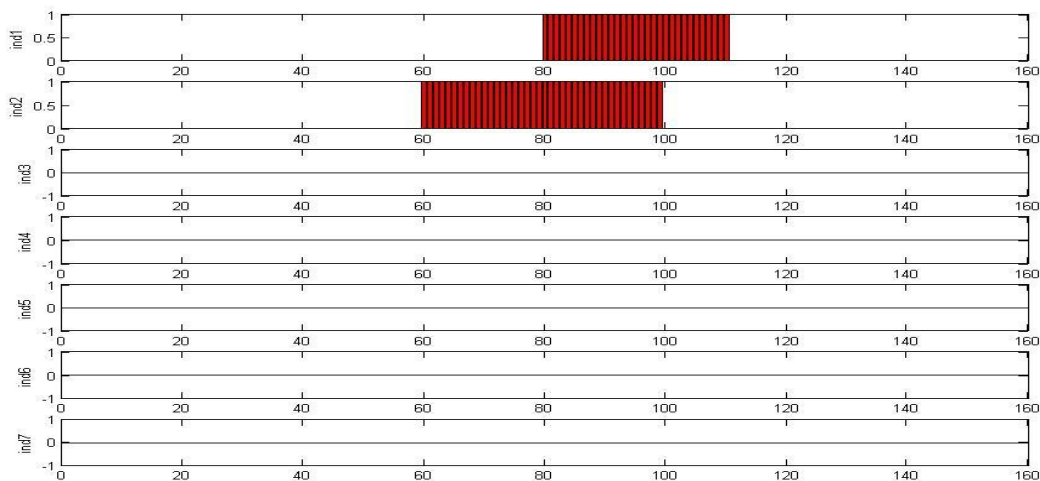
**Figure IV.12 :** Reconstruction des données par la carte de Kohonen (SOM) de l'exemple 2.

La figure IV.12 présente la reconstruction des données de l'exemple 2 à l'aide de la carte de Kohonen, ce qui garantit l'estimation des données et l'élimination des défauts qui intervient sur les variables.



**Figure IV.13 :** Distances entre les données et leurs classes pour les deux cas de l'exemple 2. (cas d'un fonctionnement normal et avec des défauts simultanés)

La figure IV.13 représente l'évolution des différents Distances entre les données et leurs classes pour les deux cas de l'exemple 2 (cas d'un fonctionnement normal et avec des défauts simultanés), les distances au dessus du seuil de détection indique l'intervention des défauts dans les variables  $x_1$  pour  $t \in [80,130]$  et  $x_2$  pour  $\in [60,100]$  , ce qui confirme la capacité de détecter et localiser les défauts avec l'approche proposée.



**Figure IV.14 :** Détection et localisation des défauts de l'exemple 2

La figure IV-14 montre les différents indicateurs de détection issues de la comparaison des distances avec les seuils de détection de chaque classe, ce qui montre la présence des défauts aux variables  $x_1$  pour  $t \in [80,130]$  et  $x_2$  pour  $t \in [60,100]$ . En effet, la validation des performances de notre approche de détection et localisation de défaut.

#### **IV.6.Conclusion**

Dans ce chapitre, nous avons traité deux exemples de différentes méthodes étudiées dans ce mémoire, le premier exemple consiste à la détection et la localisation des défauts basée sur la carte de Kohonen dans le cas des défauts successifs, et le deuxième exemple consiste la détection et la localisation des défauts basée sur les deux méthodes ( ACP et la carte de Kohonen) dans le cas des défauts simultanés ou multiple .

Pour les deux exemples étudiés, les résultats obtenus sont satisfaisants, la carte de Kohonen (SOM) et aussi l'analyse en composant principale assurent la modélisation et le diagnostic du système des données.

## V. CONCLUSIONS ET PERSPECTIVES

Nous avons présenté dans ce mémoire, deux méthodes de détection et localisation des défauts : l'une fondée sur approche statistique basée sur l'ACP, l'autre, sur un modèle neuronale, utilisant la carte auto-organisatrice de Kohonen.

Pour ce qui est de la détection et de la localisation du défaut par ACP, celle-ci est utilisée pour modéliser le comportement d'ensemble du processus en fonctionnement normal, puis les défauts sont ensuite détectés en comparant le comportement observé à celui donné par le modèle. Ce procédé indique s'il existe un défaut au niveau global, c'est à dire pour l'ensemble des capteurs, sans information sur l'origine de ce défaut. L'inconvénient de cette méthode c'est quelle devient lourde du point de vue temps de calcul lorsque l'espace de représentation des données devient grand. De plus le modèle ACP n'est pas adaptable à la non linéarité des systèmes. Pour contourner ce problème, nous avons choisi un modèle connexionniste basé sur les cartes topologiques de Kohonen. Dans ce modèle, nous avons démontré que la procédure de détection et de localisation des défauts devient une opération simple, rapide du point de vue temps de calcul, une fois la phase d'apprentissage effectuée. Le modèle de non linéarité ne se pose pas le choix de ce modèle.

Les perspectives d'étendre ce travail à d'autres travaux réside en la combinaison de ces différentes techniques entre elles pour atteindre les objectifs désirés. Dans le cas de données quantitatives, on peut au préalable réduire la dimension en faisant une analyse en composantes principales et en ne conservant qu'un nombre réduit de coordonnées. On peut se servir de la classification en super-classes réalisée à partir de l'algorithme de Kohonen pour définir une nouvelle variable qualitative par exemple.

# BIBLIOGRAPHIE

- [AGU 80] Aguilar-Martin J., Balssa M., R.D.M. Estimation récursive d'une partition. Exemple d'apprentissage et auto-apprentissage dans R. Technical Report No. 880139, LAAS-CNRS, 1980.
- [BAL09] Baligh Mnassri, Diagnostic de Défauts par Analyse en Composantes Principales, 2009.
- [BRA89] Bradburn D. S. Reducing transmission error effects using a self organizing network. In *Proceedings of IJCNN*, volume 2, pages 531-37, Washington, 1989.
- [DEN97] Denoeux *et al.* EM2S - High level data fusion. Rapport n°: P 22442 - EM2S, November 1997.
- [DUB 87] Dubes, R. C.. How many clusters are best – an experiment. *Pattern Recognition*, vol. 20, no. 6, pp. 645-663, novembre 1987.
- [DUN 96] Dunia R., Qin S. J. and Edgar T. F. (1996). Identification of faulty sensors using principal component analysis. *AIChE Journal*, vol. 42, N<sup>o</sup>. 10, pp. 2797-2812.
- [ERW 91] Erwin, E., Obermayer, K., et Schulten, K.J., Convergence properties of Self Organizing Maps. *Proceedings of ICANN91*, pages 409-14, 1991.
- [FAD 10] Fadda M-L, Bennacer L, HarkatM-F, Détection et localisation des défauts par analyse en composantes principales (ACP) et carte de Kohonen (SOM),CIFIE.2010.
- [FRA 00] Frank P. M., Ding S. X., et Marcu T., Model-Based fault diagnosis in technical process. *Transactions of the Institution of Measurement and Control*, vol. 22 no. 1, pp. 57-101. 2000.
- [FRI 91] Fritzke. B. *Unsupervised clustering with growing cell structures*. Proc. Of the IJCNN, Seattle, 1991.
- [FRI 93].Fritzke B. Kohonen feature maps and Growing Cell Structures - A performance comparison. In C. L. Giles, S. J. Harson, and J. D. Cowan, editors, *Advances in Neural Information Processing Systems*, volume 5, pages 123-30. Kaufmann San Mateo CA, 1993.
- [FRI 94] Fritzke.B. *Growing cell structures – a self organizing network for unsupervised and supervised learning*. *Neural Network*, Vol. 7, 1994.
- [FRI 95-1] Fritzke. B. *Growing grid – a self organizing network with constant neighborhood range and adaptation strength*. *Neural Processing Letter*, Vol. 2, 1995.
- [FRI 95-2]. Fritzke. B. *A growing neural gas network learns topologies*. *Advances in Neural Information Processing Systems*. MIT Press, 1995.
- [GAR 98] Garcia E., Morant F. “Aplicaciones de los modelos de eventos discretos obtenidos mediante el operador de composición sincronía” XIX Jornadas de Automatica, Madrid., pp. 199-204, 1998.
- [GAR 99] Garcia E., Morant F., “Diagnostico de fallos modular y control supervisor para un proceso químico de mezclas” XX Jornadas de Automatica, Salamanca, pp. 267-272, 1999.
- [HAR 01] Harkat. M, Mourot.F, Ragot. J, Détection de défauts de capteurs d'un réseau de surveillance de la qualité de l'air, Centre de Recherche en Automatique de Nancy, Ecole Nationale Supérieure de Géologie, 2001

- [HAR03] Harkat Mohamed-Faouzi, Thèse « Détection et Localisation de Défauts par Analyse en Composantes Principales ».2003
- [HAR 04] Harkat M-F, Tharrault Y, Mourot G *et* Ragot J .Modélisation et diagnostic des systèmes non linéaires par Multi-ACP.2004
- [HAR 05] Harkat M.F., Mourot G., Ragot J., « Sensor fault detection and isolation of an air quality monitoring network using non linear principal component analysis »,16th IFAC World Congress, Prague, Czech Republic, July 4-8, 2005.
- [HIM 78] Himmelblau, D. M. Fault Detection and Diagnosis in Chemical and petrochemical processes. Amsterdam: Elsevier press, 1978.
- [ISE 84] Isermann R. Process fault diagnosis based on modelling and estimation methods- A survey. *Automatica*, vol 20, pp. 387-404, 1984.
- [JAC 79] Jackson J. E., Mudholkar G. S.. Control procedures for residuals associated with principal component analysis. *Technometrics*, vol. 21, N± 3. 1979.
- [JAI 88] Jain A. K. et Dubes, R. C., Algorithms for clustering Data Prentice- Hall advanced references series. Prentice-Hall, Inc., Upper Saddle River, N. J., 1988.
- [JAD 88] Jain, A. K. and Dubes, R. C. Algorithms for Clustering Data. Prentice-Hall advanced reference series. Prentice-Hall, Inc., Upper Saddle River, NJ., 1988.
- [KIN 67] King, B. Step-wise clustering procedures, *Journal of the American Statistical Association*, vol. 69, pp. 86-101, 1967.
- [KOH81]. Kohonen, T, Automatic formation of topological maps of patterns in a self organizing system. In *Proceedings of 2<sup>nd</sup> Scandinavian Conference on Image Analysis*, pages 214-220,1981.
- [KOH 82] Kohonen, T, Self organized formation of topological correct feature maps. *Biological Cybernetics*, 43, 1982.
- [KOH 84] Kohonen, T, Makisara, K. et Saramaki., T. Phonotopic maps - Insightfull representation of phonological features for speech recognition. In *Proceedings of the 7th International Conference on Pattern Recognition*, 182-185, Montreal, 1984.
- [KOH 88] T. Kohonen, *Self-organization and associative Memory*. Springer-Verlag, Third Edition, Vol. 8, 1988.
- [KOH 90] Kohonen T, Raivio K, Simula O, Nenta O, et Enriksson J. An adaptative discrete signal detector based on Self Organizing Maps. In *Proceedings of IJCNN*, volume 2, pages 249-52, Washington, 1990.
- [KOH 91] Kangas, editors, *Proceedings of ICANN91*, Vol. 1, 777-782, North Holland, 1991.
- [KOH 95-1] Kohonen, T, Hynninem, J., Kangas, J., et Laaksonen, J., The Self Organizing Map program package SOM PAK V3.1 SOM Programming Team of the Helsinki University of Technology – Laboratory of Computer and Information Science, 1995.
- [KOH 95-2] Kohonen, T, Self Organizing Maps, volume 30 of Springer series of information sciences. Springer Verlag, 1995.
- [KOO 89] Koodmen F., LAMDA : *Logiciel d'Analyse de Données Multidimensionnelles par Apprentissage et Reconnaissances de Formes*. Manuel de reference et d'utilisation. LAAS-CNRS, Toulouse. 1989.
- [KOU 96] KOURTI T., MACGREGOR J. F., « Multivariate SPC Methods for Processes and Product monitoring », *Journal of Quality Technology*. vol. 28, no 4,1996
- [KOU 05] KOURTI T., « Application of latent variable methods to process control and multivariate statistical process control in industry », *International Journal of Adaptive Control and Signal Processing*. vol. 19, no 4, 2005.
- [KRA 93] Kramer M. A., et Mah R. S. H., Model-Based Monitoring. In: Rippin D., Hale J., et Davis J. (Eds), *Proceedings of the second International Conference on “foundations of computer-aided process operations” (FOCAPO)*, pp. 45-68,1993.

- [LAF 91] S. Lafortune et E. Chen, "A relational algebraic approach to the representation and analysis of discrete-event systems" in proceedings of the American Control Conference, Boston, MA, pp. 2893-2898, 1991.
- [MAM 77] Mamdani E., "application of fuzzy logic to approximate reasoning using linguistic system." Fuzzy sets and systems, vol. 26, pp. 1182-1191, 1977.
- [MAR 90] Martinetz, T, Ritter, H. J. et Schulten, K. J. 3 dimensional neural network for learning visuo-motor coordination of a robot. IEEE Transactions on Neural Networks, Vol. 1, 131-36, 1990.
- [MCL 88] McLachlan, G.J. et Basford, K.E., Mixture Models: Inference and Applications to Clustering, New-York: Marcel Dekker, 1988.
- [MCQ 67] McQueen, J. Some methods for classification and analysis of multivariate observations. In proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp. 281-297, 1967.
- [MIN 04] Ming Ge, R. Du, and Yangsheng Xu. Fault Detection Using Hierarchical Self-organizing Map. 2004.
- [MNA 08] MNASSRI B., EL-ADEL E.-M., OULADSINE M., « Fault Localization Using Principal Component Analysis Based on a New Contribution to the Squared Prediction Error», 16th Mediterranean Conference on Control and Automation, Ajaccio, France, 2008, p. 65-70.
- [MUR 84] Murtagh, F., A survey of recent advances in hierarchical clustering algorithms which use cluster centers. Comput. J., vol. 26, pp. 354-359, 1984.
- [PET 98] Petit-Renaud, S. et Denoeux, T., A neuro-fuzzy system for missing data reconstruction. Proc. IEEE Workshop on Emerging Technologies, Intelligent Measurement and Virtual systems for instrumentation and measurement, 78-84, Saint-Paul, USA, May 1998.
- [QIN 00] Qin S.J., Dunia R., « Determining the number of principal components for best reconstruction», Journal of Process Control, vol. 10, 2000, p. 245-250.
- [QIN 03] Qin.S. Joe, Hongyu Yue and Ricardo Dunia. A Self-validating Inferential Sensor for Emission Monitoring. vol. 17, no 8-9, 2003
- [RIT 92] Ritter, H., Martinetz, T. et Schulten., K. Neural computation and Self Organizing Maps: an introduction. Addison Wesley, New-York, 1992.
- [RAM 87] Ramadge P. J. et Wonham W. M., "Modular Feedback logic for discrete event systems," SIAM Journal of Control and Optimization, vol. 25, no. 5, pp. 1202-1218, mai 1987.
- [SAM 94] Sampath M., Sengupta R., Lafortune S., Sinnamohideen K. and Teneketzis D., "Failure diagnosis using discrete-event models" IEEE Transactions on Control Systems, vol. 4, no. 2, pp. 105-124, mars 1994.
- [SAM 95] Sampath M., R. Sengupta, S. Lafortune, K. Sinnamohideen and D. Teneketzis, "Dioagnosability of discrete-event systems" IEEE Transactions on Automatic Control, vol. 40, no. 9, pp. 1555-1575, 1995.
- [STE 97] Stephan Morgenthaler, Introduction à la statistique, collection Méthodes mathématiques pour l'ingénieur, Presses Polytechniques et Universitaires Romandes, 1997.
- [SNE 73] Sneath, P. H. A. et Sokal, R. R., Numerical Taxonomy. Freeman, London, UK, 1973.
- [TAK 85] Takagi T., Sugeno M., Fuzzy identification of systems and its application to modelling and control. IEEE Transactions on systems, man and cybernetics, vol. 15, no. 1, pp. 116-132, 1985. *Références* 149.

- [TOS 05] Toscano R., Commande et diagnostic des systèmes dynamiques: Modélisation, analyse, commande par PID et par retour d'état, diagnostic. Ellipses édition Marketing, S. A., 2005.
- [THA08] Tharrault Y, thèse « Diagnostic de fonctionnement par analyse en composantes principales Application à une station de traitement des eaux usées ». 2008
- [ZRE92] Zrehen S. et Blayo F. A geometric organization measure for Kohonen's map. In Proceedings of NeuroNimes, pages 603-10, Nîmes, November 1992.
- [VAL00] VALENTIN N, Construction d'un capteur logiciel pour le contrôle automatique du procédé de coagulation en traitement d'eau potable. Thèse, 2000.
- [VEN 03] Venkatasubramanian V., Rengaswamy R. Yin K., Kavuri S., A review of process fault detection and diagnosis Part I: Quantitative model-based methods, Computer and Chemical Engineering, 2003.
- [VER 90] Vercauteren, L., Sieben, G., Praet, M., Otte, G., Vingerhoeds, L., Boullart, L., Lalliauw L, and Roeds, H., The classification of brains tumours by a topological map. In Proceedings of ICNN, Vol. 1, 387-391, Paris, 1990.
- [VIL97] Villmann Th, H.U. Bauer. *The GSOM-algorithm for growing hypercubical output spaces in self-organizing maps*. WSOM, 1997.
- [WAR 63] Ward, J. H. Jr. Hierarchical grouping to optimise an objective function. Journal of the American Statistical Association, vol. 58, 236-244, 1963.
- [WES98] WESTERHUIS J. A., GURDEN S. P., SMILDE A. K., « Generalized contribution plots in multivariate statistical process monitoring », *Chemometrics and Intelligent Laboratory Systems*.
- [ZAD 65] Zadeh, L. Fuzzy sets. Information Control, vol. 8, pp. 338-353, 1965.
- [ZAN 71] Zahn, C. T.. Graph-Theoretical methods for detecting and describing gestalt clusters. *IEEE Trans. Comput. C-20* (Apr.), pp. 68-86., 1971.

# **ANNEXE**

# Annexe A

## Résumé de la méthode de l'analyse en composantes principales (ACP)

L'ACP consiste à remplacer les variables par de nouvelles variables de variance maximale, non corrélées deux à deux et qui sont des combinaisons linéaires des variables d'origine. Ces derniers peuvent être structurés dans une matrice comme suit :

$$X = \begin{bmatrix} x_{1,1} & \dots & x_{n,1} \\ \vdots & \ddots & \vdots \\ x_{1,m} & \dots & x_{n,m} \end{bmatrix} \quad (\text{A.1})$$

### 1-La valeur moyenne

La première opération consiste à déterminer la valeur moyenne de chaque colonne de la matrice de donnée .

$$\bar{x}_k = \frac{\sum_{i=1}^n x_{k,i}}{n} \quad (\text{A.2})$$

### 2-La variance et L'écart type

La notion de moyenne n'est pas suffisante pour décrire, même succinctement, un ensemble de nombres. La dispersion de ces nombres autour de la moyenne est une information tout aussi intéressante et qui vient compléter l'information donnée par la moyenne. La dispersion est assimilée à la notion d'écart à la moyenne. L'écart quadratique est égal à la somme des carrés des écarts sur chaque valeur de la variable. Sa valeur moyenne nommée variance servira à estimer cette dispersion:

$$\sigma_k^2 = \frac{1}{n} \sum_i^n (x_{k,i} - \bar{x}_k)^2 \quad (\text{A.3})$$

### 3- Normalisation des données

Normaliser les données consiste à centrer et réduire les données sur l'origine du repère en s'assurant que la moyenne des données normalisées suivant chacun de critère de classement soit égale à l'unité.

Si  $Y$  est la matrice normalisée associée à  $X$ , alors :

$$y_{i,j} = \frac{x_{i,j} - \bar{x}_j}{\sigma_j} \quad (\text{A.4})$$

### 3-Calcul de corrélations

La matrice de corrélation  $\Sigma$  associée à la nouvelle matrice  $X$  centrée et réduite est

$$\Sigma = \frac{1}{(N - 1)} X^T X \quad (\text{A.5})$$

Cette matrices est carrées (de taille  $m$ ), symétriques, et réelles. Elles est donc diagonalisables dans une base orthonormée.

Les coefficients corrélations indiquant l'influence que les variables ont les unes sur les autres.

### 4- valeurs et vecteurs propres

Avec une décomposition spectrale de la matrice de corrélation  $\Sigma$  permet d'écrire sous la forme suivante :

$$\Sigma = P\Lambda P^T = \sum_{i=1}^m \lambda_i p_i p_i^T \quad (\text{A.6})$$

où  $p_i$  est le  $i^{eme}$  vecteur propre de  $\Sigma$  et  $\lambda_i$  est la valeur propre correspondante.

### 5-Détermination de la structure du modèle

La structure du modèle ACP est basée sur la détermination du nombre de composantes à retenir. Quelques approches sont présentées par la suite pour la détermination du nombre de composantes [QIN 00, QIN 03].

#### Pourcentage cumulé de la variance totale (PCV)

L'idée de base de cette approche est que chaque composante principale est représentative d'une portion de la variance des mesures de l'information étudiée. Les valeurs propres de la matrice de corrélation sont des mesures de cette variance et peuvent donc être utilisées dans la sélection du nombre de composantes principales. Pour faire le choix de  $\ell$ , il faut estimer le pourcentage de la variance totale qu'on veut conserver. Le nombre de composantes est alors le plus petit nombre pris de telle sorte que ce pourcentage soit atteint ou dépassé; les composantes sont choisies successivement dans l'ordre des variances décroissantes. Le pourcentage de variance expliquée par les premières composantes est donné par :

$$PCV(\ell) = 100 \left( \frac{\sum_j^\ell \lambda_j}{\sum_j^m \lambda_j} \right) \% \quad (\text{A.7})$$

La variance du bruit étant inconnue a priori, la décision basée seulement sur le pourcentage de la variance expliquée est un peu arbitraire. Sa capacité à fournir le nombre correct de composantes principales dépendra fortement du rapport signal sur bruit.

### la moyenne des valeurs propres

Cette règle consiste à ne prendre en considération que les composantes pour lesquelles la valeur propre est supérieure à la moyenne arithmétique de toutes les valeurs propres. En particulier, si on travaille sur les données centrées réduites, cela revient à négliger les composantes dont la variance est inférieure à l'unité

$$\frac{1}{m} \text{trace}(\Sigma) = 1 \quad (\text{A.8})$$

Dans le cas du modèle de calcul à partir de la matrice de covariance  $\Sigma$ , la moyenne arithmétique des valeurs propres est donnée par  $\frac{1}{m} \text{trace}(\Sigma)$ .

### 2.5.3. Validation croisée

La validation croisée est un critère très populaire pour le choix du nombre de composantes dans un modèle ACP. La base de cette méthode est d'estimer les mesures d'un jeu de données de validation à partir d'un modèle qui a été calculé à partir d'un jeu de données d'identification et de comparer ces estimations avec les valeurs mesurées. Cette procédure de validation croisée est basée sur la minimisation de la somme des carrés des erreurs de prédiction (*PRESS*) entre les données mesurées et celles estimées par le modèle obtenu à partir d'un jeu d'identification différent.

$$PRESS(\ell) = \frac{1}{Nm} \sum_{k=1}^N \sum_{i=1}^m (\hat{x}_i^{(\ell)}(k) - x_i(k))^2 \quad (\text{A.9})$$

Où  $N$  étant la taille du jeu de validation.

Une version simplifiée de l'algorithme permettant le calcul du nombre de composantes principales par la validation croisée est la suivante :

1. Diviser les données en un jeu d'identification et un jeu de validation.

2. réaliser une ACP avec  $\ell$  composantes ( $\ell = 1, \dots, m$ ) sur le jeu d'identification et calculer les critères correspondant sur le jeu de validation  $PRESS(1), \dots, PRESS(m)$ .
3. la  $\ell^{eme}$  composante pour laquelle le minimum de  $PRESS$  apparaît sera la dernière composante à retenir et  $\ell$  sera le nombre de composantes principales retenu.

### Principe de la variance non reconstruite

La notion de variance non reconstruite est basée sur le principe de reconstruction des données chapitre-II, alors le nombre de composantes principales du modèle en minimisant l'équation (2.12) .

La figure suivante illustre les différentes approches pour l'exemple 2

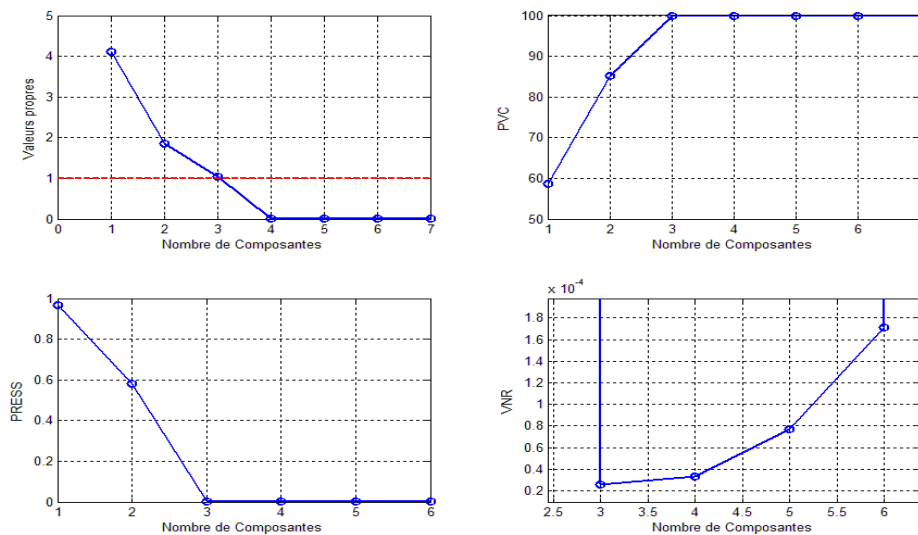


Figure A.1 – Evolution des différents critères de sélection du nombre de composantes en fonction du nombre de composantes pour l'exemple 2

## Annexe B

# Résumé des cartes auto-organisatrices de Kohonen (SOM)

### 3.4 L'algorithme Apprentissage de Kohonen

L'apprentissage met en correspondance l'espace des entrées et la carte d'adaptation des prototypes de telle manière que les individus proches dans l'espace d'entrée soient associés au même neurone ou à des neurones proches dans la carte. La mise à jour des poids pour chaque neurone se fait par évaluation du voisinage du neurone gagnant dans la carte. L'adaptation est d'autant plus forte que les neurones sont voisins.

Ainsi, l'algorithme d'apprentissage de la carte topologique de Kohonen comprend principalement les phases suivantes [KOH 81]:

- **Phase d'Initialisation**

- a) **Initialisation de Données**

Une fois que le groupe des variables à utiliser dans la classification est sélectionné, chaque variable doit être normalisé. Ceci empêche n'importe quel variable de maîtriser le reste du groupe pendant les étapes de la classification.

- b) **Initialisation des Poids**

Généralement, l'algorithme d'apprentissage de la carte de Kohonen consiste à initialiser les poids des neurones en leur attribuant des valeurs aléatoires faibles.

Une fois le réseau correctement initialisé, trois phases apparaissent lors de la formation de la carte auto-organisée de Kohonen.

- **Phase de compétition**

La phase de compétition consiste à chercher le neurone vainqueur d'après d'un critère similarité[KOH 88].

Soit le vecteur d'entrée  $x$  de  $n \times m$  dimension de l'espace d'entrée:

$$x_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T \quad i = 1, 2, \dots, n \quad (\text{B. 1})$$

Les vecteurs des poids ont la même dimension que l'espace d'entrée.

Soit un vecteur poids d'un neurone  $w_j$ , noté :

$$w_j = [w_{j1}, w_{j2}, \dots, w_{jm}]^T \quad j = 1, 2, \dots, l \quad (\text{B. 2})$$

où  $l$  est le nombre total de neurones dans le réseau.

A l'instant de la présentation d'un échantillon  $x_i$  à l'entrée de la carte, le neurone gagnant  $g(x_i)$  correspondant est déterminé par la comparaison de cet échantillon à tous les vecteurs poids, le neurone gagnant est celui dont le vecteur poids est le plus proche. Alors le neurone  $g(x_i)$  est le neurone qui minimise la distance euclidienne entre  $x_i$  et les vecteurs poids  $w_j$  :

$$g(x_i) = \operatorname{argmin} \|x_i - w_j\|^T \quad j = 1, 2, \dots, l \text{ et } i = 1, 2, \dots, n \quad (\text{B. 3})$$

Une fois trouvé le vecteur de poids gagnant, le réseau s'ajuste de manière à nouveau. Selon Kohonen, il est nécessaire de chercher et modifier aussi l'ensemble des neurones actifs et qualifiés de coopérants, pour que les modifications ne concernent pas seulement le neurone gagnant, cet ensemble des neurones dit le voisinage correspond au neurone gagnant  $g(x_i)$

Généralement la fonction de voisinage  $v_{jg}(x_i)$  la plus employée est de type gaussienne, pour éviter que la carte ne passe, en cours d'apprentissage, par des états stables, alors que les vecteurs prototypes n'ont pas encore atteint leurs positions finales. Une telle situation peut amener un blocage.

La fonction de voisinage est donnée comme suit [ERW 91, FRI 95-1].:

$$v_{jg}(x_i)(t) = \exp \left[ -\frac{d_{jg}^2(x_i)}{2\sigma^2(t)} \right] \quad t = 0, 1, 2, \dots \quad (\text{B. 4})$$

Où  $d_{jg}$  est la distance entre le neurone vainqueur  $g(x_i)$  et le neurone voisin,  $\sigma$  correspond à la largeur effective du voisinage. Ainsi, quand  $t$  croît, la largeur du voisinage  $\sigma(t)$  décroît à un taux exponentiel.

L'intérêt d'avoir un voisinage conséquent est essentiel pour corrélérer la direction des modifications des poids d'un nombre important de neurones excités dans la couche de compétition. Quand la largeur du voisinage décroît, alors la corrélation du nombre de neurones qui modifient leurs poids est de plus en plus grande.

- **Phase d'adaptation des poids**

Cette phase consiste la mise à jour des neurones du voisinage du neurone gagnant, pour que le réseau s'auto-organise, cette mise à jour a pour effet de faire guider les vecteurs poids pour suivre la distribution des vecteurs d'entrées, du fait de la modification du voisinage. L'algorithme va donc conduire à un ordonnancement topologique de la carte.

Le vecteur des poids  $w_j$  du neurone  $j$  doit évoluer, en relation avec le vecteur d'entrée  $x_i$  par la relation suivante :

$$\begin{cases} w_j(t+1) = w_j(t) + \eta(t) \cdot v_{jg(x_i)}(t) (x_i - w_j(t)) & \text{si } j \in \text{voisinage} \\ w_j(t+1) = w_j(t) & \text{si } j \notin \text{voisinage} \end{cases} \quad (\text{B. 5})$$

Où  $\eta(t)$  est le taux d'apprentissage compris entre 0 et 1 ( $0 < \eta(t) < 1$ ), qui doit être positive, décroissante et monotone. Sa valeur est en général assez élevée pour les premières itérations et décroît progressivement vers zéro.

Généralement, la fonction de taux d'apprentissage est une fonction de décroissance linéaire qui permet d'avoir une bonne performance de l'algorithme. Cette forme est donnée par la relation suivant :

$$\eta(t) = \begin{cases} \eta_0 - \frac{(\eta_0 - \eta_\tau)}{\tau} t & \text{si } t < \tau \\ \eta_\tau & \text{sinon} \end{cases} \quad (\text{B. 6})$$

où  $\tau$  est le paramètre qui détermine la fin de la phase d'organisation et le début de la phase de convergence.  $\eta_0$  est le taux d'apprentissage initial, Selon *Kohonen.1995*,  $\eta_0$  doit prendre des valeurs proches de 1 pour les 1000 premières étapes puis décroître progressivement,  $\eta_\tau$  est le taux d'apprentissage en phase de convergence.

- **Phase de Continuation :**

Cette phase est une décision « terminer l'évolution du vecteur prototype ou continuer avec le retour de phase de compétition ».

Théoriquement, l'apprentissage du réseau s'arrête lorsqu'il n'y a plus de modifications à apporter aux prototypes

Pratiquement, il est préférable de stopper l'algorithme au début de la phase de stabilisation. Il est donc très important de déterminer à partir de combien d'itérations débute la phase de stabilisation. Selon Kohonen, le nombre de phases d'apprentissage ou d'échantillons à présenter doit être au moins égal à 500 fois le nombre de cellules constituant la carte pour obtenir une bonne précision statistique.

L'algorithme de Kohonen est comme suit :

---

### Algorithme

---

- **Initialisation des vecteurs des poids de la carte.**
- **Choisir le vecteur d'entrée.**
- **Recherche du neurone gagnant  $g(x_i)$  en utilisant le critère de la distance euclidienne minimale (neurone plus proche).**

$$g(x_i) = \operatorname{argmin} \|x_i - w_j\|^T$$

- **Modification du poids des neurones appartenant au voisinage topologique utilisant :**

$$w_j(t+1) = w_j(t) + \eta(t) \cdot v_{jg(x_i)}(t) (x_i - w_j(t))$$

- **Répéter si le changement des poids demeure non négligeable.**
- 

### Paramètres de la carte de Kohonen

Les résultats de l'algorithme varieront selon le choix des paramètres d'apprentissage suivants :

- Le nombre de neurones et la configuration de carte dépendront du type de problème posé. Si la configuration des données est simple, une structure linéaire du réseau (unidimensionnelle, 1-D) peut suffire. Il a été montré que, dans des cas plus

compliqués une carte bidimensionnelle (2-D) est préférable. (Définition des dimensions de la carte).

- La fonction de voisinage

Il est à noter que la taille du voisinage du prototype gagnant qui détermine les prototypes candidats au rapprochement. Alors, plus on avance dans la phase d'apprentissage plus la taille du voisinage se réduit.

- Le pas de l'apprentissage contrôle la vitesse d'apprentissage. Si il est trop petit, le modèle ne s'adaptera pas assez aux données, si il est trop grand, il y aura risque d'instabilité.

- Condition de convergence

Le choix des valeurs de la largeur effective du voisinage et le pas de l'apprentissage assez grandes au départ et les réduire graduellement.

Comme indiqué dans la liste des paramètres, plusieurs paramètres doivent être choisis correctement pour obtenir la convergence désirée. Comme pour la plupart des autres techniques des réseaux de neurones, la technique SOM est une science d'une part et un art de l'autre. Tous les paramètres dépendront du type de problème réel et doivent être choisis par expérience.