

UNIVERSITE SAAD DAHLAB DE BLIDA

Faculté des Sciences de l'Ingénieur
Département d'Electronique

MEMOIRE DE MAGISTER

Spécialité: Electronique
Option: signaux et systèmes

**MODELISATION DES ACTIONS DE CONTRÔLE DE LA
QUANTITE DE PAC DANS UN PROCESSUS
D'EPURATION DES EAUX**

Par

TIDJANI Nawel

Devant le jury composé de

A.GUESSOUM	Professeur, U. de Blida	Président
H.SALHI	Maître de Conférences A, U. de Blida	Examineur
K.BENMANSOUR	Maître de Conférences A, U. de Médéa	Examineur
M.HADJ SADOK	Maître de Conférences A, U. de Blida	Rapporteur

Blida, juin 2010

RESUME

Les eaux de surface sont rarement potables et contiennent de nombreux composés (matières en suspension, colloïdes ...) qui sont à l'origine de la turbidité, la couleur, voire la toxicité de ces eaux. Les coagulants et flocculants sont utilisés en traitement des eaux, pour rassembler ces particules et colloïdes dans le but d'augmenter leur taille et faciliter ainsi leur séparation.

Le procédé physico-chimique le plus utilisé dans un processus d'épuration des eaux reste, sans conteste, la technique séparative par coagulation floculation. L'eau de la rivière, après dégrillage et ajustement du pH, arrive au niveau du réacteur de précipitation où un coagulant (du polychlorure basique d'aluminium (PAC)) est introduit. L'opération de mélange est critique, et la quantité de PAC déversée par l'opérateur, doit être déterminée de manière à ce que la turbidité de l'eau traitée soit maintenue au dessous d'un certain seuil de tolérance.

Dans les stations de traitement des eaux, on utilise fréquemment, un modèle statique représenté par une formule empirique pour la détermination de la quantité de PAC à ajouter à l'eau non traité, les résultats obtenus sont en générale acceptables mais pas optimales.

On se propose dans cette contribution, de modéliser les décisions d'un habile et expérimenté opérateur humain, pour la détermination des quantités du composé chimique PAC à ajouter à l'eau de rivière afin d'optimiser le rendement du processus (turbidité de l'eau traitée). On utilisera, pour ce travail, les techniques de modélisation floue par la méthode de Sugeno et des relevés expérimentaux sur une durée de cinquante jours.

Mots clés: traitement des eaux, modélisation ARMA, réseaux de neurones, logique floue type Takagi-Sugeno.

ملخص

في هذا العمل نقوم بعرض خوارزمية التعريف و وضع ثلاثة نماذج للأنظمة الفيزيائية غير الخطية و هي:

نمذجة ARMA.

نمذجة الشبكة العصبونية.

النمذجة الغامضة بطريقة تاكاجي- سوجينو التي تمثل قواعد مداخل_مخارج للنظام. النتيجة في هذه القواعد توضع على شكل تمثيل للحالة.

تستعمل هذه النماذج بالاستناد إلى معطيات سلوكيات التحكم من طرف خبير, خلال سير عملية تنقية المياه السطحية, حيث يجب على العامل اليدوي إضافة كمية من مادة كيميائية, و ذلك لتثبيت درجة تعكر الماء تحت مستوى حرج معين.

الكلمات الرئيسية: تنقية المياه, نمذجة ARMA , نمذجة الشبكة العصبونية, النمذجة الغامضة تاكاجي - سوجينو .

ABSTRACT

In the water purification plant, raw water is promptly purified by injecting chemicals. The amount of chemicals is directly related to the water quality such as the turbidity of raw water.

At present, however, the process of the chemical reaction to the turbidity has not been clarified yet, so the amount of chemical cannot be calculated from the data of water quality only. Accordingly, it has been judged and determined by skilled operators according to the data obtained from past experience.

In connection with this, we studied three methods to determine the amount of chemicals automatically by reproducing the judgments of the operators by means of:

- ARMA model (auto regressive moving average)
- Neural network model
- Fuzzy implication Takagi Sugeno, and performed a field test in the water purification plant using a fuzzy controller.

Key words: water purification, ARMA modelisation, neural network, fuzzy logic Takagi-Sugeno.

REMERCIEMENTS

Je remercie tout d'abord Mr M.Hadj sadok de m'avoir proposé ce sujet, de m'avoir initiée à la recherche, en me guidant pour réaliser ce modeste travail.

Je remercie aussi les membres du jury, d'avoir accepté d'examiner mon travail.

Mes vifs remerciements à tous mes collègues, à tous mes amis, ainsi que tous ceux qui m'ont encouragé et soutenus.

DEDICACE

A mon fils Ahmed Yacine

TABLE DES MATIERES

RESUME

REMERCIEMENTS

DEDICACE

TABLE DES MATIERES

LISTE DES ILLUSTRATIONS GRAPHIQUES ET TABLEAUX

INTRODUCTION	13
1. INTRODUCTION AUX PROCEDES PHYSICO-CHIMIQUES DE TRAITEMENT DES EAUX	13
1.1 Introduction	17
1.2 L'eau de surface	17
1.2.1 Eau de rivière (partie amont)	18
1.2.2 Eau de rivière (partie aval)	18
1.3 Les procédés de traitement des eaux usées	18
1.3.1 Les différentes pollutions	18
1.3.2 Caractéristiques des matières polluantes	19
1.4 Le traitement des eaux usées	20
1.5 Les différents traitements du processus d'épuration d'eau	20
1.6 Les paramètres mesurables pour évaluer la qualité de l'eau	23
1.6.1 La température	24
1.6.2 Le pH	24
1.6.3 L'alcalinité	24
1.6.4 La turbidité	25
1.7 Traitement physico-chimiques des eaux usées	25
1.7.1 Élimination de la pollution colloïdale	25
1.7.2 La coagulation	25
1.8 Coagulants utilisés	26
1.9 L'effet des différents paramètres sur la coagulation	27

1.9.1 Effet de la température	27
1.9.2 Effet du pH	28
1.9.3 Effet de la turbidité	28
1.10 La floculation	29
1.11 La décantation	29
1.12 Conclusion	30
2. MODELISATION DU PROCESSUS D'EPURATION DE L'EAU	
2.1 Introduction	28
2.2 Modélisation du modèle ARMA	29
2.3 Identification paramétriques du modèle ARMA	30
2.4 Estimation des paramètres du modèle ARMA	31
2.4.1 Méthode globale des moindres carrés	31
2.4.2 Méthode des moindres carrés récurrents (MCR)	35
2.5 Application au processus d'épuration de l'eau	37
2.6. Modélisation du processus d'épuration de l'eau	37
2.7 Modèle linéaire	40
2.8 Modélisation statique	47
2.9 Conclusion	51
3. MODELISATION NEURONALE	
3.1 Introduction	52
3.2 Propriétés des neurones biologiques	53
3.3 Les neurones formels	54
3.4 Fonction d'activation	56
3.5 Les réseaux de neurones artificiels	56
3.5.1 Réseaux de neurones bouclés	57
3.5.2 Réseaux de neurones non bouclés	57
3.6 Apprentissage	58
3.6.1 Apprentissage non supervisé	58
3.6.2 Apprentissage supervisé	59
3.7 L'algorithme de rétropropagation :	60
3.8 Application des réseaux de neurones à la modélisation des actions de contrôle de l'opérateur dans la station d'épuration de l'eau	63

3.8.1	Introduction	63
3.8.2	Acquisition de données d'entrée/sortie du réseau	64
3.9	Mise en œuvre du réseau de neurones perceptron multicouche :	64
3.10	Paramétrage et test du réseau I	65
3.11	Application du réseau II	71
3.12	Modélisation neuronale du processus d'épuration	73
3.13	La mise en cascade	79
3.14	Conclusion	82
4.	IDENTIFICATION DE MODELES FLOUS PAR LA METHODE DE TAKAGI SUGENO (TS) A PARTIR DE DONNEES D'ENTREE/SORTIE	
4.1	Introduction	83
4.2	Techniques de modélisation floue et représentation du contrôleur flou	84
4.3	Structure générale et différents types de modèles flous	85
4.3.1	Contrôleur de type Mamdani	87
4.3.2	Contrôleur de type Takagi-Sugeno	87
4.4	Modélisation floue type Takagi-Sugeno	88
4.5	Algorithme d'identification	88
4.5.1	Identification de la structure du modèle flou type Takagi Sugeno	89
4.5.2	Identification des paramètres du modèle flou	91
4.6	Application à la modélisation floue	95
4.6.1	Application au processus d'épuration des eaux	95
4.6.2	Résultats de la modélisation floue	108
4.7	Résultats et discussions	113
4.8	Conclusion	115
	CONCLUSION	116

LISTE DES ILLUSTRATIONS GRAPHIQUES ET TABLEAUX

Figure 1.1	Etapas d'une filière d'épuration des eaux	21
Figure 2.1	Modèle du processus d'épuration	32
Figure 2.2	Schéma bloc de la représentation temporelle du modèle ARMA	34
Figure 2.3	Représentation spectrale du modèle ARMA	37
Figure 2.4	Processus d'épuration d'eau	40
Figure 2.5	Diagramme de processus de régulation	41
Tableau 2.1	Relevés expérimentaux	43
Figure 2.6	Evolution des paramètres du modèle linéaire	44
Figure 2.7	Turbidité estimée par le modèle et celle donnée par l'expert	46
Figure 2.8	Erreur en moyenne entre turbidité expert et celle du modèle linéaire	46
Figure 2.9.	PAC régulateur et PAC expert pour un seuil de tolérance égal à la valeur 3	47
Tableau 2.2	Valeur du PAC suivant la consigne	49
Figure 2.10	Turbidité due au PAC modèle et celle limitée par le seuil de tolérance	50
Figure 2.11	PAC limité suivant un seuil de tolérance imposé égal à 3.5	51
Figure 2.12	Limitation de la turbidité par la valeur 1.5	51
Figure 2.13	PAC limité suivant un seuil de tolérance imposé égale à 1.5	52

Figure 2.14	PAC limite en fonction de la turbidité limitée	52
Figure 3.1	Schéma d'un neurone biologique	56
Figure 3.2	Schéma d'un neurone formel	57
Figure 3.3	Différents types de fonction d'activation	58
Figure 3.4	Apprentissage non supervisé	60
Figure 3.5	Apprentissage supervisé	61
Figure 3.6	Perceptron multicouches de type '2-3-1'	67
Figure 3.7	Initialisation aléatoire des poids et des biais	69
Figure 3.8	Evolution et convergence des poids et des biais	70
Figure 3.9	Sortie normalisée	71
Figure 3.10	PAC réseau I et PAC expert	71
Figure 3.11	Perceptron multicouches de type '3-5-1'	72
Figure 3.12	Variance de l'erreur PAC expert-PAC réseau II=f (nombre de neurones de la couche cachée)	73
Figure 3.13	Moyenne de l'erreur entre PAC Modèle flou, PAC modèle neuronal, PAC statique et PAC expert	74
Figure 3.14	PAC statique et celui issu du réseau II, réseau I par rapport au PAC Expert	74
Figure 3.15	Perceptron multicouches de type '4-4-1'	75
Figure 3.16	Initialisation aléatoire des poids W et Z et des biais w_0 et z_0	76
Figure 3.17	Evolution des poids et des biais	77
Figure 3.18	Erreur de la turbidité du réseau par rapport à celle donnée par l'expert	78
Figure 3.19	Turbidité donnée par l'expert et celle donnée par le réseau	78
		79
Tableau 3.1	Indice de performance et variances	
Figure 3.20	tracé des courbes de turbidité	79
Figure 3.21	Mise en cascade du réseau d'expert et du réseau du processus d'épuration.	80

Figure 3.22	Schéma boîte noire de la mise en cascade	81
Figure 3.23	Confrontation des turbidités (mise en cascade).	82
Figure 4.1	Représentation interne d'un contrôleur flou	85
Figure 4.2	Choix des variables linguistiques	92
Figure 4.3	Fonction d'appartenance de la variable PH	97
Tableau 4.1	Indice de performance par modèle	98
Figure 4.4	Sortie du modèle flou avec la variable (TE) dans la prémisses	99
Figure 4.5	Comparaison des erreurs du modèle flou et celui du modèle statique avec la variable (TE) dans la prémisses	99
Figure 4.6	Fonction d'appartenance de la variable TE divisé en 4 sous espaces flous	100
Tableau 4.2	Indices de performance par modèle	101
Figure 4.7	Fonctions d'appartenance des variables (TE, PH) dans l'antécédent.	102
Figure 4.8	Sortie du modèle flou avec les variables (TE, PH) dans la prémisses donnant $I_p=41.86$	103
Figure 4.9	Comparaison des erreurs du modèle flou et celui du modèle statique avec la variable (TE, PH) dans la prémisses	103
Tableau 4.3	Indice de performance par modèle	105
Figure 4.10	Fonction d'appartenance des variables TE, PH et AL	105
Figure 4.11	Sortie du modèle flou avec les variables (TE, PH, AL) dans la prémisses donnant $I_p=18.08$.	106
Figure 4.12	Comparaison des erreurs du modèle flou et celui du modèle statique avec les variables (TE, PH, AL) dans la prémisses avec $I_p=18.08$	106
Figure 4.13	Fonctions d'appartenances	107
Figure 4.14	Sortie du modèle flou avec les variables (TE, PH, AL) dans la prémisses donnant $I_p=14.3$	107

Figure 4.15	Comparaison des erreurs du modèle flou et celui du modèle statique avec les variables (TE, PH, AL) dans la prémisse avec $I_p=14.3$	108
Figure 4.16	Fonction d'appartenance	108
Figure 4.17	Sortie du modèle flou avec les variables (TE, PH, AL) dans la prémisse donnant $I_p=4.89$	109
Tableau 4.4	Illustration des résultats flous, statiques et mesurés	111
Figure 4.18	PAC statique, PAC modèle flou et PAC de l'expert.	111
Figure 4.19	Erreur PAC expert/ PAC modèle flou et celle PAC expert/PAC statique	112
Figure 4.20	Architecture du processus d'épuration de l'eau	113
Figure 4.21	Modèle neuro-flou	114

INTRODUCTION

L'eau est à l'origine de la vie, très largement répandue à la surface de la terre et jouant un rôle essentiel dans la structure organique des êtres vivants et des végétaux.

Sous l'effet de la chaleur, l'eau des océans, des rivières et des lacs s'évapore et monte dans le ciel et au contact des couches d'air froid de l'atmosphère, la vapeur d'eau se condense en minuscules gouttelettes qui, à la rencontre d'une masse d'air plus froid, forment des nuages. Elles se chargent de poussière atmosphérique et deviennent trop lourdes pour demeurer en suspension dans l'atmosphère et tombent sous forme de pluie.

Une fois au sol, l'eau peut s'infiltrer et former des nappes souterraines ainsi que par ruissellement, elle sera acheminée vers les rivières, fleuves pour finir à la mer. Durant ce trajet, l'eau entraîne tous les déchets rencontrés, mais une fois arrivée à la mer ou aux rivières elle transforme et élimine naturellement certaines pollutions; c'est ce qu'on appelle l'autoépuration grâce aux phénomènes de filtration et d'oxydation, combinés à l'action des organismes vivant dans le milieu aquatique : bactéries, protozoaires, algues, qui permettent à l'eau de retrouver sa qualité première et préserver l'équilibre de son écosystème.

Mais au fil des années, l'accroissement de la population et l'augmentation du nombre d'industries, qui favorise la concentration en substances polluantes, ont dégradé le cycle de l'eau qui devient incapable de libérer les impuretés par autoépuration.

Ces substances peuvent avoir des origines polluantes diverses : physique (présence des matières en suspension), chimique, organique et bactériologique.

Pour faire face à ce problème, l'homme a mis en place des réseaux d'égout qui collectent les eaux usées et les acheminent vers les stations d'épuration, où le traitement des effluents se fait à l'aide de techniques efficaces, afin de minimiser l'effet polluant avant que l'eau soit rejetée dans le milieu naturel.

L'épuration d'un effluent pollué peut comporter des traitements assez variés: biologiques, chimiques et/ou physico-chimiques.

Les traitements physico-chimiques sont utilisés dans l'épuration des eaux, en rassemblant les particules et colloïdes dans le but d'augmenter leur taille et faciliter ainsi leur séparation. On reviendra dans le chapitre 1 plus en détail sur le traitement physico-chimique des eaux usées.

Le procédé physico-chimique le plus utilisé dans un processus d'épuration des eaux reste, sans conteste, la technique séparative par coagulation floculation. L'eau de la rivière, après dégrillage et ajustement du pH, arrive au niveau du réacteur de précipitation où un coagulant (du polychlorure basique d'aluminium, le PAC) est introduit. L'opération de mélange est critique, et la quantité de PAC déversé par l'opérateur, doit être déterminée de manière à ce que la turbidité de l'eau traitée soit maintenue au dessous d'un certain seuil de tolérance.

Dans les stations de traitement des eaux, on utilise fréquemment, un modèle statique représenté par une formule empirique pour la détermination de la quantité de coagulant à ajouter à l'eau pollué, les résultats obtenus sont, en général, acceptables mais pas optimaux.

Le processus de station d'épuration des eaux, est un système dynamique. Les lois physiques décrivant le système sont trop complexes et régies par des paramètres variables qui peuvent changer selon plusieurs critères.

Le problème apparaît lorsque la quantité de coagulant (PAC) déversé dans le réservoir n'est pas optimal, de ce fait l'agglomération des impuretés, sera mal faite et la turbidité de l'eau traitée, mauvaise.

Il est alors nécessaire d'avoir un modèle adéquat qui permettra de surmonter ce problème et qui représentera d'une manière optimale le système d'épuration de l'eau.

Dans le domaine du traitement des eaux, l'amélioration de la performance dans les industries de production, ainsi que les nouvelles normes législatives ont rendu nécessaire la modélisation, l'identification et la commande en temps réel du procédé du traitement physico-chimique.

Ce procédé repose sur la mesure, l'analyse et le contrôle des variables caractérisant les effluents.

Notre travail consiste en une contribution à la modélisation des décisions d'un opérateur humain habile et expérimenté, pour la détermination des quantités du composé chimique PAC à ajouter à l'eau de surface afin d'optimiser le rendement du processus (turbidité de l'eau après le traitement).

Le présent mémoire est constitué de quatre chapitres:

Le premier chapitre est une introduction aux procédés de traitement physico-chimique des eaux de surface. Le problème de pollution des effluents y est décrit, ainsi qu'une présentation des concepts de base sur les paramètres mesurables servant à l'évaluation de la qualité de l'eau.

Le deuxième chapitre aborde les techniques d'identification des paramètres du système autorégressif ARMA pour la modélisation du processus d'épuration de l'eau.

Dans le troisième chapitre une modélisation du même système par les réseaux de neurones, est présentée. Deux réseaux de neurones sont utilisés, le premier modélise les actions de contrôle de la quantité de PAC à ajouter d'un opérateur humain habile, le deuxième, modélise la station d'épuration des eaux de surface,

Dans le quatrième chapitre, les techniques de modélisation floue sont utilisées par la méthode de Takagi-Sugeno, pour essayer d'améliorer le modèle général de la station obtenue dans le troisième chapitre.

Dans cette contribution, le réseau de neurone représentant la station sera mis en cascade avec un modèle flou simulant les décisions d'un opérateur humain habile et expérimenté pour la détermination des quantités de PAC à ajouter aux eaux non traitées pour l'obtention d'une turbidité optimale. On verra que les résultats obtenus sont améliorés d'une manière très significative. Ce qui justifiera notre conclusion.

CHAPITRE 1

Introduction aux procédés physico-chimiques de traitement des eaux

Ce chapitre est consacré à quelques notions de procédés de traitement des eaux. Techniques à partir desquelles est positionnée la problématique de pollution de l'eau. Le traitement physico-chimique des eaux usées est l'une des filières majeures.

1.1 Introduction

L'eau est indispensable à la vie humaine, végétale, industrielle, ou agricole, qui requièrent en abondance une bonne qualité de l'eau et malheureusement engendrent à travers ces diverses utilisations une pollution.

A cause des problèmes d'environnement liés à l'élévation de la population ainsi qu'au développement des activités industrielles, le phénomène naturel d'autoépuration ne peut pas à lui seul réaliser une diminution significative de la concentration des rejets d'effluents polluants; la qualité de l'eau s'est dégradée et par conséquent il s'en est suivi la perturbation de tout l'écosystème.

De ce fait, le traitement des effluents dans les stations d'épuration est réalisé sous forme d'un ensemble de techniques afin de compléter les cycles naturels d'autoépuration.

Ce chapitre, présente quelques concepts fondamentaux liés à la pollution et les techniques utilisées pour le traitement d'une eau destinée à la consommation humaine.

1.2 L'eau de surface

L'eau de surface est répartie en trois catégories : eau de rivière (partie amont), eau de rivière (partie aval) et eau de lac.

1.2.1 Eau de rivière (partie amont)

L'amont d'une rivière est en général situé dans une région montagneuse, où la densité de population est faible et les industries pratiquement inexistantes. Les principales caractéristiques de ces eaux sont présentées ci-dessous :

- ✓ Turbidité élevée. Le régime des rivières étant torrentiel les eaux transportent de grandes quantités de matières en suspension.
- ✓ Température froide. Ces eaux proviennent soit de sources, soit de la fonte des neiges et des glaciers.

1.2.2 Eau de rivière (partie aval)

L'aval d'une rivière est en général situé dans une région où la population est dense, les eaux sont de moins bonne qualité et plus difficile à traiter.

Les principales caractéristiques de ces eaux sont présentées ci-dessous :

- ✓ Contamination bactérienne élevée.
- ✓ Contamination organique et inorganique élevée.

1.3 Les procédés de traitement des eaux usées

Cette section parcourt les éléments de base de traitement des eaux usées ; en particulier ceux de la filière physico-chimique.

1.3.1 Les différentes pollutions

La pollution engendre la dégradation de la qualité de l'eau et peut se présenter à l'origine sous différentes formes relatives aux activités humaines.

- La pollution urbaine et d'origine domestique :

L'eau qui provient des vannes et celle utilisée pour les besoins domestiques, rejetée sous forme usée comportant des détergents, graisses, matière organique, apporte des quantités de matières en suspension et les micropolluants.

- La pollution industrielle :

Toute activité industrielle engendre des rejets polluants provenant des matières organiques, des métaux, les gaz acides toxiques des fumées rejetées par les usines et qui retombent avec la pluie, cette pollution a un rapport avec la spécificité des activités industrielles.

- La pollution agricole :

Pour le traitement des cultures, les spécialistes (agronomes) utilisent des engrais qui apportent aux végétaux les nutriments nécessaires à leur croissance, et des pesticides qui sont des produits chimiques.

Par ruissellement, l'eau de pluie et d'arrosage transporte ces produits vers les eaux de surfaces.

1.3.2 Caractéristiques des matières polluantes

1.3.2.1 Matières en suspension

Ce sont des particules solides très fines qui peuvent être visibles à l'œil nu. Théoriquement, elles déterminent la turbidité de l'eau et limitent la pénétration de la lumière diminuant ainsi la teneur en oxygène dissous et nuisent au développement de la vie aquatique.

1.3.2.2 Matières colloïdales

Ce sont des substances en forme de colle ou de gel qui ont la capacité de rester en suspension. Cette caractéristique est due à leur structure moléculaire.

La décantation naturelle de ces matières est extrêmement lente. Il faut donc empêcher que ces matières soient solubles en utilisant un produit chimique pour faciliter leur élimination par décantation.

1.4 Le traitement des eaux usées

L'eau de surface provenant de plusieurs origines est polluée donc impropre à la consommation.

Pour pouvoir l'utiliser, l'eau doit subir un traitement pour la clarifier par l'élimination des matières en suspension, ainsi que des matières colloïdales et des macromolécules susceptibles de communiquer à l'eau une turbidité ou une couleur indésirable.

1.5 Les différents traitements du processus d'épuration d'eau

L'eau est le véhicule de transport et de dissémination idéal de nombreux polluants. Les contraintes d'assainissement, de plus en plus strictes, exigent le traitement d'un nombre plus important de polluants.

Dans les stations d'épuration de l'eau, la production d'une eau potable est produite à partir d'une eau brute plus ou moins polluée.

Pour ce faire, l'eau brute est soumise à diverses étapes de traitement réalisées dans plusieurs unités de station d'épuration de l'eau.

L'eau polluée doit donc subir divers traitements par un enchaînement successif.

Le schéma fonctionnel de la filière d'épuration des eaux [1] est indiqué sur la figure 1.1.

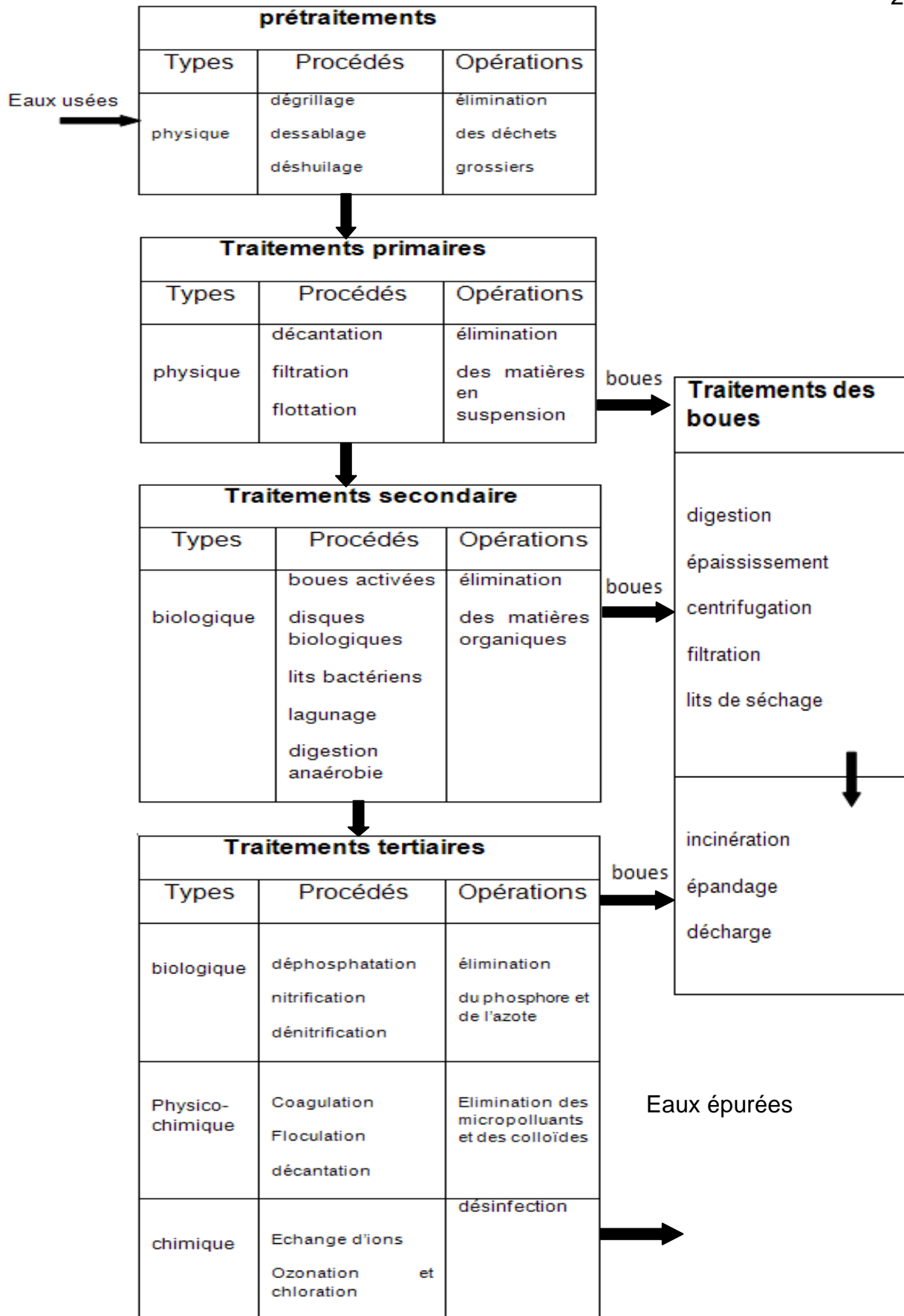


Figure 1.1 Etapes d'une filière d'épuration des eaux

Les procédés de traitement des eaux qui recueillent des eaux usées sont composés de plusieurs phases [2] [3] [4] [5]. L'épuration d'un effluent pollué peut comporter cinq phases principales:

1.5.1 Le prétraitement

Cette phase élimine les gros déchets insolubles, et cela se fait en passant par trois étapes successives. Parmi ces méthodes de séparation [5], il y a :

- Le dégrillage, il s'agit de séparer des eaux brutes, les matières les plus volumineuses, en faisant passer l'effluent d'entrée à travers des barreaux dont l'espacement est déterminé en fonction de la nature de l'effluent.
- Le dessablage, l'élimination des sables présents dans l'effluent brut est indispensable si on veut protéger les conduites et pompes contre la corrosion et aussi éviter le colmatage des canalisations par une sédimentation au cours du traitement.
- Le déshuilage dégraissage, c'est une opération destinée à réduire les graisses et huiles non émulsionnées par simple sédimentation physique en surface et qui sera éliminée par un raclage.
- La décantation, l'élimination des matières en suspension présentes dans le milieu liquide est réalisée par sédimentation, en utilisant uniquement les forces de gravité.

1.5.2 Le traitement primaire

Ce traitement a pour but d'extraire le maximum de matières en suspension et de matières organiques qui peuvent se décanter sous forme de boues.

Trois voies de traitement sont possibles : décantation, filtration et flottation.

1.5.3 Le traitement secondaire

Le traitement secondaire se fait par voies biologiques et a pour objectif principal l'ajout d'une quantité adéquate d'oxygène à l'eau qui était déjà dégradée

par la présence des composés solubles d'origine organique, pour préserver les animaux aquatiques.

1.5.4 Le traitement tertiaire

C'est un traitement complémentaire qui permet d'affiner les traitements précédents, et qui nécessite trois phases successives:

Un traitement biologique, qui permet d'éliminer le phosphore et l'azote.

Un traitement physico-chimique pour permettre la coagulation, ainsi que l'élimination des matières en suspension.

Un traitement chimique pour la désinfection des effluents.

1.5.5 Le traitement des boues

Il a pour but le traitement et le conditionnement des boues résiduelles extraites à la suite de la phase de décantation.

1.6 Les paramètres mesurables pour évaluer la qualité de l'eau

Les rejets dans les milieux naturels entraînent des déchets qui vont perturber le fonctionnement écologique et rendre l'eau trouble. Un traitement, par l'ajout d'un coagulant est nécessaire. La quantité de ce dernier est déterminée en fonction des mesures de certaines grandeurs responsables de la turbidité de l'eau.

Ces grandeurs évoluent et varient en fonction des phénomènes de pollution, et aussi de saison en saison, et donc l'opération d'ajout du coagulant est critique et la décision est délicate. Parmi ces grandeurs on trouve :

- La température,
- Le pH (potentiel hydrogène),
- L'alcalinité,
- La turbidité.

1.6.1 La température

Il est important de connaître la température de l'eau avec une bonne précision. Sachant que la température des eaux dépend de celle de l'air, de sa couleur ; une eau sombre absorbe plus fortement la chaleur.

Les rejets domestiques et industriels élèvent généralement la température de l'eau. La mesure de la température est effectuée sur le terrain, en général faite à l'aide de plusieurs capteurs placé en différents endroits du bassin.

1.6.2 Le pH

L'eau contient des particules ionisées, qui résultent de la dissociation ionique de l'eau :

- Les premières sont chargées positivement, par exemple H^+ : ion hydrogène (fonction acide).
- Les secondes sont chargées négativement, par exemple OH^- : ion alcalin (fonction basique).



La dissociation augmente avec l'élévation de la température, et donc le pH diminue. De ce fait il est indispensable de mesurer la concentration des ions de l'eau, ce qui impose de prendre des mesures du pH.

Le pH est mesuré selon une échelle qui va de 0 à 14

Une mesure inférieure à 7 signifie que l'eau est acide.

Une mesure supérieure à 7 désigne une eau basique, ou alcaline.

1.6.3 L'alcalinité

C'est la mesure de la capacité d'une eau à neutraliser les acides, c'est à dire maintenir un pH constant **[6]**.

1.6.4 La turbidité

Les matières dissoutes dans l'eau, telles que les algues, la boue, les microbes ou autres particules, absorbent et diffusent la lumière qui les traverse. Plus les particules sont nombreuses, plus le degré de turbidité augmente.

1.7 Traitement physico-chimique des eaux usées

1.7.1 Élimination de la pollution colloïdale

Ce traitement consiste à enlever la stabilité aux matières colloïdales présentes dans l'eau en provoquant la neutralisation de leurs charges électriques superficielles. L'agglomération de ces particules permet de former un flocon volumineux qui peut se décarter facilement. Ce flocon est appelé floc.

La première étape est appelée coagulation et se réalise dans le traitement d'eau par l'utilisation d'un produit chimique appelé polychlorure basic d'aluminium.

La deuxième étape est la floculation.

L'intérêt de ce travail se base sur la première étape, pour introduire la quantité nécessaire qu'il faut ajouter à l'eau, en imitant les actions de l'opérateur humain.

1.7.2 La coagulation

La turbidité et la couleur d'une eau sont principalement causées par des particules très petites, minérales (argiles), végétales (acides humiques, algues), animales (zooplancton), humaines (suspensions graisseuses, rejets) ou industrielles.

Ces particules sont dites colloïdales, leur élimination ne peut se baser sur la simple décantation, car leur vitesse de sédimentation est extrêmement faible. Le temps nécessaire pour parcourir un mètre en chute libre peut prendre plusieurs années [7].

La coagulation et la floculation sont les processus qui permettent l'élimination des colloïdes.

Dans une eau de surface, les particules possèdent habituellement une charge électrique négative située à leur surface qui attire des ions négatifs accompagnés d'une faible quantité d'ions positifs [7].

La charge électrique et la couche d'eau qui entourent les particules tendent à éloigner les particules les unes des autres et par conséquent à les stabiliser dans la solution.

La coagulation a pour but principal de déstabiliser ces particules pour favoriser leur agglomération.

1.8 Coagulants utilisés

Les principaux coagulants utilisés pour déstabiliser les particules et pour produire un floc sont à base d'aluminium et de fer [8]. La réaction des sels de ces métaux avec l'eau engendre un ensemble de produits complexes qui, par différents moyens déstabilisent les particules colloïdales et favorisent leur agglomération.

Tout coagulant mis en solution consomme de l'alcalinité, cette perte est fonction de la nature et de la concentration du composé mis en jeu.

Cette perte d'alcalinité se traduit par une chute de pH, donc par une acidification de l'eau [6].

Parmi ces coagulants il y a le polychlorure basique d'aluminium (PAC) qui est fabriqué à partir de chlorure d'aluminium et de soude.

1.9 L'effet des différents paramètres sur la coagulation

La coagulation est l'étape la plus essentielle et la plus délicate dans la chaîne de traitement des eaux, la détermination de la quantité à ajouter à l'eau usée est critique afin d'avoir une eau pure.

Les coagulants agissent en créant une réaction chimique et en éliminant les charges négatives qui font que les particules se repoussent entre elles.

Le mélange coagulant-eau est remué lentement et la floculation a lieu. Le fait de battre l'eau favorise la coagulation de minuscules particules en suspension et forme des grumeaux plus grands et plus faciles à retirer, les «flocs».

Pour cela il est nécessaire de connaître la quantité qu'il faut mettre par rapport aux différents paramètres qui évaluent la qualité de l'eau, tel que la température, le pH, l'alcalinité, ainsi que la turbidité qui nécessite la présence d'un contrôleur afin de bien l'évaluer.

1.9.1 Effet de la température

La variation de la température de l'eau entraîne plusieurs situations dont lesquelles la quantité du coagulant nécessaire est différente. Toutefois, l'expérience révèle que durant l'hiver lorsque la température de l'eau est très faible, le floc se décante plus difficilement et a tendance à pénétrer plus profondément dans les filtres.

Une diminution de la température de l'eau entraîne une augmentation de sa viscosité et rétrécit la plage optimale du pH, ce qui explique la difficulté de décantation du floc. Alors il faut augmenter la quantité du coagulant, ce qui a pour effet d'accroître la probabilité de collision entre les particules et d'élargir la plage optimale du pH.

1.9.2 Effet du pH

Le pH est la variable la plus importante à prendre en considération au moment de la coagulation. Pour chaque type d'eau, il existe une plage de pH pour laquelle, la coagulation aura lieu rapidement, selon le coagulant utilisé, de sa concentration et de la composition de l'eau à traiter. Lorsque la coagulation se produit hors de cette plage optimale, il faut, soit augmenter la quantité du coagulant, soit se contenter d'une eau de moins bonne qualité. Pour les sels d'aluminium et de fer, les plages de pH optimales s'étendent respectivement de 5 à 7 et de 4 à 6.

1.9.3 Effet de la turbidité

La turbidité provient de la présence des particules colloïdales de glaise, qui sont de grosses particules qui peuvent se déposer naturellement. Le diamètre de ces particules est de l'ordre de $1\mu m$. La coagulation de ces particules est assez facile lorsqu'on maintient le pH dans sa plage optimale [7].

Il faut ajouter une concentration minimale de coagulant pour obtenir une certaine quantité de floc.

Lorsque la turbidité augmente, il faut augmenter la concentration du coagulant.

Lorsque la turbidité est très élevée, la quantité de coagulant nécessaire est relativement faible, car la probabilité de collision entre les particules est très élevée, ce qui aidera la sédimentation des floes.

Par contre, lorsque la turbidité est faible. La coagulation est difficile.

Lorsque la turbidité est due à des particules de diamètres variés, il est plus facile d'effectuer la coagulation que lorsqu'elle est imputable à des particules de diamètres sensiblement égaux.

La coagulation d'une faible concentration de particules de glaise dépend principalement du coagulant et de la composition chimique de l'eau.

1.10 La floculation

Après une coagulation des particules colloïdales, celles-ci ont tendance à s'agglomérer entre elles les unes des autres. Le taux d'agglomération dépend de la probabilité des contacts.

La floculation a pour but d'augmenter la probabilité des contacts entre les particules, et ceci par agitation de l'eau.

Cette agitation facilite l'agrégation en augmentant la probabilité des collisions entre les particules [9].

1.11 La décantation

La décantation est la méthode de séparation la plus fréquente des matières en suspensions et colloïdes (rassemblés sous forme de floc après l'étape de coagulation –floculation) [9].

Le bassin de décantation est de forme rectangulaire ou circulaire dans lequel l'eau reste le temps qu'il faut pour sédimenter les flocons et ceci selon la capacité du réservoir. Ce procédé sépare l'eau du flocon sous l'effet de la pesanteur. Par la suite les particules qui s'accumulent au fond du bassin de décantation sont extraites périodiquement pour prévenir l'accumulation.

L'eau sera ensuite acheminée vers un autre bassin afin de filtrer les petites particules.

1.12 Conclusion

Le traitement des eaux usées est un domaine vaste, qui englobe plusieurs disciplines, comme la microbiologie, le génie des procédés et l'automatique. Les notions de base des différentes filières pour le traitement des eaux ont été décrites dans ce premier chapitre. Le travail effectué repose particulièrement sur la chaîne de traitement physico chimique, qui est une phase difficile à maîtriser.

Le manque d'informations et d'outils de supervision efficaces rend la tâche de suivi encore plus délicate pour l'opérateur qui doit alors faire appel à son expérience personnelle. Cependant, l'opérateur dispose de nombreuses informations issues du procédé lui-même et de son environnement, et à la fin prendre la décision adéquate.

En absence de l'opérateur, la formule empirique utilisée dans les stations d'épuration actuelles, ne prend pas en compte tous les paramètres pour faire une modélisation satisfaisante. Ceci nécessite l'introduction d'un autre modèle pour identifier le processus d'épuration de l'eau.

Le chapitre suivant sera consacré à la première méthode de modélisation : la modélisation ARMA (Auto-Regressive à Moyenne Ajustée).

Chapitre 2

Modélisation du processus d'épuration de l'eau

2.1 Introduction

La plupart des systèmes réels sont complexes et non linéaires. Il est toutefois nécessaire de comprendre le comportement naturel du système réel si l'on veut, lui imposer une autre conduite répondant à des performances souhaitées par l'utilisateur.

La modélisation des systèmes, consiste justement à déterminer le système de commande automatique assurant le bon fonctionnement du procédé étudié.

Selon la connaissance des phénomènes physiques, mis en jeu, deux possibilités de modélisation, peuvent se présenter :

La première repose sur les modèles de connaissance qui nécessitent une analyse suivant les lois de la physique. Pour cela, il est nécessaire d'établir les équations mathématiques régissant le fonctionnement du processus **[10]**.

La seconde approche permet de modéliser le système en imposant une structure du modèle selon un critère donné (fonction de transfert, équation d'état, équation récurrente,...).

La forme du modèle qui représente correctement le fonctionnement du système réel obtenu qu'il soit de connaissance ou de représentation, nécessite une identification des paramètres de façon optimale pour permettre de représenter fidèlement le système sur le domaine de fonctionnement choisi.

L'identification paramétrique ne peut être effectuée qu'à l'aide des seules données disponibles: des signaux d'entrée judicieusement choisis et la réponse du système à ces signaux d'entrée.

Les paramètres du modèle sont obtenus par la minimisation de l'erreur de prédiction entre le signal de sortie mesuré et le signal estimé suivant un critère d'optimalité.

Ce chapitre est consacré à la modélisation du processus d'épuration de l'eau, avec la disposition des mesures d'entrée et de sortie du système réel afin d'identifier les paramètres de notre modèle.

2.2 Modèle ARMA

Le processus d'épuration est caractérisé par plusieurs grandeurs de sortie, mesurables, dont l'une constitue le résultat du processus (turbidité) et aussi par plusieurs grandeurs d'entrées qui sont de deux types :

- Une entrée sur laquelle il est possible d'agir (entrée de commande : le PAC).
- Des entrées sur lesquelles il n'est pas possible d'agir (perturbation), mais ces dernières sont mesurables.

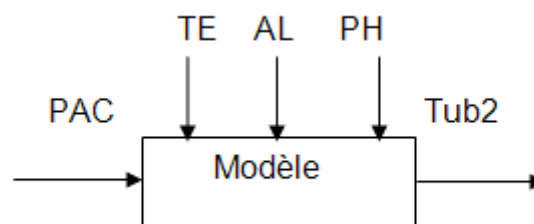


Figure 2.1 Modèle du processus d'épuration

Le modèle considéré est de type « boîte noire » qui est construit essentiellement sur la base de mesures effectuées sur les entrées et les sorties. Les paramètres des équations récurrentes sont à estimer à partir des mesures

disponibles, de manière à obtenir la meilleure précision possible (avec le nombre de paramètres le plus petit possible).

L'estimation de ces paramètres fait appel aux méthodes d'identification telles que : la méthode globale des moindres carrés ainsi que celle des moindres carrés récursifs qui permet de suivre l'évolution des paramètres par rapport au temps.

Le modèle mathématique utilisé pour ce processus, est de type ARMA (Auto Régressif à Moyenne Ajustée).

Ce modèle génère une fonction de transfert discrète $\frac{B(z)}{A(z)}$ qui relie chaque sortie à chaque entrée. Ce qui permet d'avoir des équations récurrentes à temps discret.

Le but de cette modélisation est de pouvoir agir sur la commande pour l'obtention d'une sortie désirée optimale.

2.3 Identification paramétriques du modèle ARMA

La modélisation auto-régressive à moyenne ajustée d'ordre (n, m) notée ARMA, peut être aussi définie par l'équation aux différences donnée par l'équation (2.1) [11].

$$\sum_{i=0}^n a_i y(k-i) = \sum_{i=0}^m b_i u(k-i) + e(k) \quad (2.1)$$

Avec $a_0 = 1$

$u(k)$: Entrée du système

$y(k)$: Sortie du système

$e(k)$: Bruit blanc.

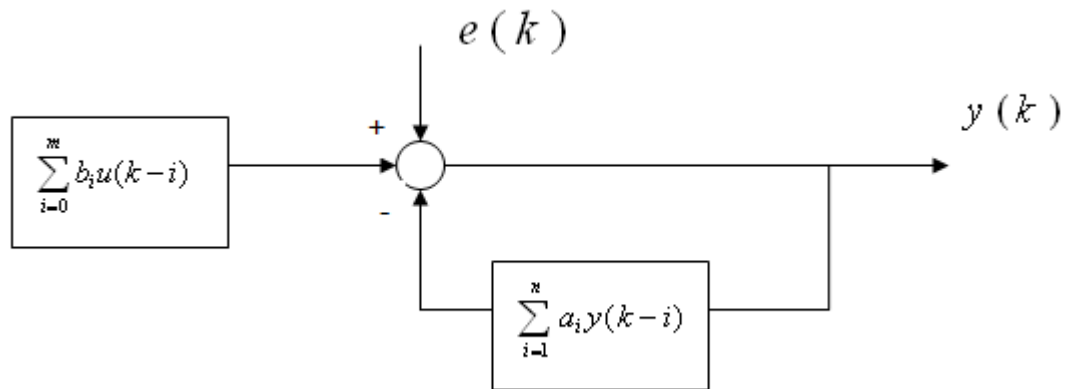


Figure 2.2 Schéma bloc de la représentation temporelle du modèle ARMA

En utilisant la transformée en Z, l'équation (2.1) peut s'écrire :

$$\sum_{i=0}^n a_i Y(Z) Z^{-i} = \sum_{i=0}^m b_i U(Z) Z^{-i} + E(Z) \quad (2.2)$$

Avec :

$$A(Z) = \sum_{i=0}^n a_i Z^{-i} \quad ; B(Z) = \sum_{i=0}^m b_i Z^{-i}$$

L'équation (2.2) s'écrira sous cette forme :

$$A(Z)Y(Z) = B(Z)U(Z) + E(Z)$$

En développant l'équation (2.1), la relation entrée sortie du système [10], s'écrira alors sous la forme:

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) - \dots - a_n y(k-n) + b_0 u(k) + \dots + b_m u(k-m) + e(k) \quad (2.3)$$

Dans cette relation $e(k)$ est une grandeur aléatoire représentant l'erreur de modélisation ou de l'effet des perturbations agissant sur le processus.

2.4 Estimation des paramètres du modèle ARMA

2.4.1 Méthode globale des moindres carrés

La méthode des moindres carrés a été introduite par Karl Gauss en 1809. Elle a été à la base de toutes les méthodes d'identification et d'estimation des paramètres, cette méthode est basée sur la minimisation d'une fonction quadratique **[11]**, définie comme:

$$J = \sum_{k=1}^N [\varepsilon(k)]^2$$

On définit l'erreur de prédiction comme étant la différence entre la sortie du système et la sortie du modèle:

$$\varepsilon(k) = y(k) - \hat{y}(k)$$

Où $\varepsilon(k)$ représente l'erreur de prédiction commise sur l'estimation.

La méthode des moindres carrés est basée sur la détermination des paramètres optimaux, c'est-à-dire ceux qui minimiseront un certain critère d'optimalité. Il représente la somme des carrés des erreurs de prédictions (Cf. Equation 2.5).

Le critère quadratique **[10]** à minimiser a pour expression:

$$J_N(\theta) = \frac{1}{N} \sum_{k=1}^N [e(k)]^2 \quad (2.4)$$

$$J_N(\theta) = \frac{1}{N} \sum_{k=1}^N [y(k) - \theta(k)^T \phi(k-1)]^2$$

Le minimum du critère J est obtenu par le jeu de paramètres qui annule sa dérivée dans l'espace paramétrique.

La minimisation du critère $J_N(\theta)$ consiste à trouver un optimum [12]:

$$\left[\frac{\delta J_N(\theta)}{\delta \theta} \right]_{\theta = \hat{\theta}(N)} = 0 \quad (2.5)$$

$$\frac{\delta J_N(\theta)}{\delta \theta} = \frac{2}{N} \left\{ \sum_{k=1}^N \phi(k) \left[y(k) - \theta^T \phi(k) \right] \right\} \Big|_{\theta = \hat{\theta}(N)} \quad (2.6)$$

De l'équation (2.5) et (2.6), on déduit la solution optimale au sens des moindres carrés de la forme de l'équation (2.7) :

$$\hat{\theta}(N) = \left[\sum_{k=1}^N \phi(k) \phi^T(k) \right]^{-1} \sum_{k=1}^N \phi(k) y(k) \quad (2.7)$$

Cette méthode est utilisée pour l'identification des paramètres du processus considéré dans l'application.

Soit l'exemple d'un processus à une entrée $u(k)$ et une sortie $y(k)$ [13], modélisé par la fonction de transfert:

$$\begin{aligned} H(z) &= \frac{B(z)}{A(z)} \\ &= \frac{z^{-1}(b_1 + b_2 z^{-1})}{1 - a_1 z^{-1}} \end{aligned}$$

a_1 : Pôle avec $|a_1| < 1$

b_2 : Coefficient du au retard pur.

b_1 : Coefficient qui assure la conservation du gain statique entre l'analogique et le discret.

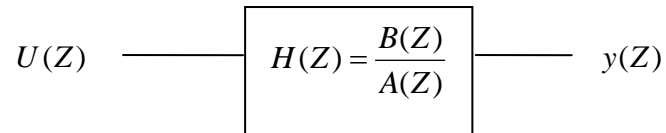


Figure 2.3 Représentation spectrale du modèle ARMA

L'équation de récurrence de ce système est :

$$y(k) = a_1 y(k-1) + b_1 u(k-1) + b_2 u(k-2) \quad (2.8)$$

$y(k)$ est une combinaison linéaire des mesures pondérées par les paramètres à identifier **[13]**.

L'équation (2.8) est mise sous la forme :

$$y(k) = \phi(k-1) * \theta \quad (2.9)$$

Soient :

$$\phi(k-1) = [y(k-1) \ u(k-1) \ u(k-2)]$$

$$\theta = \begin{bmatrix} a_1 \\ b_1 \\ b_2 \end{bmatrix}$$

ϕ, θ sont des vecteurs représentant respectivement les mesures et les paramètres à déterminer.

Avec

θ : Vecteur des paramètres à identifier

$\phi(k-1)$: Vecteur des données.

$y(k)$: Sortie du système.

Quelques opérations de calcul matriciel sont effectuées pour aboutir à l'algorithme.

Chaque terme de l'égalité de l'équation (2.9) est multiplié par $\phi^T(k-1)$

Ce qui permet d'obtenir l'équation (2.10)

$$\phi^T(k-1) * y(k) = \phi^T(k-1) * \phi(k-1) * \theta \quad (2.10)$$

l'expression de θ , est obtenue après une multiplication à gauche de chaque terme de l'équation (2.10) par l'inverse de $\phi^T(k-1) * \phi(k-1)$

$$\begin{aligned} & [\phi^T(k-1) * \phi(k-1)]^{-1} * \phi^T(k-1) * y(k) \\ &= [\phi^T(k-1) * \phi(k-1)]^{-1} * \phi^T(k-1) * \phi(k-1) * \theta \end{aligned}$$

Ce qui donne l'estimation du vecteur paramètres :

$$\hat{\theta} = [\phi^T(k-1) * \phi(k-1)]^{-1} * \phi^T(k-1) * y(k) \quad (2.11)$$

2.4.2 Méthode des moindres carrés récursifs (MCR)

La méthode des moindres carrés récursifs est utilisée pour rechercher, en temps réel, un modèle numérique d'un processus physique ou d'un signal.

Le principe consiste à minimiser un critère quadratique correspondant au carré de l'erreur, à l'instant courant, entre la sortie du modèle et la valeur de la sortie du processus ou celle du signal que l'on veut modéliser.

La mise en œuvre de l'algorithme récursif consiste aux étapes suivantes [11].

Soit :

$$R(t) = \sum_{k=1}^t \phi(k)\phi^T(k) = R(t-1) + \phi(k)\phi^T(k) \quad (2.12)$$

L'expression de l'équation (2.12) est introduite dans l'équation (2.11), ce qui donne:

$$\hat{\theta}(t) = R^{-1}(t) \sum_{k=1}^t \phi(k)y(k)$$

$$\hat{\theta}(t) = R^{-1}(t) \left[\sum_{k=1}^{t-1} \phi(k)y(k) + \phi(t)y(t) \right]$$

$$\hat{\theta}(t) = R^{-1}(t) [R(t-1)\hat{\theta}(t-1) + \phi(t)y(t)]$$

$$\begin{aligned}\widehat{\theta}(t) &= R^{-1}(t) \left[R(t) \widehat{\theta}(t-1) - \phi(t) \phi^T(t) \widehat{\theta}(t-1) + \phi(t) y(t) \right] \\ \widehat{\theta}(t) &= \widehat{\theta}(t-1) + R^{-1}(t) \phi(t) \left[y(t) - \widehat{\theta}^T(t-1) \phi(t) \right]\end{aligned}\quad (2.13)$$

D'après l'équation (2.13), la solution des moindres carrés récursive contient le terme $R^{-1}(t)$ qui nécessite une inversion matricielle.

Pour calculer cette expression, il faut utiliser le lemme d'inversion matricielle **[12]**, qui se présente sous la forme de l'équation (2.14):

$$A = B + C.D^T \quad (2.14)$$

La matrice A^{-1} inverse de A , quand elle existe est donnée par la relation (2.15):

$$A^{-1} = B^{-1} - B^{-1} C D^T B^{-1} \left[1 + D^T B^{-1} C \right]^{-1} \quad (2.15)$$

$$\text{Or } R(t)^{-1} = \left[R(t-1) - \phi(t) \phi^T(t) \right]^{-1} \quad (2.16)$$

$$\text{Si } B = R(t-1), C = \phi(t), D = \phi^T(t)$$

L'application de ce lemme à l'équation (2.16) donne alors:

$$R^{-1}(t) = R^{-1}(t-1) - \frac{R^{-1}(t-1) \phi(t) \phi^T(t) R^{-1}(t-1)}{1 + \phi^T(t) R^{-1}(t-1) \phi(t)} \quad (2.17)$$

L'introduction de la matrice du gain d'adaptation $P(t) = R^{-1}(t)$ (Cf. Equation 2.17), permet la mise en œuvre de l'algorithme des moindres carrés récursifs MCR et donc l'expression finale de remise à jour du vecteur paramètres :

$$\widehat{\theta}(t) = \widehat{\theta}(t-1) + p(t) \phi(t) \left[y(t) - \widehat{\theta}^T(t-1) \phi(t) \right] \quad (2.18)$$

$$\text{Avec : } p(t) = p(t-1) - \frac{p(t-1) \phi(t) \phi^T(t) p(t-1)}{1 + \phi^T(t) p(t-1) \phi(t)}$$

2.5 Application au processus d'épuration de l'eau :

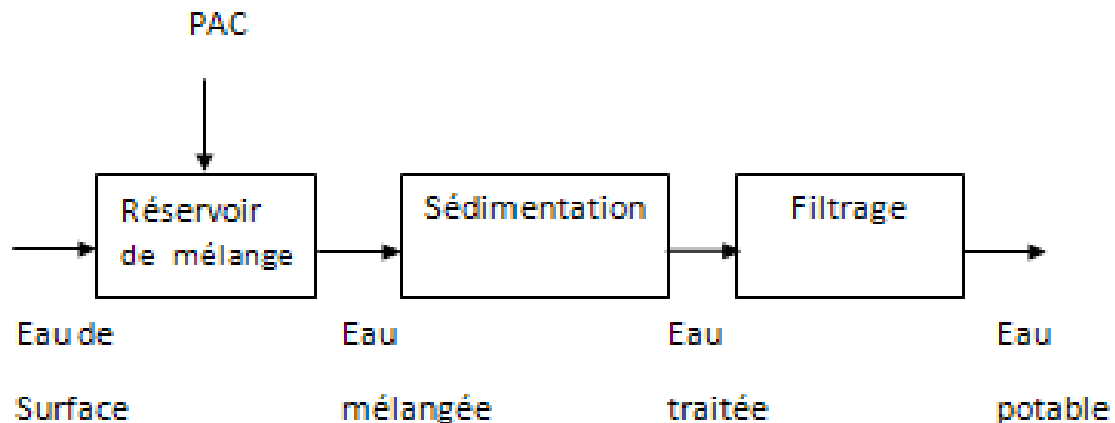


Figure 2.4 Processus d'épuration d'eau.

Dans ce processus, l'eau trouble se déverse dans un premier réservoir où une certaine quantité de coagulant (PAC) est ajoutée.

L'étape de sédimentation survient dans le second réservoir, où elle dure 3 à 5 heures selon la capacité de ce dernier; afin de réaliser la décantation des floccs. L'eau traitée sera filtrée puis stérilisée à l'aide de la chlorure (Figure 2.4).

Le modèle étudié auparavant est, adapté à cette application.

Les variables utilisées sont :

- TUB1 : Turbidité de l'eau originale (ppm)
- TUB2 : Turbidité de l'eau traitée (ppm)
- PAC : Quantité du PAC (ppm)
- TE : Température de l'eau ($^{\circ}\text{C}$)
- pH : Potentiel hydrogène
- AL : Alcalinité ou dureté (ppm) (basic)
- CL : Chlorure (ppm)

2.6. Modélisation du processus d'épuration de l'eau

Deux processus sont étudiés dans ce travail : celui des actions de contrôle de l'opérateur et celui de la station de l'épuration.

Le premier génère la commande introduite au processus d'épuration de l'eau dont la sortie, représente la turbidité.

Le processus d'épuration de l'eau est illustré par la figure. 2.5.

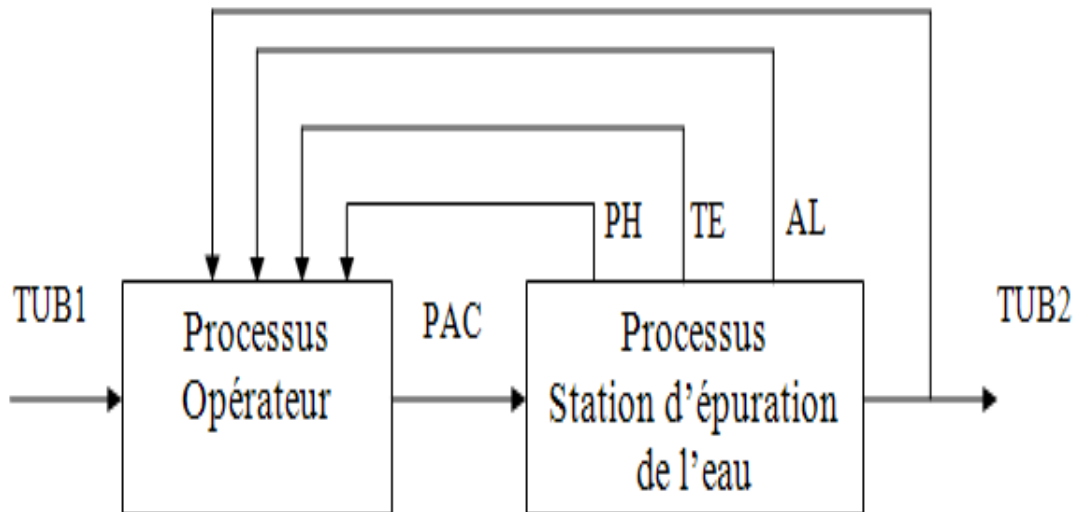


Figure 2.5 Diagramme de processus de régulation.

L'opérateur mesure la turbidité, en tenant compte des paramètres de l'environnement; ce qui lui permet de prendre une décision sur la quantité du coagulant à déverser. Une deuxième mesure de la turbidité est prise pour évaluer la qualité de l'eau.

Dans notre travail, nous avons utilisé les relevés expérimentaux d'une station d'épuration de l'eau locale sur une durée de 50 jours (Tableau 2.1).

PH	TE	AL	TUB1	PAC expert	TUB2
7,1	18,8	53	10	1300	1
7	18,6	50	17	1300	1
7,3	19,4	46	22	1400	2
7,1	19,5	40	50	1400	1
7,3	23,3	48	9	900	4
7,1	20,7	50	11	900	1
7,2	21,3	50	12	900	3
7,2	23,6	53	14	900	4
7,1	17,3	40	18	1100	1
7,2	18,8	55	12	900	3
7,2	18	50	8	1000	1,5
7,1	19,2	49	11	1000	2
7	18	37	50	1200	1,5
7	18	37	50	1200	1,5
7	17,3	41	30	1100	1,5
7,1	19,3	42	16	1100	3
7,1	18,8	50	10	1300	1
7	18,6	46	17	1300	1
7,1	23,3	50	22	1400	2
7,2	19,2	49	50	1400	1
7,25	23	41	9	900	4
7,39	19,5	42	11	900	1
7	18	48	12	900	3
7,29	17,7	42	14	900	4
7,1	17,3	42	18	1100	1
7,2	21,3	50	12	900	3
7,26	19	51	8	1000	1,5
7,1	18	51,5	11	1000	2
7,2	21	52,5	50	1200	1,5
7,39	22	53,5	35	1200	1
7,28	19	52,1	10	1300	1
7,29	19,1	52	17	1300	1
7,31	19,2	52,2	22	1400	2
7,32	19,3	52,25	50	1400	1
7,34	19,4	52,27	9	900	4
7,35	19,5	52,28	11	900	1
7,36	19,55	52,29	12	900	3
7,37	19,81	52,3	14	900	4
7,38	19,82	52,4	18	1100	1
7,39	19,88	52,5	12	900	3
7,2	17	50	18	1100	1

7,1	17,5	50,5	12	900	3
7,25	17,4	50,4	8	1000	1,5
7,15	19	51	11	1000	2
7,19	19,8	51,5	50	1200	1,5
7,26	19,8	52	35	1200	1,5
7	20,4	53,5	30	1100	1,5
7,45	21,5	52,5	16	1100	3
7,4	23	51,9	9	900	4

Tableau 2.1 : Relevés expérimentaux

TUB2 : turbidité de l'eau mesurée à l'instant courant k .

TUB1 : turbidité de l'eau mesurée à l'instant $(k-1)$.

PAC : commande décisionnelle du coagulant donnée par l'expert.

PH, TE, AL : paramètres d'environnement (perturbations)

2.7 Modèle linéaire

Dans notre cas, l'équation de récurrence correspondant au processus d'épuration de l'eau est donnée par l'équation (2.19):

$$y(k) = a_1 y(k-1) + b_1 u(k-1) + c TE(k-1) + d AL(k-1) + e PH(k-1) \quad (2.19)$$

Suivant le processus étudié, $y(k)$ représente la turbidité mesurée à l'instant k , $u(k-1)$ est la quantité de PAC ajoutée à l'instant $(k-1)$. Les instants $(k-1)$ et k correspondent respectivement à ceux d'avant et d'après le traitement.

Avec les notations du tableau 2.1, TUB2 correspond à la turbidité mesurée à l'instant k donc : $y(k) = TUB2$

TUB1 correspond à la turbidité mesurée à l'instant $(k-1)$ donc : $y(k-1) = TUB1$ et $u(k-1) = PAC$.

Les relevés expérimentaux génèrent le vecteur de paramètres de la relation (2.11), en utilisant la méthode globale des moindres carrés:

Suivant l'algorithme vu précédemment,

$$y(k) = \phi(k-1) * \theta$$

Avec :

$$\phi(k-1) = [y(k-1) \ u(k-1) \ TE(k-1) \ AL(k-1) \ PH(k-1)]$$

$$\theta^T = [a_1 \ b_1 \ c \ d \ e]$$

L'application permet d'estimer le vecteur de paramètres correspondant au modèle entrées – sorties, donné par l'équation (2.20).

$$\hat{\theta} = \begin{bmatrix} 0.0003 \\ -0.0036 \\ 0.2010 \\ -0.0143 \\ 0.3749 \end{bmatrix} \quad (2.20)$$

La méthode globale permet d'estimer les paramètres pour un faible nombre d'échantillons. Une estimation des paramètres est donc, donnée par la méthode des moindres carrés récursifs pour avoir une évolution des paramètres en temps réel jusqu'à stabilité (figure 2.6).

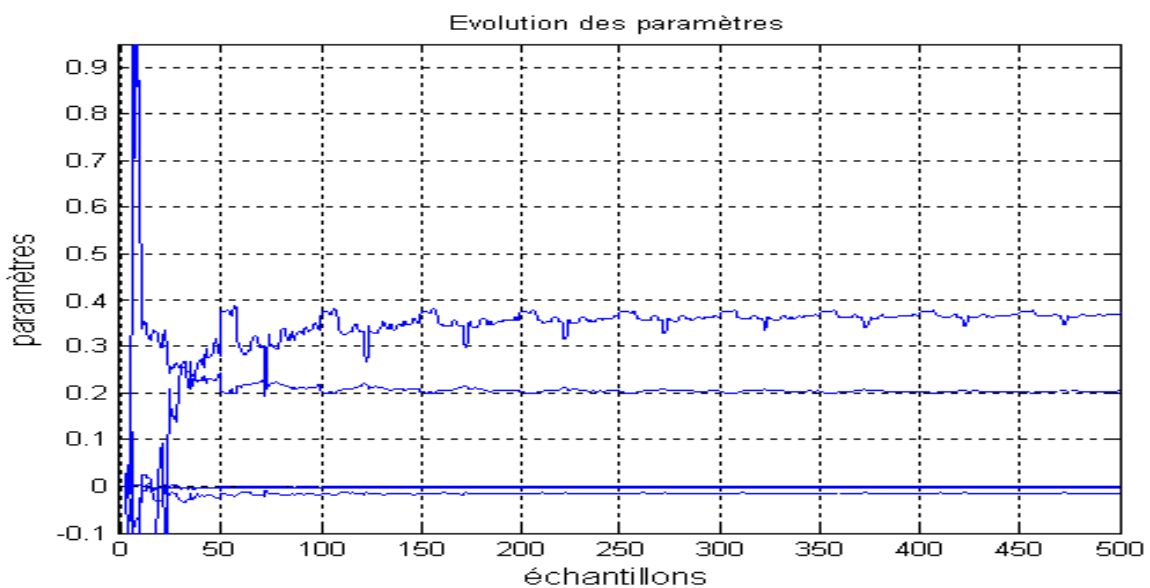


Figure 2.6 Evolution des paramètres du modèle linéaire

Nous avons obtenu le vecteur de paramètres suivant : $\hat{\theta} = \begin{bmatrix} 0.0003 \\ -0.0036 \\ 0.2010 \\ -0.0143 \\ 0.3748 \end{bmatrix}$ quasiment

égal à celui donné par la méthode globale.

Maintenant que le vecteur de paramètres est identifié, la sortie (turbidité) peut être calculée, et comparée à celle donnée par l'expert.

$$\phi(k-1) = [y(k-1) \ u(k-1) \ TE(k-1) \ AL(k-1) \ PH(k-1)]$$

$$y(k) = \phi(k-1) * \hat{\theta}$$

$$y(k) = \phi(k-1) * \begin{bmatrix} 0.0003 \\ -0.0036 \\ 0.2010 \\ -0.0143 \\ 0.3748 \end{bmatrix}$$

D'après l'équation (2.19), nous obtenons l'équation (2.21) fréquentielle en Z.

$$y(Z) = a_1 Z^{-1} y(Z) + b_1 Z^{-1} u(Z) + c Z^{-1} TE(Z) + d Z^{-1} AL(Z) + e Z^{-1} PH(Z) \quad (2.21)$$

$$y(Z)(1 - a_1 Z^{-1}) = b_1 Z^{-1} u(Z) + c Z^{-1} TE(Z) + d Z^{-1} AL(Z) + e Z^{-1} PH(Z)$$

$$y(Z) = \frac{b_1 Z^{-1}}{(1 - a_1 Z^{-1})} u(Z) + \frac{c Z^{-1}}{(1 - a_1 Z^{-1})} TE(Z) + \frac{d Z^{-1}}{(1 - a_1 Z^{-1})} AL(Z) + \frac{e Z^{-1}}{(1 - a_1 Z^{-1})} PH(Z) \quad (2.22)$$

La figure (2.7) représente l'évolution de la turbidité estimée par le modèle ARMA et celle donnée par l'expert.

Un indice de performance (Ip) qui représente la somme des valeurs absolues des erreurs entre la sortie du modèle et la sortie désirée, est calculé pour chaque modèle, jusqu'à définir le modèle optimal.

La variance obtenue correspondante à ce modèle, indique une valeur de 0.55 avec un indice de performance $I_p = 0.54$.

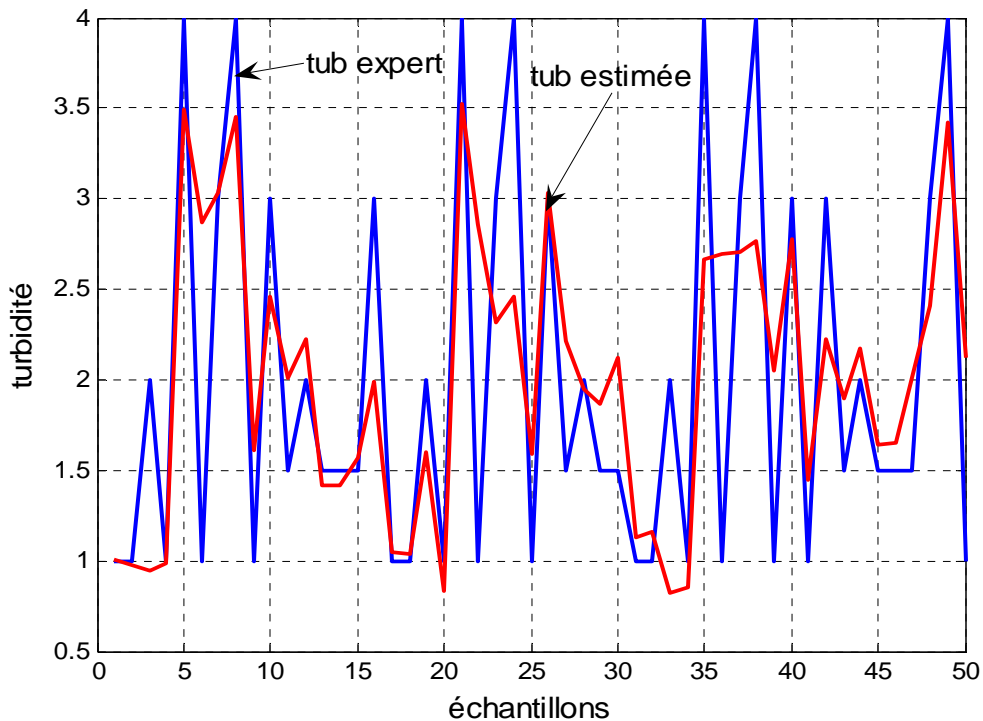


Figure 2.7 Turbidité estimée par le modèle et celle donnée par l'expert.

L'erreur en moyenne entre les deux turbidités est de 0.00047227.

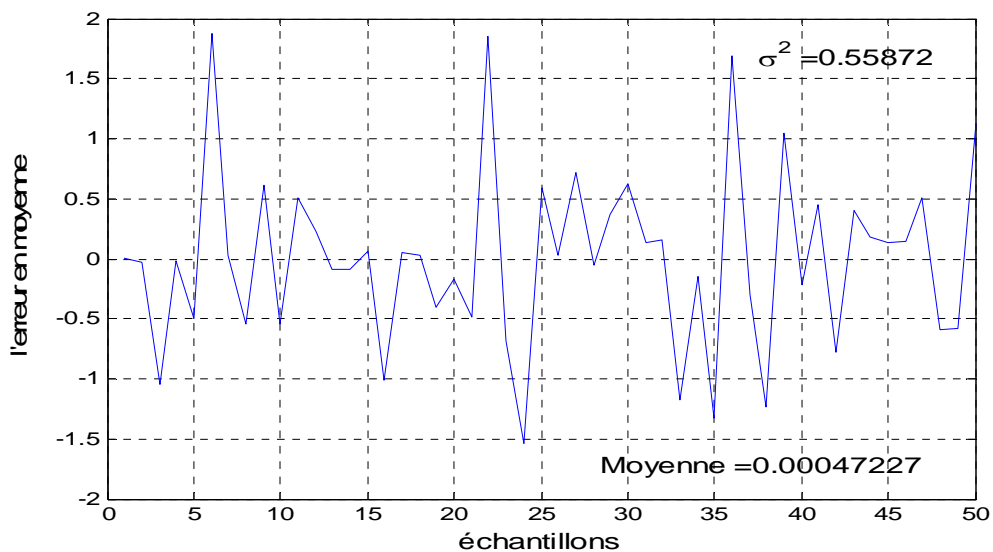


Figure 2.8 Erreur en moyenne entre la turbidité de l'expert et celle du modèle linéaire

Grâce au modèle ARMA du processus, nous pouvons estimer la quantité de PAC qu'il faut pour une turbidité donnée en sortie.

Le modèle ARMA permet d'avoir le modèle prédicteur suivant :

$$\hat{y}(k+1) = a_1 y(k) + b_1 PAC(k) + c TE(k) + d AL(k) + e PH(k) \quad (2.23)$$

Suivant les différentes mesures effectuées sur $y(k)$ qui est TUB1, $TE(k)$, $AL(k)$, $PH(k)$, le modèle ARMA indique le $PAC(k)$ (Cf. Equation 2.24) à injecter pour avoir une turbidité TUB2 à l'instant $(k+1)$.

$$PAC(k) = \frac{\hat{y}(k+1) - a_1 y(k) - c TE(k) - d AL(k) - e PH(k)}{b_1} \quad (2.24)$$

En appliquant cette commande au processus d'épuration de l'eau suivant une tolérance ne dépassant pas la valeur 3 (à partir des valeurs correspondant aux lignes du tableau 2.1); nous obtenons alors le PAC régulateur correspondant à la figure (2.9).

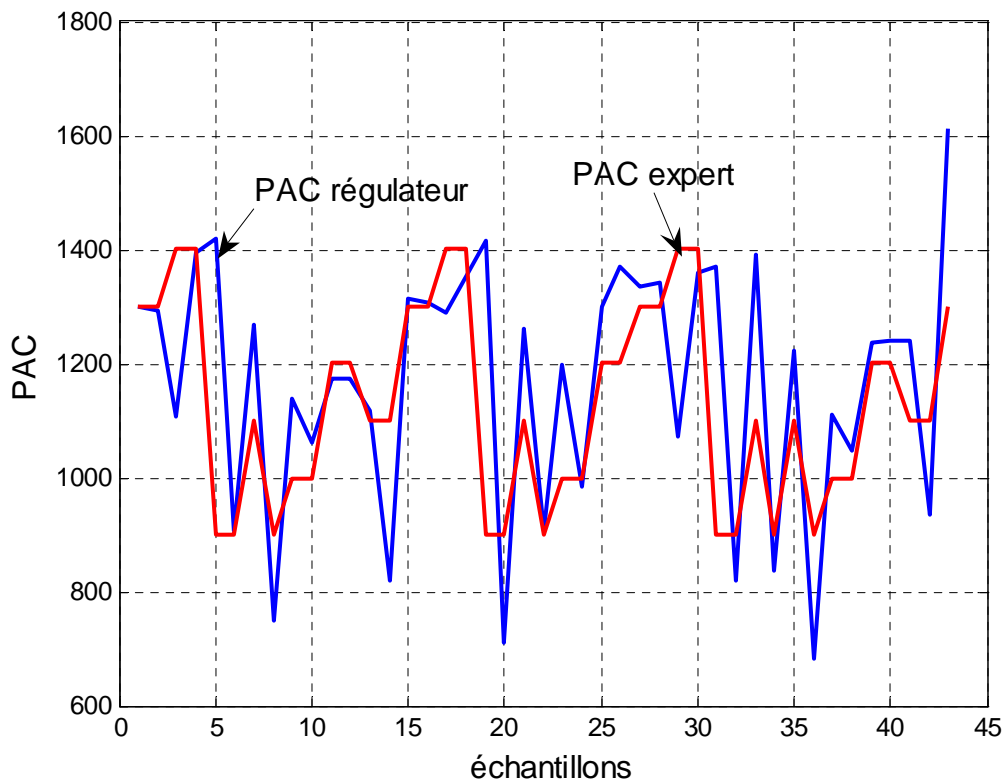


Figure 2.9 PAC régulateur et PAC expert pour un seuil de tolérance égal à la valeur 3

L'indice de performance correspondant au PAC ($Ip=115$) donné par la modélisation ARMA, permet d'améliorer les résultats obtenus par la formule empirique donnant un indice de performance de 201.11, utilisée jusqu'à présent dans les stations d'épuration d'eau (Cf. Equation 2.25).

$$PAC_{statique} = 9.11\sqrt{TB1} - 79.8PH + 12.7CL + 1255.6 \quad (2.25)$$

Le tableau 2.2 montre la valeur du PAC donnée par le régulateur pour une turbidité désirée.

Les valeurs données par le régulateur, approximativement celles données par l'expert.

PH	TE	AL	TUB1	TUB2 consigne	PAC expert	PAC régulateur
7,1	18,8	53	10	1	1300	1301,4
7	18,6	50	17	1	1300	1292,3
7,3	19,4	46	22	2	1400	1106,7
7,1	19,5	40	50	1	1400	1395,4
7,1	20,7	50	11	1	900	1419,5
7,2	21,3	50	12	3	900	907,9
7,1	17,3	40	18	1	1100	1269,9
7,2	18,8	55	12	3	900	748,5
7,2	18	50	8	1,5	1000	1140
7,1	19,2	49	11	2	1000	1061,9
7	18	37	50	1,5	1200	1174,3
7	18	37	50	1,5	1200	1174,3
7	17,3	41	30	1,5	1100	1117,7
7,1	19,3	42	16	3	1100	817,9
7,1	18,8	50	10	1	1300	1313,3
7	18,6	46	17	1	1300	1308,2
7,1	23,3	50	22	2	1400	1287,8
7,2	19,2	49	50	1	1400	1353,4
7,39	19,5	42	11	1	900	1414,4
7	18	48	12	3	900	710,8
7,1	17,3	42	18	1	1100	1262
7,2	21,3	50	12	3	900	907,9
7,26	19	51	8	1,5	1000	1198,1
7,1	18	51,5	11	2	1000	985
7,2	21	52,5	50	1,5	1200	1301,1
7,39	22	53,5	35	1	1200	1371,5
7,28	19	52,1	10	1	1300	1334,9
7,29	19,1	52	17	1	1300	1342,5
7,31	19,2	52,2	22	2	1400	1072
7,32	19,3	52,25	50	1	1400	1358,5

7,35	19,5	52,28	11	1	900	1369,4
7,36	19,55	52,29	12	3	900	817,8
7,38	19,82	52,4	18	1	1100	1390,5
7,39	19,88	52,5	12	3	900	838,5
7,2	17	50	18	1	1100	1223,9
7,1	17,5	50,5	12	3	900	683,3
7,25	17,4	50,4	8	1,5	1000	1110,1
7,15	19	51	11	2	1000	1048
7,19	19,8	51,5	50	1,5	1200	1237
7,26	19,8	52	35	1,5	1200	1241
7	20,4	53,5	30	1,5	1100	1241,1
7,45	21,5	52,5	16	3	1100	935,5
7,29	23,5	46	10	1	1300	1611,4

Tableau 2.2 Valeur du PAC suivant la consigne

Les valeurs obtenues par le modèle ARMA sont relativement plus proches comparativement à celles données par le modèle statique.

2.8 Modélisation statique

Le cahier des charges de la régulation de la turbidité consiste, non pas à obtenir une certaine valeur de consigne constante pour celle-ci ; mais à ce qu'elle soit toujours inférieure à une certaine valeur ou degré de tolérance.

Pour contrôler cette limitation, nous utilisons le gain statique obtenu par le modèle ARMA précédent.

La turbidité à l'instant k , est liée à la turbidité à l'instant $k-1$ au PAC déversé et aux paramètres externes.

Dans le domaine statique, à partir de l'équation (2.19), le facteur k peut être approximé à $k-1$. L'équation statique devient alors:

$$Y = a_1 Y + b_1 U + c TE + d AL + e PH$$

$$Y = \frac{b_1}{1-a_1} U + \frac{c}{1-a_1} TE + \frac{d}{1-a_1} AL + \frac{e}{1-a_1} PH \quad (2.26)$$

La limitation de la turbidité $y(k)$ implique alors celle de la commande $u(k)$ à la valeur statique U_0 . Celle-ci correspond à la valeur statique Y_0 qui est le seuil de tolérance imposé.

De ce fait l'équation (2.26) devient :

$$Y_0 = \frac{b_1}{1-a_1} U_0 + \frac{c}{1-a_1} TE + \frac{d}{1-a_1} AL + \frac{e}{1-a_1} PH \quad (2.27)$$

$$U_0 = \frac{1-a_1}{b_1} y_0 - \frac{c}{b_1} TE - \frac{d}{b_1} AL - \frac{e}{b_1} PH \quad (2.28)$$

U_0 peut aussi être obtenue par l'équation (2.22) en mettant $Z=1$

$$U = \frac{1-a_1}{b_1} y - \frac{c}{b_1} TE - \frac{d}{b_1} AL - \frac{e}{b_1} PH$$

Cette limitation nous permet de donner la valeur maximale de la quantité du PAC à ne pas dépasser.

Le PAC déversé par l'expert est appliqué au cas où la turbidité maximale est imposée à une tolérance égale à 2 (Cf. Figure 2.10).

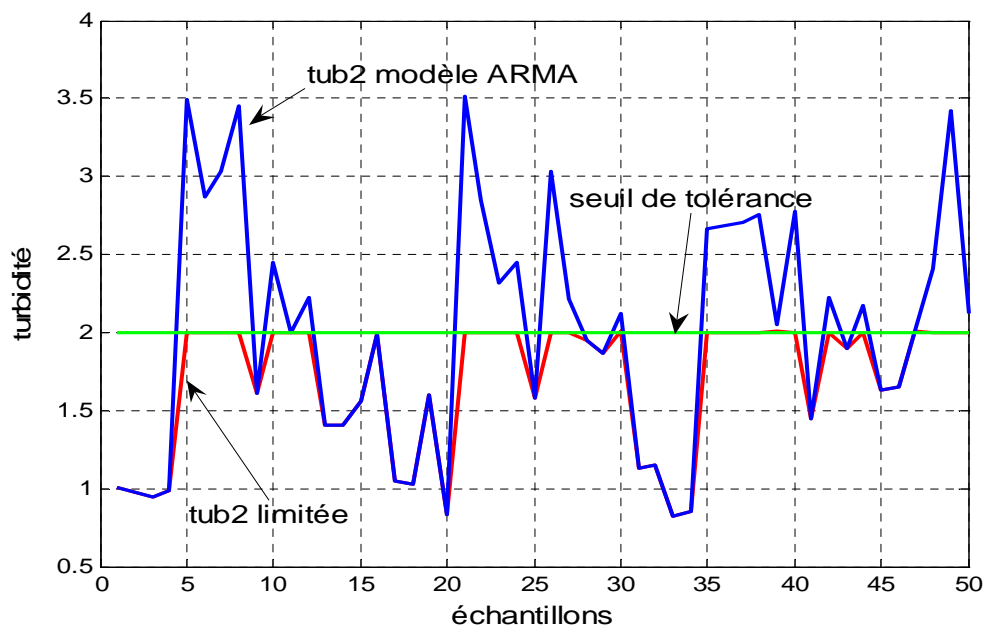


Figure 2.10 Turbidité due au PAC modèle et celle limitée par le seuil de tolérance

La figure (2.10) montre bien que la turbidité est bien limitée et ne dépasse pas la valeur maximale de 2 (seuil de tolérance).

La valeur minimale du PAC à introduire, avec limitation de la turbidité à 3.5 est de 1000 ppm, alors que celle de l'expert sans limitation est de 900 ppm. Pour avoir donc une turbidité plus faible il est nécessaire de mettre une quantité plus importante de PAC (Cf. Figure 2.11).

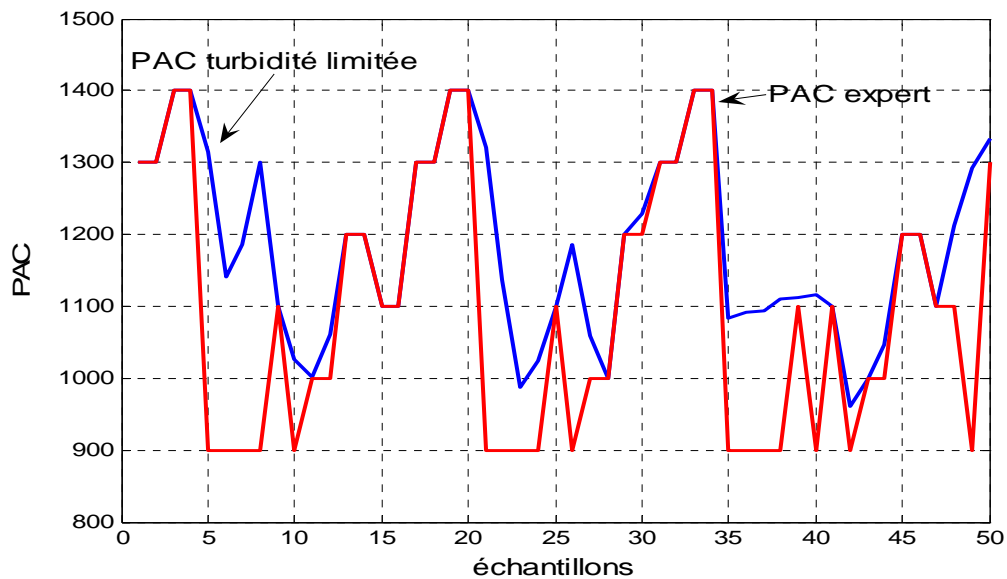


Figure 2.11 PAC limité suivant un seuil de tolérance imposé égale à 3.5

Cette courbe montre que le PAC est limité vers le haut par une valeur minimale de 970 ppm. L'indice de performance calculé pour cette limitation suivant la tolérance imposée, indique une valeur de $ip=88.49$ et une variance égale à 16630.

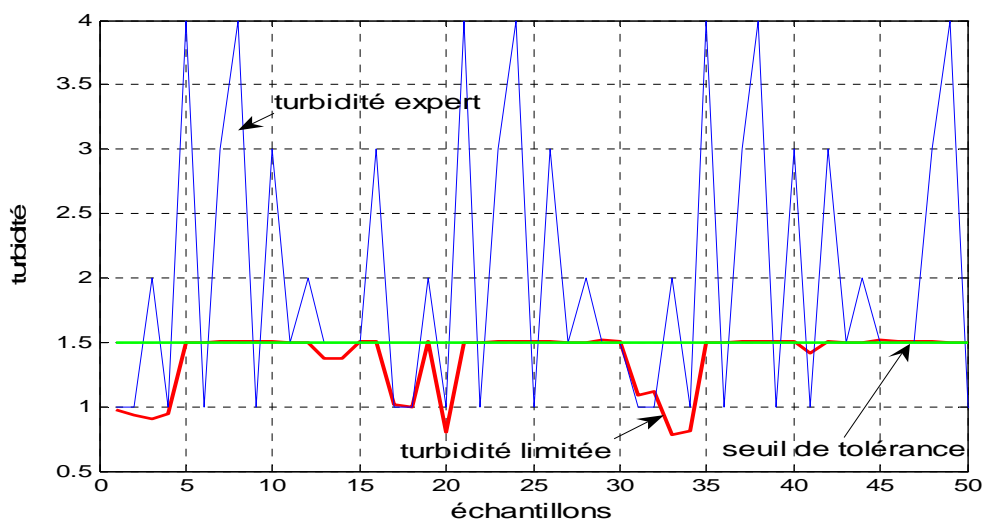


Figure 2.12 Limitation de la turbidité par la valeur 1.5

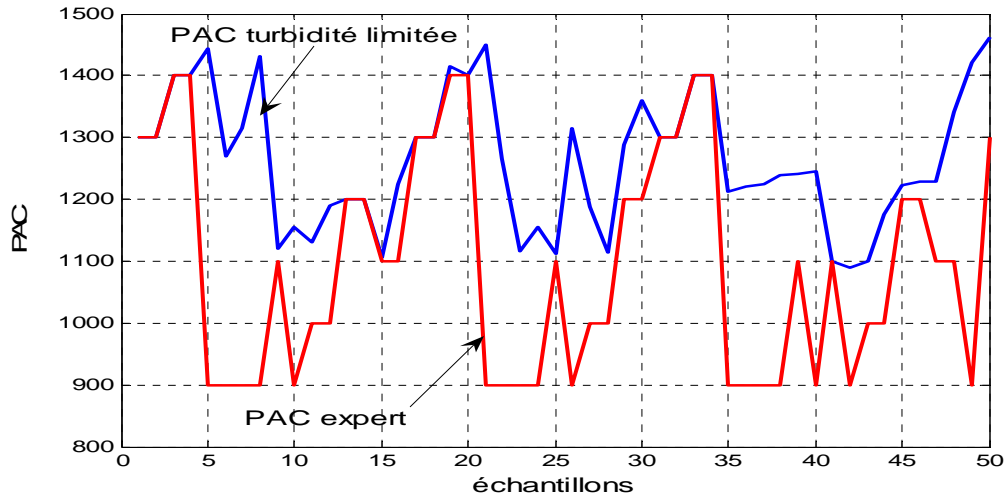


Figure 2.13 PAC limité suivant un seuil de tolérance imposé égale à 1.5

L'utilisation du gain statique obtenue à partir le modèle ARMA, limite bien la turbidité au seuil de tolérance imposé (Cf. Figure 2.12). De même le PAC est aussi limité vers le bas (970 ppm pour un seuil tolérance de 3.5, et 1100 ppm pour un seuil de tolérance de 1.5).

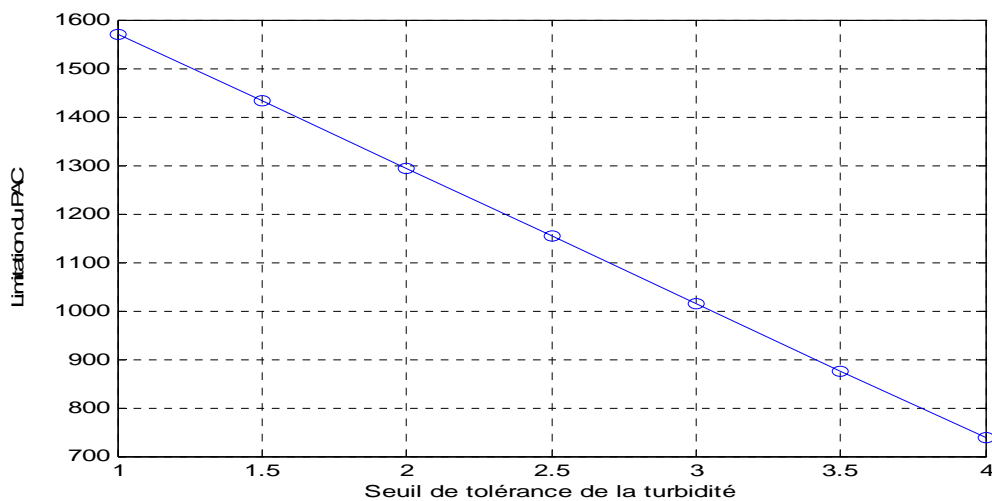


Figure 2.14 PAC limite en fonction de la turbidité limitée

Différents seuils de tolérance sont considérés, pour voir la variation du PAC. Ainsi, pour une turbidité ne dépassant pas 2.5 ppm, le PAC est aussi limité. Sa valeur doit être au delà de 1160 ppm (Cf. Figure 2.14). La figure 2.14 montre que plus le seuil de tolérance imposé est faible, plus la quantité du coagulant à mettre est grande. Ce qui indique une relation linéaire entre ces deux variables.

2.9 Conclusion

Le modèle ARMA utilisé pour le processus d'épuration de l'eau, avait pour but d'améliorer l'indice de performance par rapport à celui généré par le modèle statique, donné par la formule empirique, utilisé dans les stations d'épuration jusqu'à ce jour. Grâce au modèle prédictif, la valeur du coagulant peut être prédite pour une consigne désirée.

La limitation de la turbidité peut être imposée par le modèle statique pour l'obtention d'une eau purifiée.

Toutefois le modèle ARMA a permis une amélioration par rapport au modèle empirique, mais la détermination de l'ordre (p,q) du modèle pose une difficulté, et demande une large connaissance sur la dynamique du système.

Le chapitre suivant est consacré à la modélisation neuronale pour l'amélioration de l'indice de performance.

Chapitre 3

Modélisation neuronale

3.1 Introduction

La modélisation mathématique des systèmes peut parfois conduire à des équations mathématiques très complexes, conduisant à une implémentation trop coûteuse en temps de calcul, particulièrement dans le cas d'une application à temps réel.

Ainsi les réseaux de neurones constituent une alternative intéressante à l'élaboration des modèles pour les systèmes réels lorsqu'il n'est pas envisageable d'obtenir un modèle conventionnel.

Cette approche a été inspirée par le modèle humain qui est et restera toujours le meilleur modèle intelligent. Ce phénomène de raisonnement, de pouvoir parler, de calculer et surtout d'apprendre a toujours été le but des chercheurs qui relèvent le défi de le reprendre par une intelligence artificielle.

Deux types d'approches ont été essentiellement explorés **[15]**:

Procéder d'abord à l'analyse logique des tâches relevant de la cognition humaine et tenter de les reconstituer par programme. C'est cette approche qui a été privilégiée par l'Intelligence Artificielle et la psychologie cognitive classique. Cette démarche est étiquetée sous le nom de cognitivisme.

Puisque la pensée est produite par le cerveau qui en est une propriété, commencer par étudier comment celui-ci fonctionne et comment il fait des analyses pour traiter les informations a conduit à l'étude de réseaux de neurones formels.

La seconde approche a donc mené à la modélisation des neurones biologiques par des neurones formels qui sont des réseaux complexes d'unités de calcul élémentaire interconnectées.

3.2 Propriétés des neurones biologiques

Le neurone biologique possède plusieurs propriétés qui sont citées ci-dessous :

Les neurones sont les éléments de base des cellules nerveuses. Ce sont les éléments fonctionnels du système nerveux. Ils sont composés d'un corps cellulaire et de deux types de prolongements :

Les dendrites représentent les entrées du neurone, ils conduisent le message nerveux jusqu'au corps cellulaire, l'axone est sa sortie, il conduit le message nerveux en direction d'une cellule active jusqu'à une synapse ; zone de communication entre deux neurones.

En fait selon le type de la synapse, l'activité d'un neurone peut renforcer ou diminuer l'activité de ses voisins. On parle ainsi de synapse excitatrice ou inhibitrice.

La contribution d'un neurone au cerveau tout entier, effectue un travail très simple en réalisant des opérations basiques, et pourtant, l'interconnexion de 10 milliards de neurones ensemble, peut créer une entité pensante.

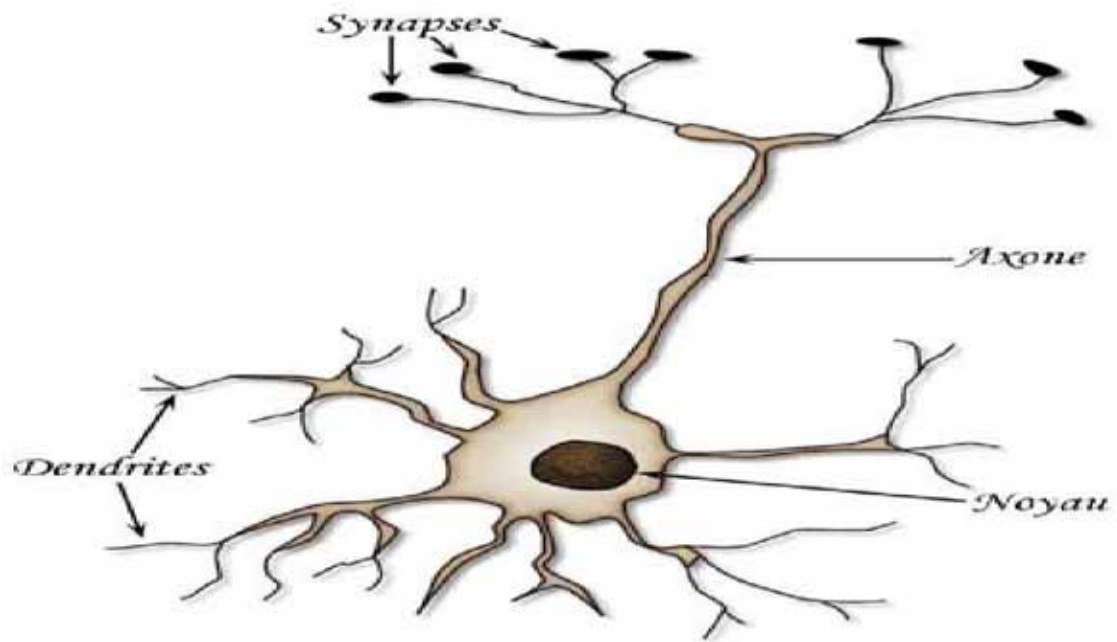


Figure 3.1 Schéma d'un neurone biologique

En effet, les performances du système nerveux sont dues à la connexion d'un grand nombre de neurones.

Cependant cela n'est pas suffisant ; un cerveau ne peut rien faire s'il n'a pas de quoi apprendre. Il a besoin d'informations venant de l'extérieur. C'est pour cela qu'il est relié aux différents organes du corps. Par exemple, il reçoit les images provenant des yeux, les sons, les douleurs. Grâce à ces informations il est capable de faire son apprentissage, reconnaître la forme d'un objet par exemple. Lors de l'apprentissage, le réseau a du modifier ses poids et biais jusqu'à la reconnaissance parfaite de cet objet.

3.3 Les neurones formels

Le premier modèle de base de réseaux de neurones est proposé en 1949 par Mac Culloch et Pitts.

L'objectif des réseaux de neurones artificiels est de modéliser le fonctionnement des neurones réels, mais aussi de permettre un apprentissage.

Un neurone est un opérateur mathématique, caractérisé par une fonction non linéaire. C'est une fonction algébrique, dont la valeur dépend des variables d'entrées, à chacune des entrées est associé un paramètre appelé poids synaptique.

La valeur de la fonction est appelée sortie du neurone qui est finalement la somme des valeurs des entrées multipliées par leurs poids correspondants. Cette somme devient l'argument de la fonction d'activation, qui est le plus souvent une des formes présentées (Cf. Figure 3.2).

Pour résumer, un neurone réalise alors trois opérations sur ses entrées :

- Pondération : multiplication de chaque entrée par un paramètre appelé poids synaptique,
- Sommation : une sommation des entrées pondérées est effectuée,
- Activation : passage de cette somme dans une fonction, appelée fonction d'activation après avoir ajouté un seuil nommé biais à la somme.

La valeur calculée est la sortie du neurone qui est transmise aux neurones suivants.

La somme $\sum_{i=1}^n w_i x_i + b$ forme le potentiel d'activation.

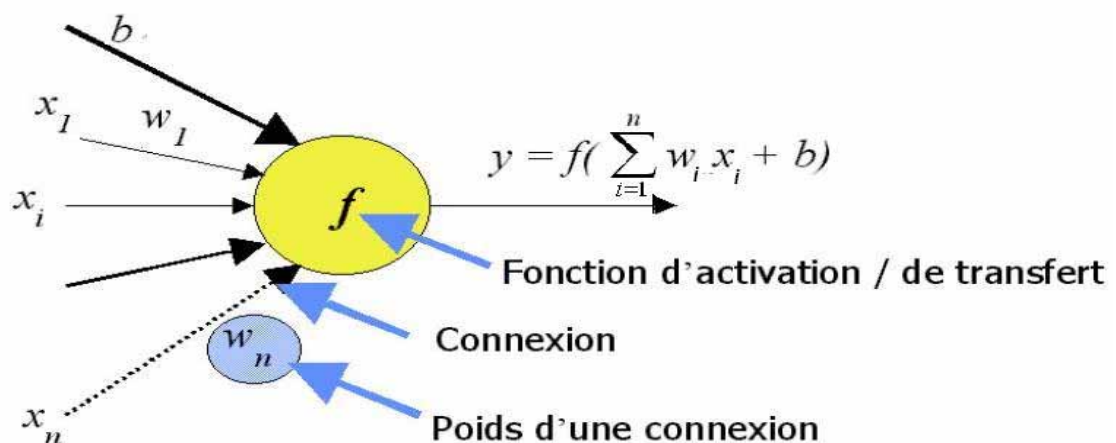
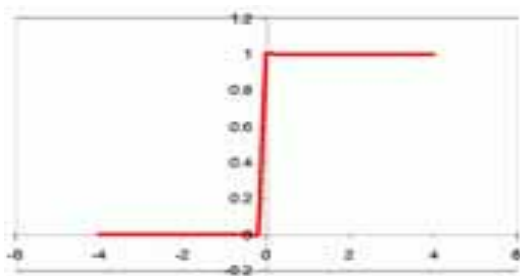


Figure 3.2 Schéma d'un neurone formel

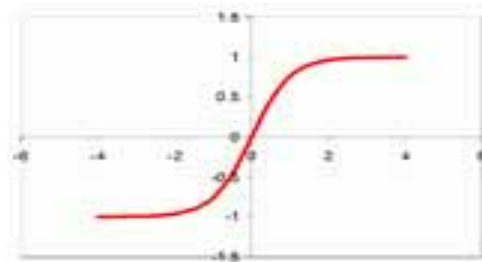
3.4 Fonction d'activation

Après une sommation des entrées pondérées par les poids synaptiques, le potentiel formé est passé à travers une fonction d'activation, afin d'obtenir la réponse du neurone. Le résultat obtenu alors est la sortie du neurone.

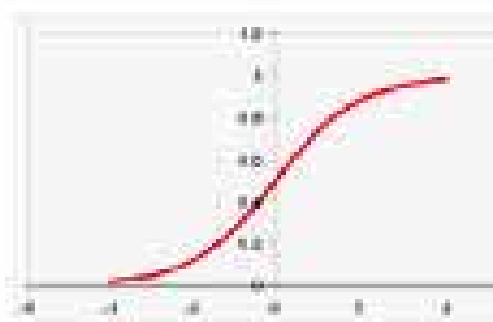
La fonction d'activation est choisie à partir du domaine d'application, voici les fonctions d'activation les plus utilisées [15]:



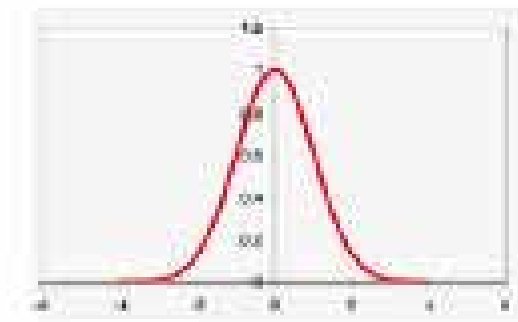
a) Fonction à seuil



b) Fonction hyperbolique



c) Fonction sigmoïde unipolaire



d) Fonction gaussienne

Figure 3.3 Différents types de fonction d'activation

3.5 Les réseaux de neurones artificiels

Un réseau de neurones artificiels est un ensemble de neurones interconnectés entre eux. La sortie de chaque neurone peut être reliée en entrée à plusieurs autres neurones. La complexité du réseau n'est définie que par le nombre

de neurones et le nombre de connexions. Le réseau peut être divisé en plusieurs couches.

Les neurones d'entrée du réseau sont reliés au monde extérieur ; les neurones de sortie donnent le résultat et entre les deux une ou plusieurs couches comportant d'autres neurones, dites couches cachées.

L'échange de l'information se propage des neurones d'entrées vers la sortie, mais cela peut être fait selon deux structures, celle des réseaux non bouclés (statiques) ou alors les réseaux bouclés (dynamiques).

3.5.1 Réseaux de neurones bouclés

Appelé aussi récurrent, contrairement à l'autre structure, ce réseau est défini par des connexions récurrentes [16], qui ramènent les informations en arrière par rapport au sens de la propagation de l'information. Dans ce cas, les sorties futures de quelques ou tous les neurones seront influencées par les sorties de quelques ou tous les neurones du réseau.

3.5.2 Réseaux de neurones non bouclés

Ce type de réseau ne contient pas de boucle synaptique fermée. Il réalise donc, de manière générale, une relation algébrique non linéaire entre ses entrées et ses sorties de façon que la sortie actuelle d'un neurone n'ait aucune influence sur les sorties futures du réseau.

Après avoir choisi le type de réseau de neurone qui convient le mieux à l'application, en se basant sur les variables d'entrées/ sorties du modèle réel, le développement du réseau fait appel à des exemples de comportement, cette phase est appelée apprentissage du réseau de neurones.

3.6 Apprentissage

L'apprentissage [16] [17] est vraisemblablement la caractéristique la plus intéressante des réseaux de neurones.

Selon la description du modèle, l'algorithme d'apprentissage doit être défini pour permettre l'ajustement de ses paramètres (poids synaptiques et biais) jusqu'à l'adaptation de manière à obtenir un réseau représentatif du système à modéliser.

Il existe deux grandes classes des algorithmes d'apprentissage; selon que l'apprentissage est dit non supervisé ou supervisé. Cette distinction repose sur la forme des exemples d'apprentissage.

3.6.1 Apprentissage non supervisé

Ce mode d'apprentissage est appelé aussi « apprentissage par compétition ». Dans ce cas, des exemples ou prototypes sont présentés au réseau puis le réseau est laissé s'auto-organiser au moyen des lois locales qui régissent l'évolution des poids synaptiques et sans la présence d'un superviseur.

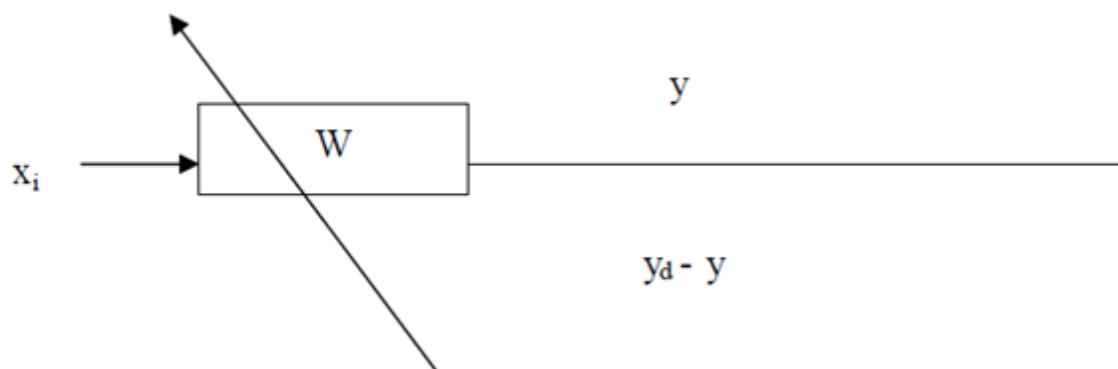


Figure 3.4 Apprentissage non supervisé

3.6.2 Apprentissage supervisé

Un apprentissage est appelé supervisé, lorsque la sortie désirée est considérée; c'est-à-dire que le réseau est forcé à converger vers un résultat précis.

Ce type d'apprentissage est réalisé si un certain nombre d'exemples de comportement d'entrées/sorties sont défini, pour chaque signal d'entrée est associée une réponse spécifique.

Après une initialisation des poids et des biais du réseau, en général à des valeurs aléatoires, un calcul de l'erreur entre la sortie désirée et la sortie du réseau est fait, ensuite une correction des poids et des biais est appliquée.

C'est une méthode d'optimisation qui modifie les poids au fur et à mesure des itérations pendant lesquelles la totalité des exemples sont présentés, afin de minimiser l'écart entre les sorties calculées et les valeurs expérimentales correspondantes.

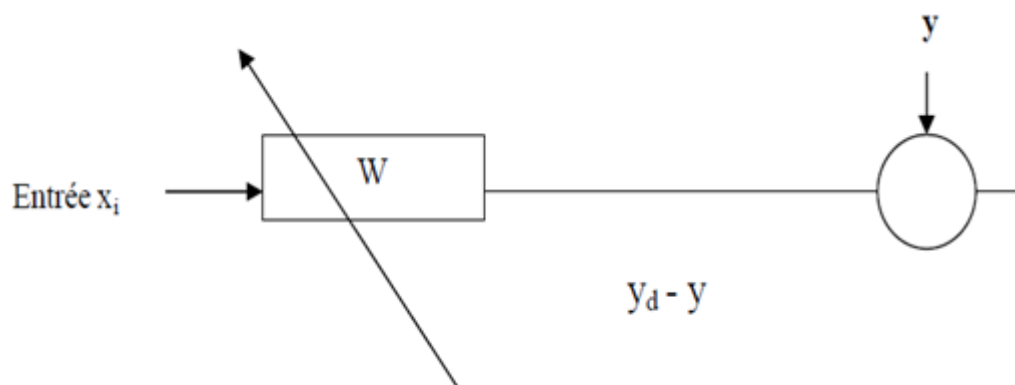


Figure 3.5 Apprentissage supervisé

3.7 L'algorithme de rétro-propagation :

La technique de rétro-propagation [18] est basée sur la méthode du gradient (Backpropagation en anglais), c'est une méthode qui permet de calculer le gradient de l'erreur en sortie du réseau, cette erreur sera transmise en sens inverse vers l'entrée.

Le but est de minimiser, à chaque pas d'apprentissage i la variance de l'erreur en sortie du réseau.

L'erreur commise en sortie du réseau à l'instant i est : $e_i = o_i - s_i$ avec

o_i : Sortie du réseau

s_i : Sortie expérimentale.

Par conséquent l'erreur totale est :

$$E_i = \frac{1}{2} e_i^2 = \sum_{i=1}^n (o_i - s_i)^2 \quad (3.1)$$

n étant le nombre d'échantillons présentés au réseau.

Pour minimiser E_i (Cf. Equation 3.1), le gradient de l'erreur par rapport à chaque poids est calculé (Cf. Equation 3.2), puis une modification des poids dans le sens inverse du gradient sera faite.

$$\nabla E_{i/z} = \frac{1}{2} \nabla [o_i^T o_i - 2 o_i^T s_i]_{/z} \quad (3.2)$$

Les fonctions d'activation g et f (Cf. Equation 3.3 et 3.4) sont considérées comme étant des sigmoïdes unipolaires respectivement pour la couche cachée et pour celle de sortie.

La sortie de la couche cachée sera donnée par :

$$h_i = g(W * x_i + w_0) \quad (3.3)$$

La sortie du réseau est :

$$o_i = f(Z * h_i + z_0) \quad (3.4)$$

Le gradient de E_i par rapport aux poids synaptiques est calculé à partir de l'équation (3.1).

- La mise à jour des poids synaptiques Z peut se calculer comme suit :

$$\nabla E_{i/z} = \frac{\partial E_i}{\partial Z} = \frac{\partial E_i}{\partial o_i} \frac{\partial o_i}{\partial (Z * h_i + z_0)} \frac{\partial (Z * h_i + z_0)}{\partial Z} \quad (3.5)$$

Le calcul des dérivées partielles donne :

$$\frac{\partial o_i}{\partial (Z * h_i + z_0)} = f(Z h_i + z_0) * [1 - f(Z * h_i + z_0)] = o_i * (1 - o_i) \quad (3.6)$$

$$\frac{\partial E_i}{\partial o_i} = o_i - s_i \quad (3.7)$$

$$\frac{\partial (Z * h_i + z_0)}{\partial Z} = h_i^T \quad (3.8)$$

Les dérivées trouvées par les équations (3.6), (3.7), (3.8) sont remplacées dans l'équation (3.5), de ce fait, la prochaine valeur ajustée des poids synaptiques sera :

$$Z(i+1) = Z(i) - \eta \nabla E_{i/z} = Z(i) + \eta (s_i - o_i) * o_i * (1 - o_i) h_k^T \quad (3.9)$$

avec η le gain d'apprentissage.

- La variation des biais synaptiques z_0 peut se calculer comme suit :

$$\nabla E_{i/z_0} = \frac{\partial E_i}{\partial z_0} = \frac{\partial E_i}{\partial o_i} \frac{\partial o_i}{\partial (Z * h_i + z_0)} \frac{\partial (Z * h_i + z_0)}{\partial z_0} \quad (3.10)$$

Le calcul des dérivées partielles donne :

$$\frac{\partial E_i}{\partial o_i} = o_i - s_i \quad (3.11)$$

$$\frac{\partial o_i}{\partial (Z * h_i + z_0)} = f(Z h_i + z_0) * [1 - f(Z * h_i + z_0)] = o_i * (1 - o_i) \quad (3.12)$$

$$\frac{\partial (Z * h_i + z_0)}{\partial z_0} = 1 \quad (3.13)$$

De ce fait, la prochaine valeur ajustée des biais synaptiques sera :

$$z_0(i+1) = z_0(i) - \eta \nabla E_{i/z} \quad (3.14)$$

Les dérivées (3.11), (3.12) et (3.13) sont remplacées dans (3.10), ce qui donne :

$$z_0(i+1) = z_0(i) + \eta (s_i - o_i) \cdot o_i \cdot (1 - o_i) \quad (3.15)$$

- La variation des poids synaptique W peut se calculer comme suit :

$$\nabla E_{i/w} = \frac{\partial E_i}{\partial w} = \frac{\partial E_i}{\partial o_i} \frac{\partial o_i}{\partial h_i} \frac{\partial h_i}{\partial (w^* x_i + w_0)} \frac{\partial (w^* x_i + w_0)}{\partial w}$$

$$\frac{\partial E_i}{\partial o_i} = o_i - s_i$$

$$\frac{\partial o_i}{\partial h_i} = \frac{\partial f(z h_i + z_0)}{\partial h_i} = z^* o_i (1 - o_i)$$

$$\frac{\partial h_i}{\partial (w^* x_i + w_0)} = h_i \cdot (1 - h_i)$$

$$\frac{\partial (w^* x_i + w_0)}{\partial w} = x_i$$

Pour les couches intermédiaires, les poids sont adaptés suivant la loi :

$$W(i+1) = W(i) - \eta \nabla E_{i/w} = W(i) + \eta Z(k) o_i (1 - o_i) \cdot h_i \cdot (1 - h_i) x_i$$

De même les biais de la couche cachée sont adaptés ainsi :

$$w_0(i+1) = w_0(i) - \eta \nabla E_{i/w_0} = w_0(i) + \eta Z(i) (s_i - o_i) \cdot o_i \cdot h_i \cdot (1 - h_i)$$

3.8 Application des réseaux de neurones à la modélisation des actions de contrôle de l'opérateur dans la station d'épuration de l'eau

3.8.1 Introduction

Les réseaux de neurones sont bien adaptés au traitement de processus fortement non linéaires en utilisant les réseaux multicouches avec rétro-propagation du gradient.

Dans cette étude une approche efficace est présentée pour la configuration optimale d'un réseau multicouche avec rétro-propagation du gradient, spécifiquement adapté à la modélisation des actions de contrôle d'un expert.

Pour cela, un premier réseau a été établi, en choisissant comme vecteur d'entrée à deux composantes, $x = [\text{tub1}, \text{tub2}]$.

Une deuxième architecture a été proposée pour ce réseau, [19] en considérant une troisième entrée qui est, l'erreur ($\text{tub2}-\text{tub1}$), le vecteur d'entrée devient alors: $x = [\text{tub1}, \text{tub2}, \text{tub2}-\text{tub1}]$.

Pour la conception et la mise en œuvre du premier réseau de neurones, voici les différentes phases établies :

Les grandeurs qui influencent significativement le phénomène que l'on cherche à modéliser ont été déterminées et représentées comme étant les entrées du réseau.

- Une collecte de mesures élaborées par un opérateur expérimenté, a été implémentée pour l'apprentissage et l'évaluation des performances du réseau de neurones.
- détermination d'un nombre approprié de neurones dans la couche cachée pour assurer une fiabilité maximale du système [20], en exploitant des signaux issus de mesure.

3.8.2 Acquisition de données d'entrée/sortie du réseau

Cette étude est basée sur l'exploitation des données expérimentales relatives à la turbidité avant et après le traitement, qui sont des données d'entrée pour l'apprentissage et pour le test du réseau de neurones. La sortie désirée est la valeur du coagulant qu'il faut rajouter pour contrôler et obtenir la valeur de turbidité désirée.

3.9 Mise en œuvre du réseau de neurones perceptron multicouche :

Pour la conception du modèle, le vecteur d'entrée a été choisi le plus compact que possible, en choisissant les informations pertinentes pour la grandeur à modéliser.

Mis à part les neurones d'entrée et de sortie, il faut décider du nombre de couches intermédiaires ou cachées. Sans couche cachée, le réseau n'offre que de faibles possibilités d'adaptation. Néanmoins, il a été démontré qu'un Perceptron multicouches avec une seule couche cachée pourvue d'un nombre suffisant de neurones, peut approximer n'importe quelle fonction avec la précision souhaitée [21].

Donc le modèle initial implémenté est de type perceptron multicouches à rétro-propagation du gradient « 2-i-1 », comportant des entrées, une couche cachée, dont la détermination du nombre approprié de neurones fera l'objet d'une optimisation, et une couche de sortie.

Il n'existe pas une théorie, qui permet de définir ou de prévoir le nombre de neurones cachés nécessaires, pour obtenir une performance spécifique du modèle.

Le type d'apprentissage adopté à ce réseau I est l'apprentissage supervisé, puisque les valeurs de la sortie désirée sont connues, que le réseau de neurones devra atteindre au cours de l'apprentissage.

3.10 Paramétrage et test du réseau I

L'objectif est de modéliser les actions de l'expert, ceci veut dire reprendre les mêmes décisions que celle prise par l'expert.

La turbidité avant et après le traitement représentent les paramètres d'entrées du réseau. Elles sont représentées par $tub1$ et $tub2$.

Le vecteur de sortie étant la quantité de coagulant à ajouter, il est représenté par la variable PAC.

Le nombre de neurones dans la couche cachée a été déterminé pendant la phase d'apprentissage. En pratique, ce nombre a été déterminé de façon empirique, par essais successifs jusqu'à l'obtention de performances satisfaisantes.

Une base de données a été utilisée pour le développement et la validation du réseau de neurones.

Pour le développement, les données de turbidité avant et après le traitement, sont utilisées afin de permettre l'apprentissage du réseau.

Et pour la validation, en se basant sur les données expérimentales du coagulant, la sortie du réseau a été évaluée.

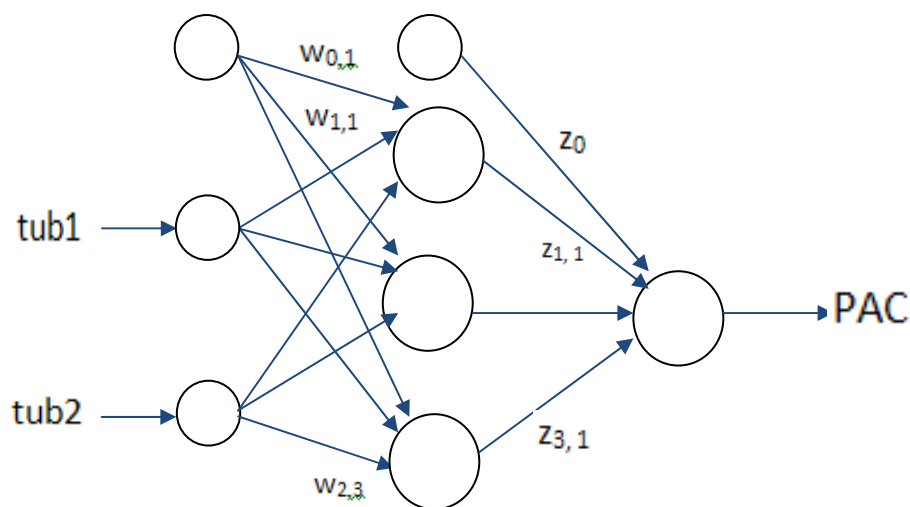


Figure 3.6 Perceptron multicouches de type '2-3-1'

Au départ les poids et les biais sont initialisés avec des valeurs aléatoires (distribution uniforme de moyenne nulle et variance 1/12).

Un algorithme d'optimisation minimise la fonction coût qui constitue une mesure d'écart entre la réponse réelle du réseau et la réponse désirée. Cette optimisation se fait de manière itérative en modifiant les poids en fonction du gradient

de la fonction de coût. Le gradient est estimé par une méthode spécifique aux réseaux de neurones appelée méthode de rétropropagation

Pendant la programmation nous avons suivi ces étapes pour la minimisation de cette fonction, et donc calculé le gradient de la fonction par rapport à chaque poids (de la couche de sortie ensuite ceux de la couche cachée) pour permettre une mise à jour des poids et des biais synaptiques, la convergence de ces derniers nécessite un certain nombre d'itérations, et un choix préalable du gain d'apprentissage.

Si ce dernier est très élevé, l'algorithme peut devenir instable, et s'il est trop petit, l'algorithme prendra un temps trop long pour converger.

Le réseau I possède deux entrées tub1 et tub2, 3 neurones dans la couche cachée et un neurone de sortie.

Avec un gain optimal pour une bonne stabilité ($\eta = 0.8$), différentes architectures sont testées pour déterminer le nombre adéquat de neurones dans la couche cachés.

Ce dernier est déterminé après une variation du nombre de ($i=1$ à 3), en calculant à chaque fois l'erreur moyenne quadratique (utilisation d'une fonction coût) pour les différentes architectures considérées. Pour ce faire, nous avons obtenu une variance minimale de test égale à $29.19 * 10^3$. pour un nombre d'itérations égal à 400.

Les fonctions d'activation sont de type sigmoïde $f(x) = \frac{1}{1 + \exp(-x)}$, dans la couche cachée ainsi que dans la couche de sortie. Ce type de fonctions a la propriété d'être différentiable ; les fonctions dérivées qui en résultent peuvent être exprimées facilement à l'aide de ces mêmes fonctions.

Le réseau I appréhendé est donc de type (2-3-1).

Les poids et les biais sont aléatoirement initialisés avant l'apprentissage, puis modifiés itérativement jusqu'à obtention d'un compromis satisfaisant entre la précision de l'approximation sur l'ensemble d'apprentissage.

La figure 3.7 montre l'initialisation aléatoire des poids et des biais

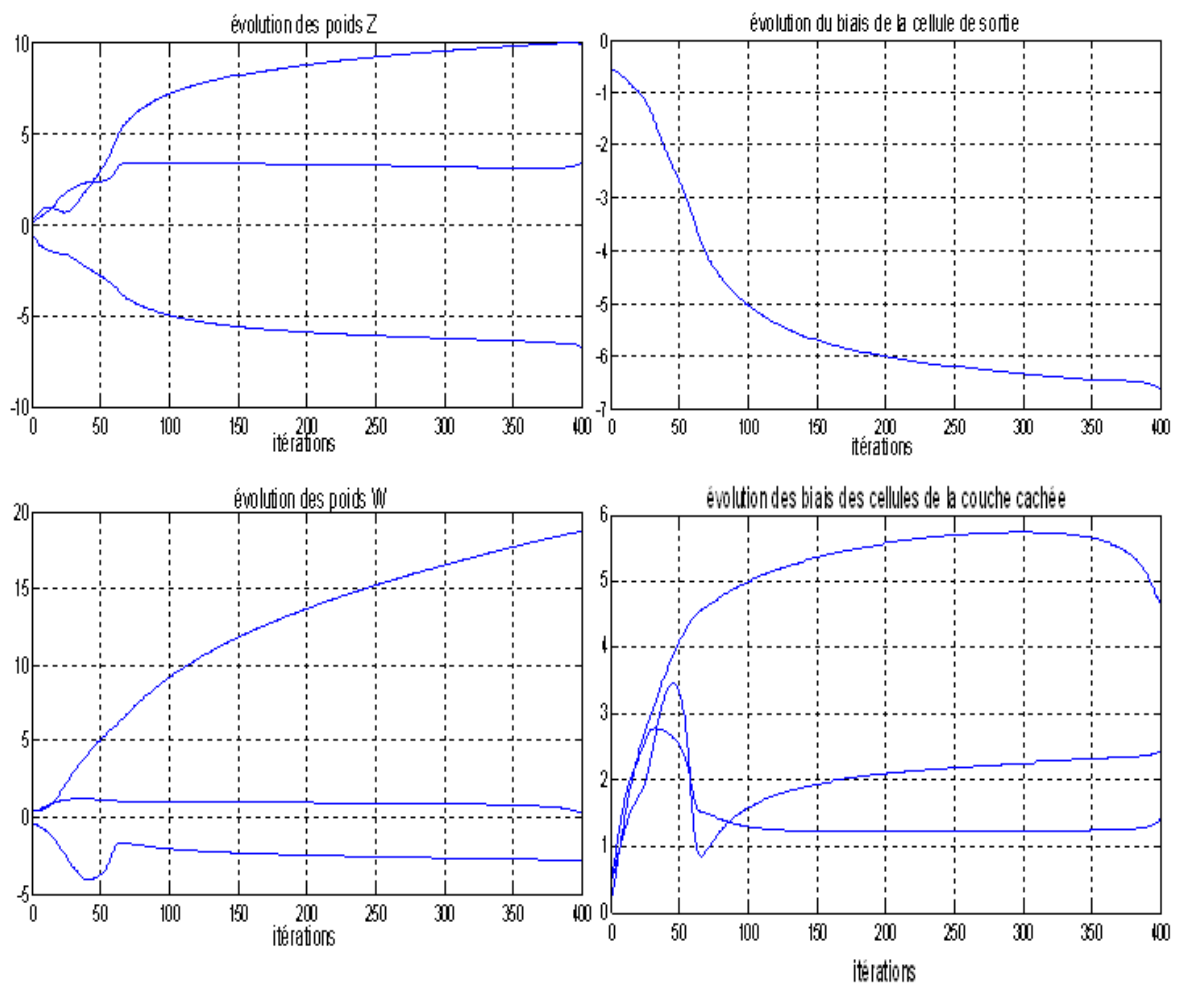


Figure 3.7 Initialisation aléatoire des poids et des biais

- Réinjection des poids et des biais dans l'algorithme d'apprentissage:

Les valeurs des poids et des biais obtenues après 400 itérations sont réinjectées dans le réseau I pour servir de nouvelles valeurs initiales.

La figure suivante montre la bonne convergence des poids et des biais après le même nombre d'itérations de 400.

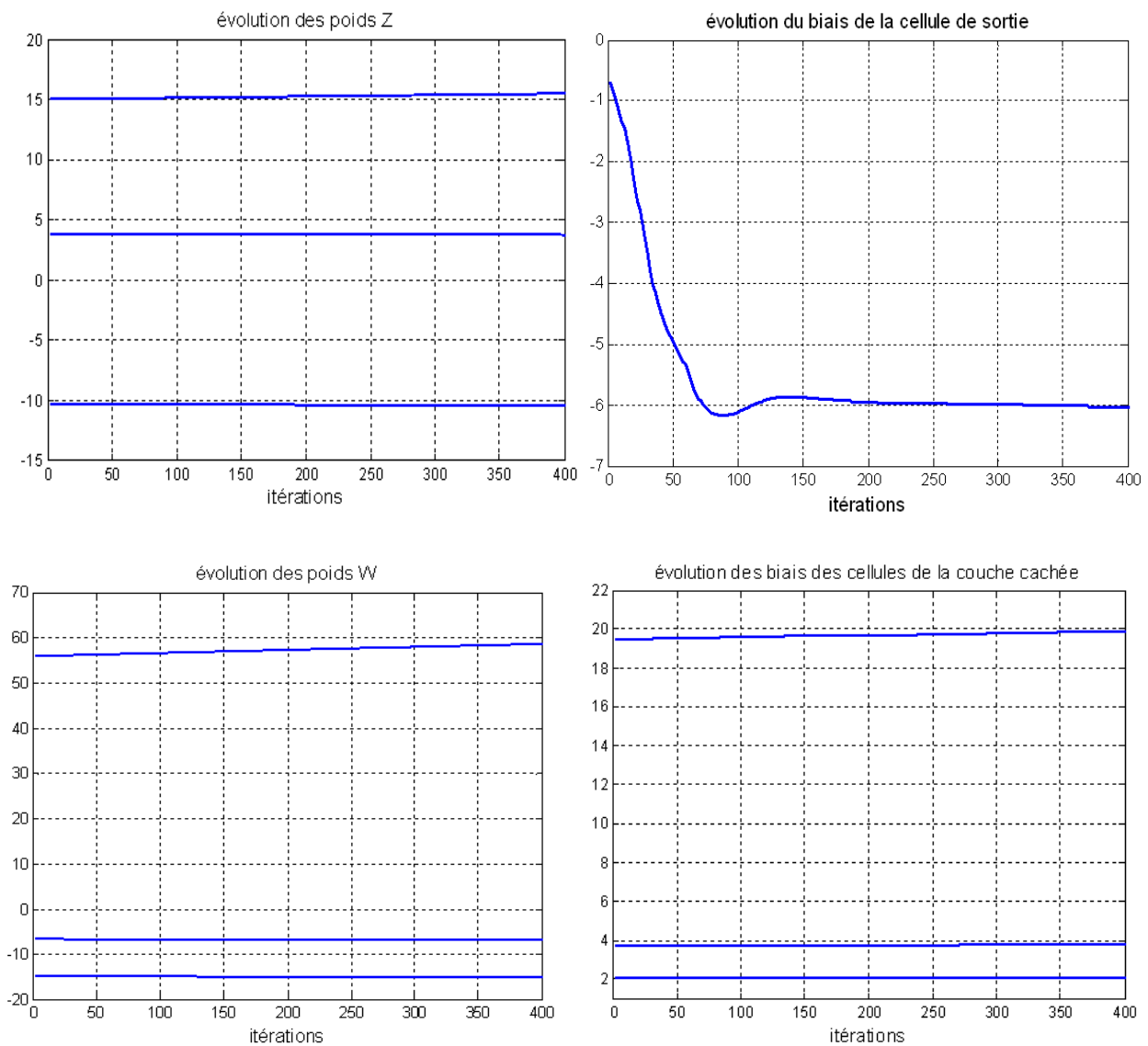


Figure 3.8 Evolution et convergence des poids et des biais

Comme les fonctions d'activation utilisées pour ce réseau sont des sigmoïdes unipolaires qui ont des valeurs en sortie entre 0 et 1, alors nous avons dû normaliser les données d'entrées /sortie fournies par l'expert.

La figure 3.9 représente la sortie normalisée donnée par le réseau I.

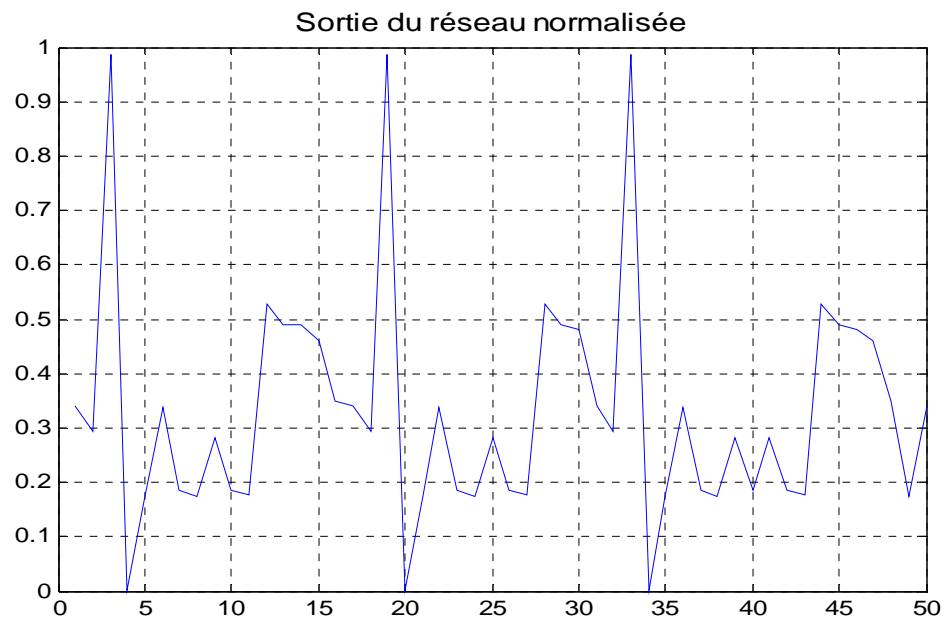


Figure 3.9 Sortie normalisée

Après dénormalisation de la sortie, la figure 3.10 montre les résultats du PAC donné par le modèle neuronal indiquant un indice de performance égal à 123.27.

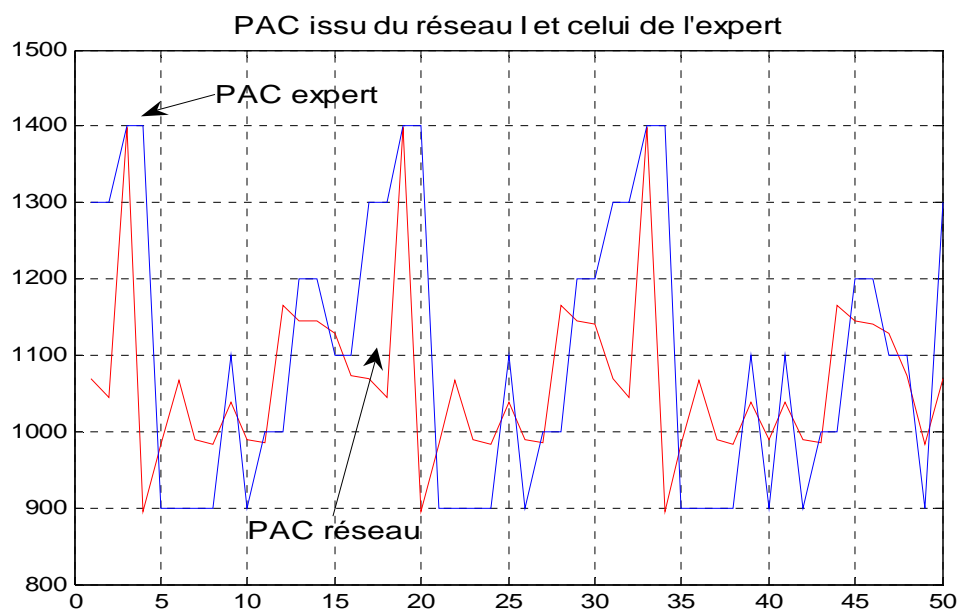


Figure 3.10 PAC réseau I et PAC expert

L'indice de performance est toujours important, pour le diminuer une deuxième architecture a été choisie, avec une troisième entrée par rapport au réseau I, définie comme l'erreur (tub2-tub1). Le réseau II (Cf. Figure 3.11) obtenu permettra de pouvoir spécifier la turbidité voulue en sortie du processus.

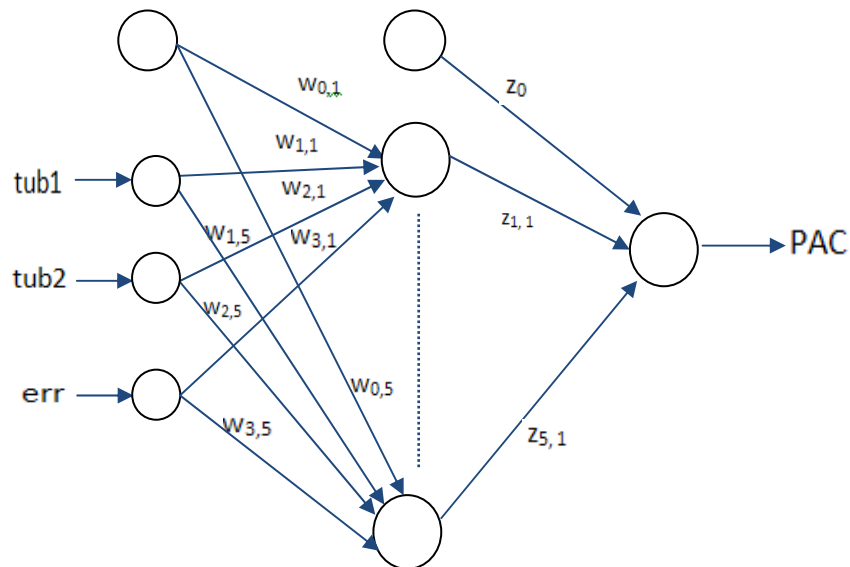


Figure 3.11 Perceptron multicouches de type '3-5-1'

Les poids et les biais qui proviennent du réseau II après stabilisation sont les suivants :

$$W = \begin{bmatrix} 2.3417 & 0.9562 & 2.6685 \\ 2.4366 & -0.3166 & 1.4683 \\ 18.2219 & -8.9528 & -7.7206 \\ -4.6742 & -17.9092 & 6.1279 \\ 2.9969 & -0.1680 & -0.6156 \end{bmatrix} ; w_0 = \begin{bmatrix} 4.4234 \\ 4.3937 \\ 10.6093 \\ 2.7231 \\ 2.7020 \end{bmatrix}$$

$$Z = [-2.0555 \quad -2.5439 \quad 10.2070 \quad -1.2184 \quad -2.3886] ; z_0 = -2.1138.$$

Dans le cas de l'apprentissage, tub2 est celle donnée par les essais de l'expert. Dans le cas de l'utilisation du réseau II, tub2 sera la consigne voulue.

L'architecture du réseau II utilise le même type de paramétrage avec 3 entrées, et 5 neurones dans la couche cachée (nombre correspondant à l'erreur moyenne quadratique la plus faible).

En modifiant le nombre de cellules cachées [20], la figure (3.12) montre la variance de l'erreur pour chaque cas, donnant le minimum de l'erreur quadratique avec 5 cellules dans la couche cachées.

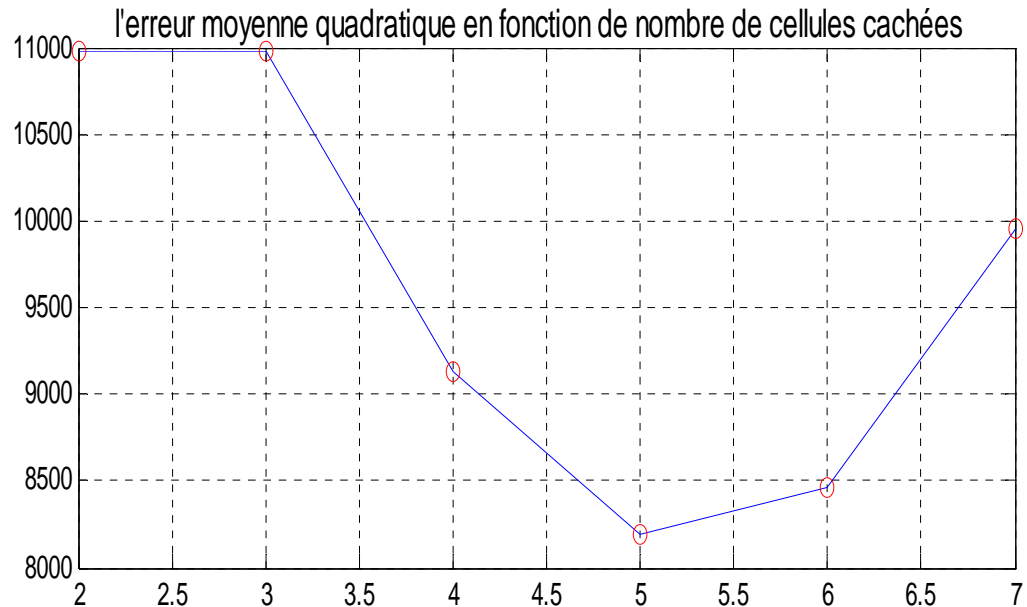


Figure 3.12 Variance de l'erreur PAC expert-PAC réseau II=f (nombre de neurones de la couche cachée)

3.11 Application du réseau II :

L'utilisation du réseau II a permis une bonne amélioration par rapport au réseau précédent et les valeurs du coagulant à rajouter sont plus proches de celles données par l'expert, avec une variance de $8.35 \cdot 10^3$ (Cf. Figure 3.13), ceci donne un gain de 70% par rapport au réseau I, et un indice de performance de 43.38.

La figure (3.14) montre l'amélioration du résultat donné par le réseau II par rapport à celui du réseau I ainsi que celui du modèle statique par rapport au modèle de l'expert.

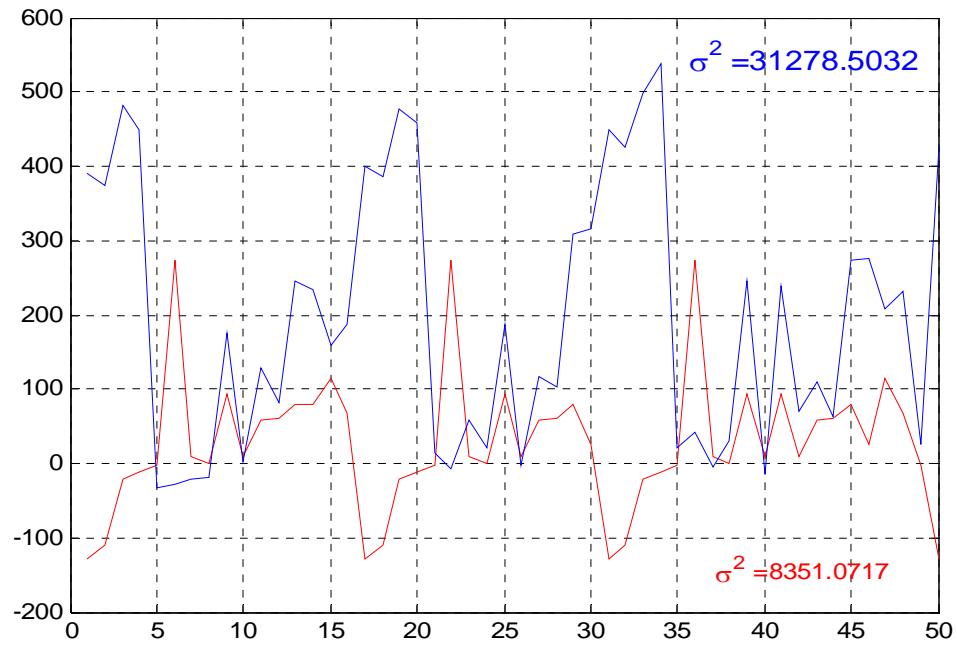


Figure 3.13 Moyenne de l'erreur entre PAC modèle neuronal/PAC expert, et PAC statique/PAC expert

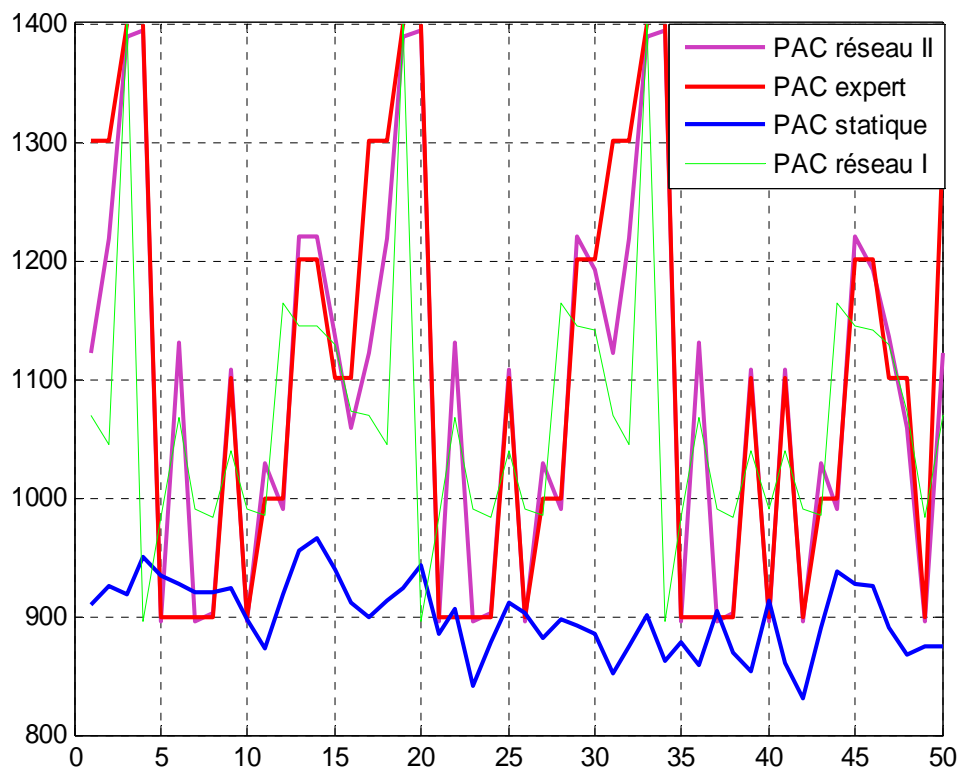


Figure 3.14 PAC statique et celui issu du réseau II, réseau I par rapport au PAC Expert

La troisième entrée du réseau a permis de réduire l'indice de performance ($I_p=43.38$) par rapport à celui calculé par le réseau I ($I_p =123.27$), cette diminution n'est pas encore optimale.

3.12 Modélisation neuronale du processus d'épuration

Le but de ce réseau est de modéliser le processus d'épuration des eaux par un réseau à partir du coagulant PAC, et les paramètres de l'environnement TE (température), pH (potentiel hydrogène) et AL (alcalinité) pour obtenir en sortie la turbidité de l'eau, cette turbidité sera comparée à celle donnée par les mesures de l'expert.

Le réseau III possède la structure suivante :

- 4 cellules d'entrées,
- 1 couche cachée de 4 cellules
- 1 sortie qui donne la turbidité de l'eau après le traitement.

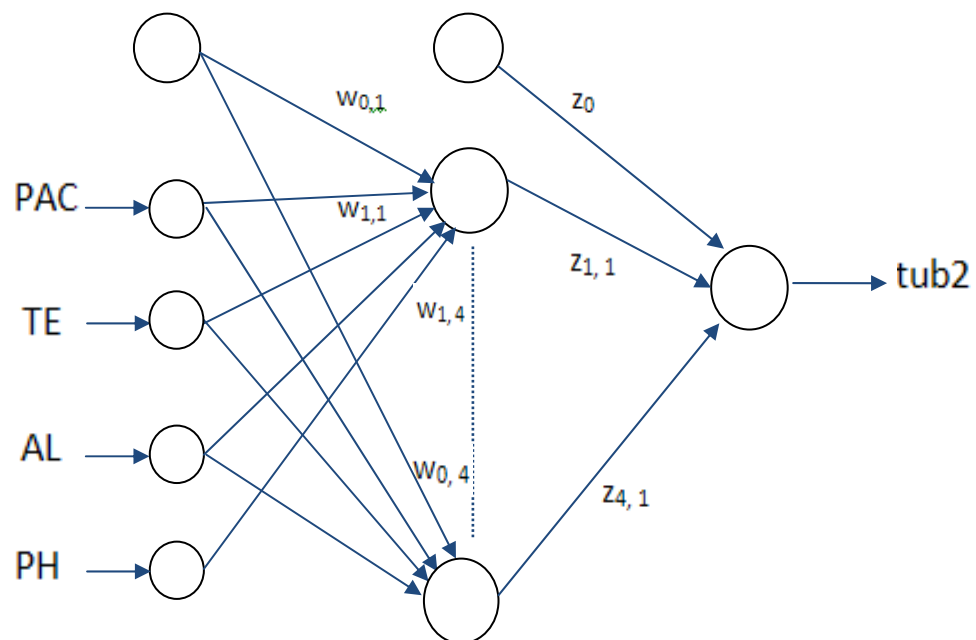


Figure 3.15 Perceptron multicouches de type '4-4-1'

L'apprentissage du réseau se fait après une initialisation aléatoire des poids et des biais.

Les figures suivantes montrent l'évolution des poids et des biais.

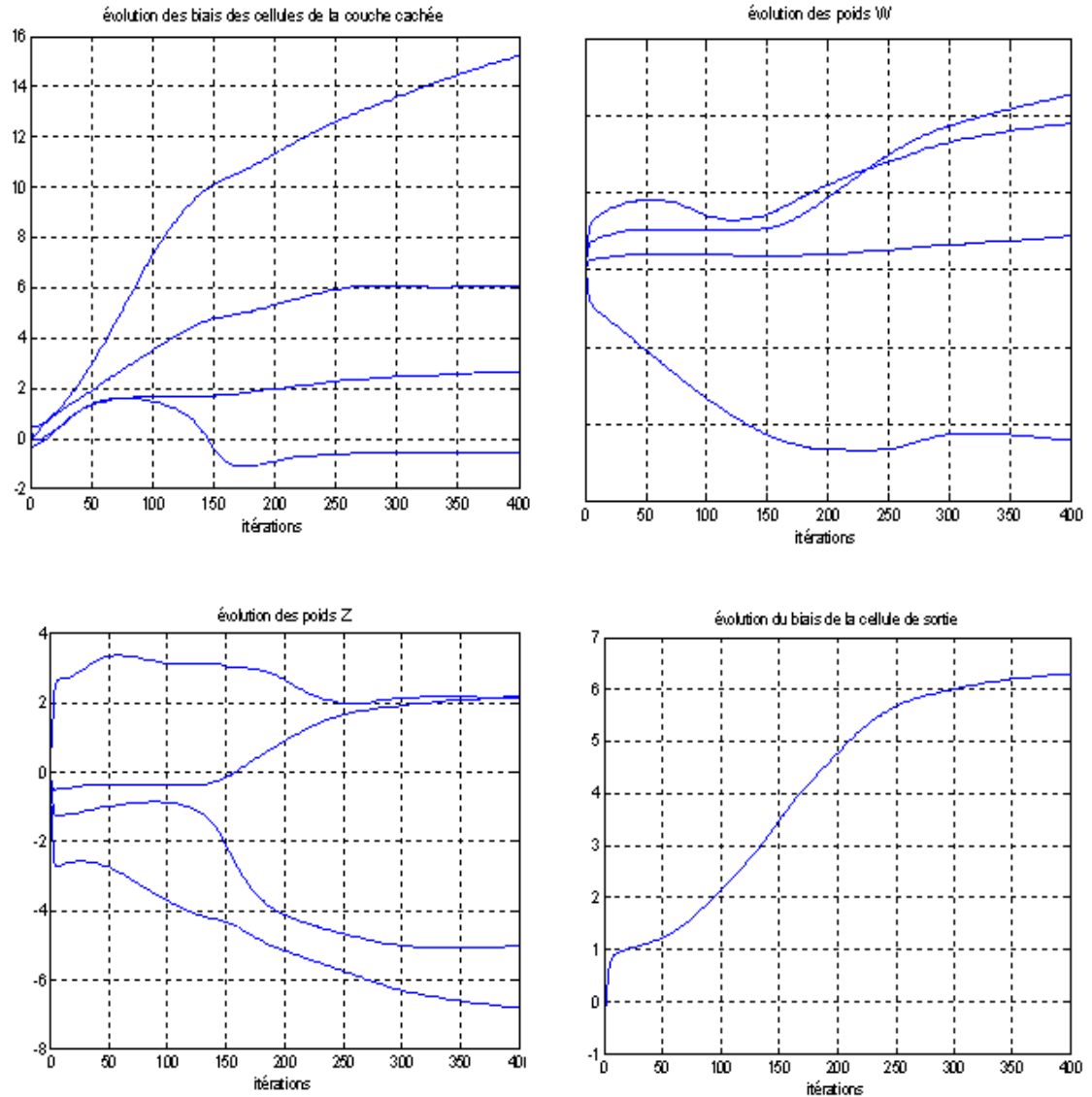


Figure 3.16 Initialisation aléatoire des poids W et Z et des biais w_0 et z_0

Au bout de ces 400 itérations, les poids et biais sont réinjectés pour servir de nouvelle valeur initiale.

La figure (3.16) montre l'évolution par la mise à jour des poids et des biais, jusqu'à stabilisation au bout de 400 itérations.

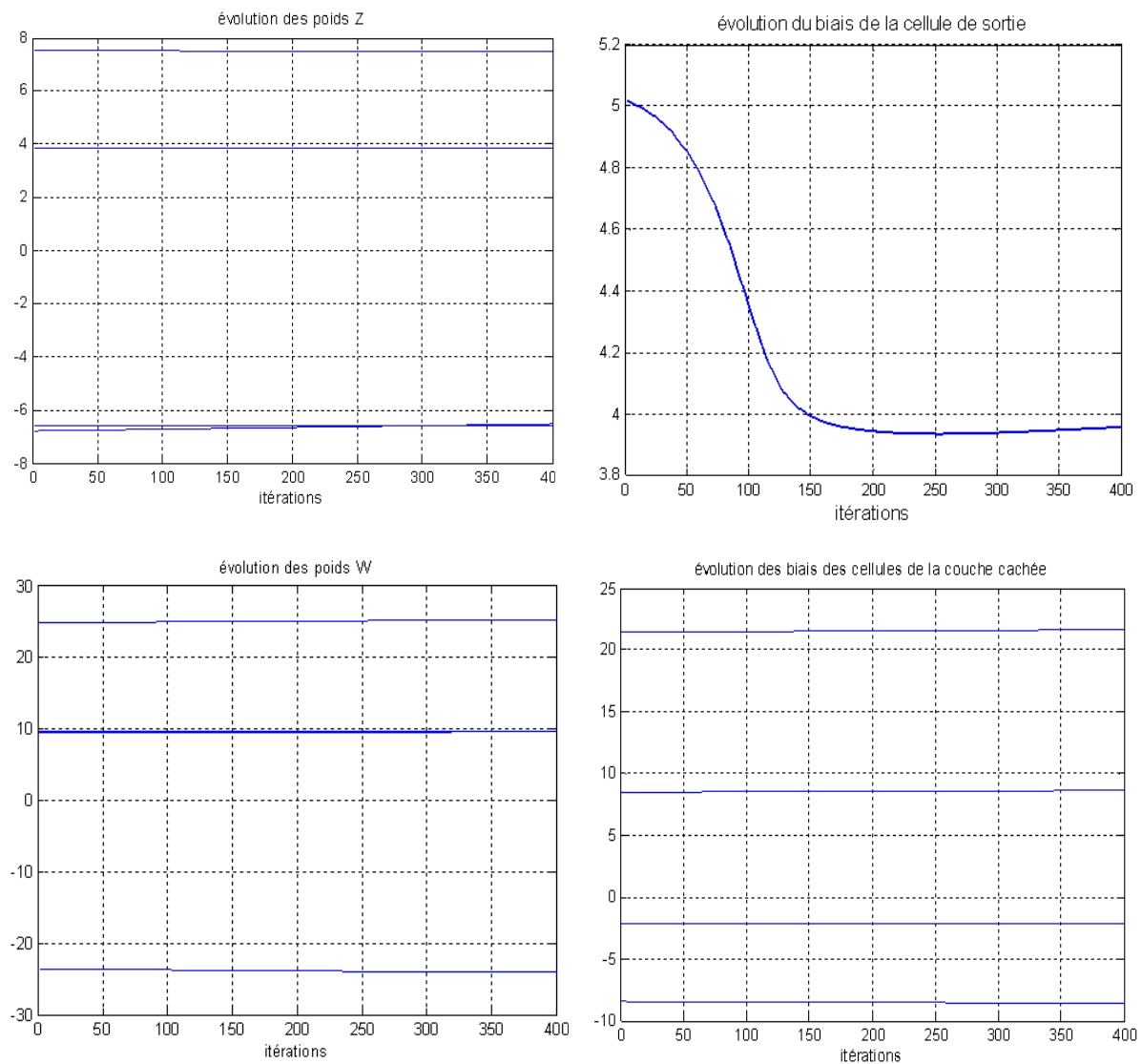


Figure 3.17 Evolution des poids et des biais

Ce réseau génère à sa sortie une turbidité qui indique un indice de performance égal à 0.33 avec une variance de 0.18, ce modèle a permis d'améliorer l'indice de performance correspondant à la turbidité (Cf. Figure 3.18).

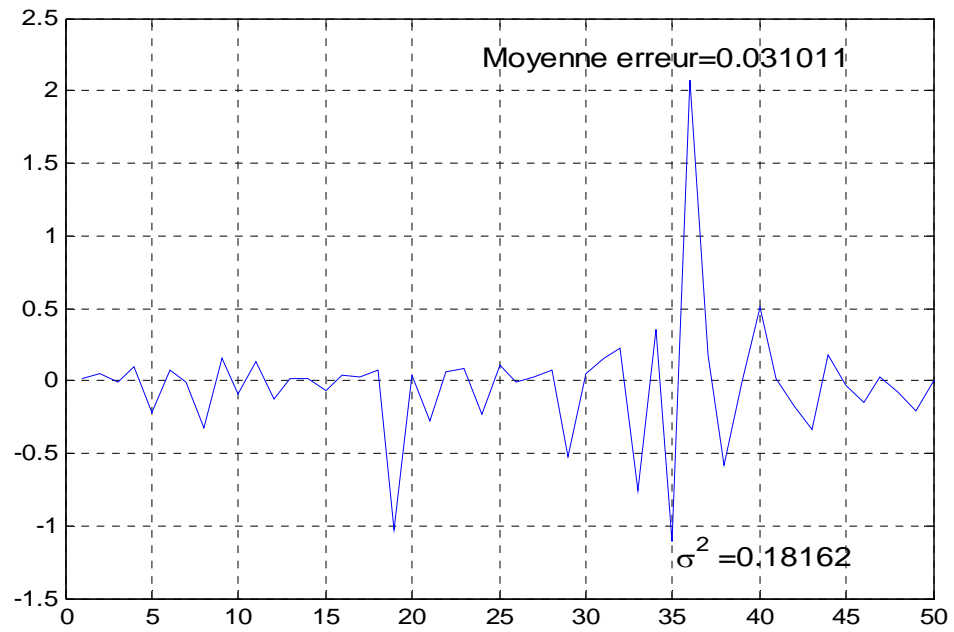


Figure 3.18 Erreur de la turbidité du réseau par rapport à celle donnée par l'expert

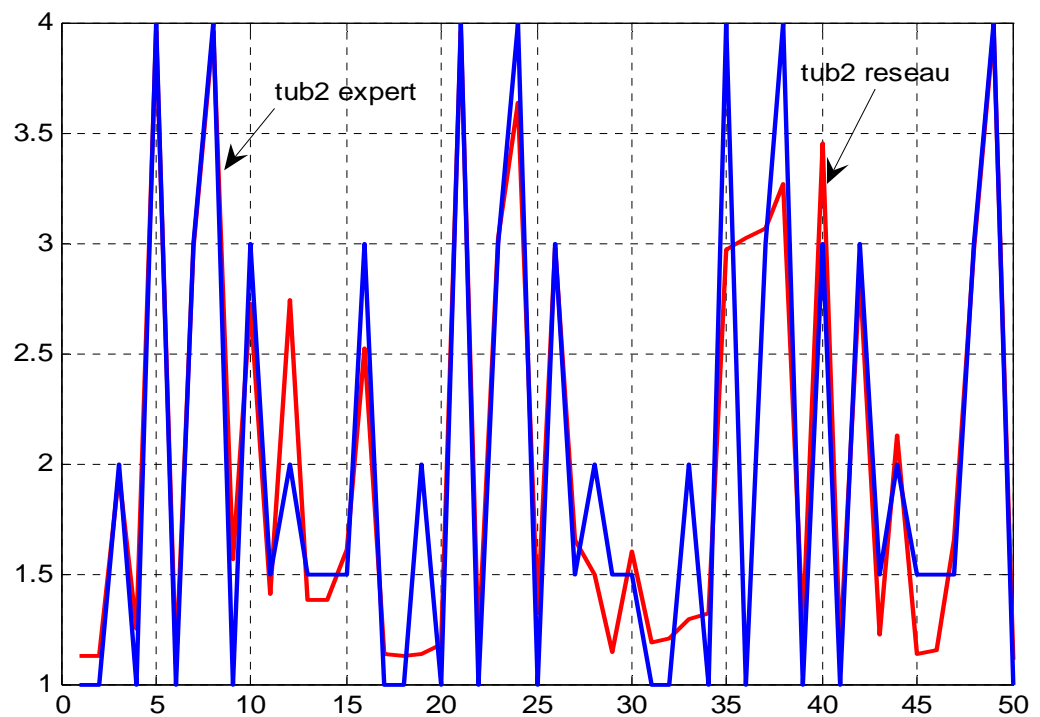


Figure 3.19 Turbidité donnée par l'expert et celle donnée par le réseau

Le tracé de la turbidité reste quand même différent par rapport à celui de l'expert (figure 3.19), une amélioration a été faite avec le modèle neuronal mais les entrées du réseau peuvent encore être modifiées

Le tableau 3.1 illustre l'indice de performance ainsi que la variance de la turbidité obtenue par les deux modèles :

	Indice de performance (ip)	Variance
Modèle ARMA	0.54	0.55
Modèle neuronal	0.33	0.18

Tableau 3.1 Indice de performances et variances

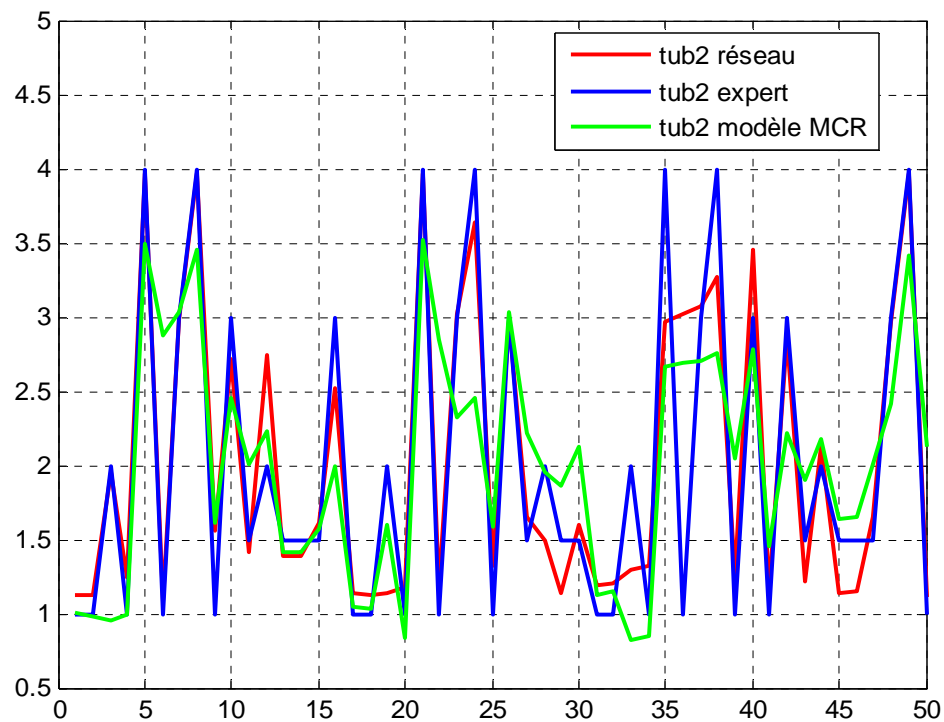


Figure 3.20 Tracé des courbes de turbidité

La figure (2.20) illustre l'amélioration de la modélisation neuronale par rapport à celle utilisée dans le chapitre précédent et cela peut être traduit à partir de la diminution de la variance ainsi que l'indice de performance (Cf. Tableau 3.1)

3.13 La mise en cascade

L'entrée PAC du processus d'épuration sera fournie par le réseau imitant les actions de l'expert, la figure 3.21 montre la mise en cascade des deux réseaux.

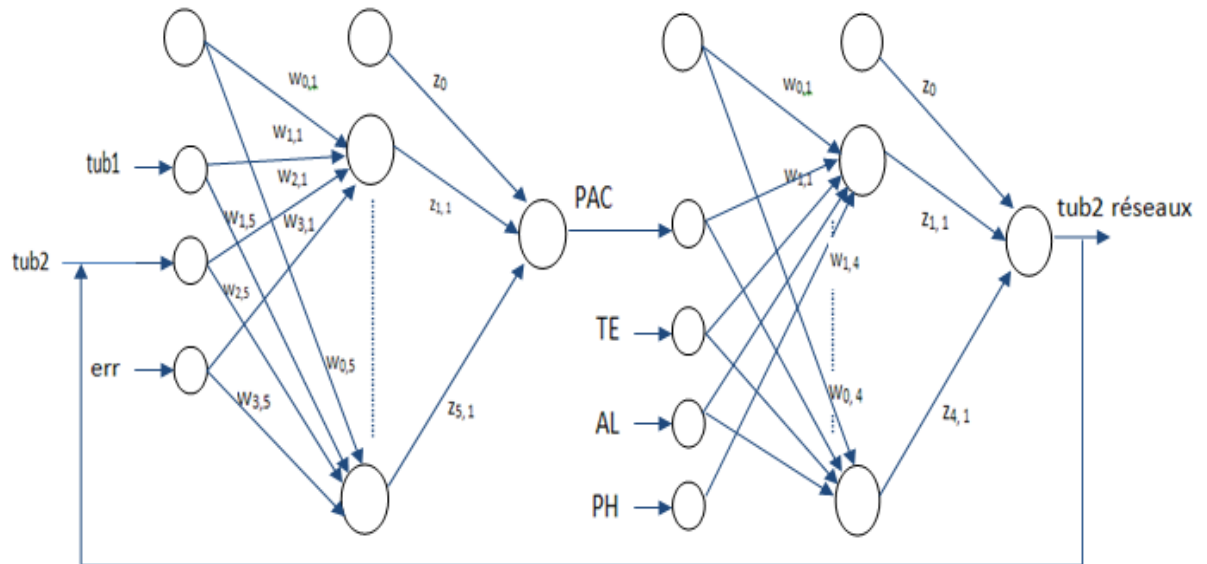


Figure 3.21 Mise en cascade du réseau d'expert et du réseau du processus d'épuration.

Les poids et les biais qui proviennent du réseau de l'opérateur (expert) sont :

$$W_{oper} = \begin{bmatrix} 2.3417 & 0.9562 & 2.6685 \\ 2.4366 & -0.3166 & 1.4683 \\ 18.2219 & -8.9528 & -7.7206 \\ -4.6742 & -17.9092 & 6.1279 \\ 2.9969 & -0.1680 & -0.6156 \end{bmatrix} ; w_{0oper} = \begin{bmatrix} 4.4234 \\ 4.3937 \\ 10.6093 \\ 2.7231 \\ 2.7020 \end{bmatrix}$$

$$Z_{oper} = [-2.0555 \quad -2.5439 \quad 10.2070 \quad -1.2184 \quad -2.3886] ; z_{0oper} = -2.1138.$$

Les poids et les biais qui proviennent du réseau processus sont:

$$W_{proc} = \begin{bmatrix} 15.7116 & -13.8005 & 6.6991 & 4.7455 \\ -0.4266 & -0.2469 & -0.6082 & 2.9318 \\ 9.1981 & -15.4295 & -5.2996 & -4.8806 \\ -9.0416 & 14.9218 & -3.2807 & -1.9378 \end{bmatrix} ; w_{0proc} = \begin{bmatrix} -6.1692 \\ 1.3326 \\ 16.0075 \\ -1.7741 \end{bmatrix}$$

$$Z_{proc} = [-4.4779 \quad 3.2533 \quad -8.3471 \quad -5.4648] ; z_0_{proc} = 7.6326 .$$

Le but de cette étape est la validation de l'imitation des actions de l'expert par le réseau « Expert » et le modèle neuronal du processus d'épuration comme schématisé dans la figure suivante :

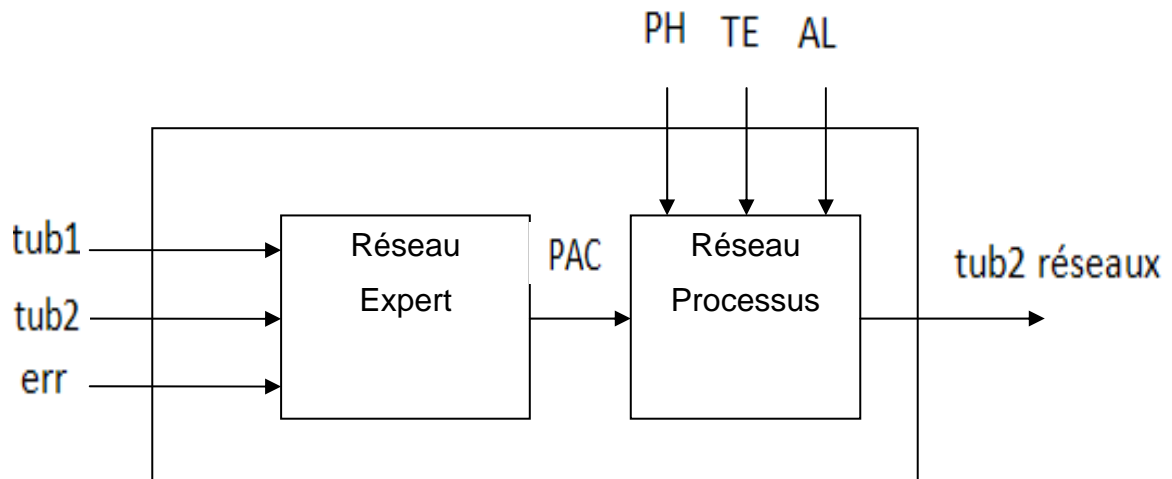


Figure 3.22 Schéma boîte noire de la mise en cascade

La mise en cascade des deux réseaux vue comme une boîte noire reçoit les valeurs de tub1 et tub2 du tableau 2.1 comme entrées ainsi que leur différence, notée $err = tub2 - tub1$ et comme sortie la turbidité (tub2 réseaux) que l'on va confronter à celle du Tableau 2.1.

Les mesures de l'environnement (TE, AL, PH) influent sur le modèle du processus d'épuration et interviennent donc, comme entrées du réseau neuronal du processus d'épuration en plus du PAC fourni par le réseau expert.

La figure (3.23) représente la courbe de la turbidité donnée par les actions de l'expert (Tableau 2.1) que l'on confronte à celle donnée par la mise en cascade des deux réseaux.

L'indice de performance calculé pour cette cascade est de l'ordre de 0.44 et une variance de 0.36.

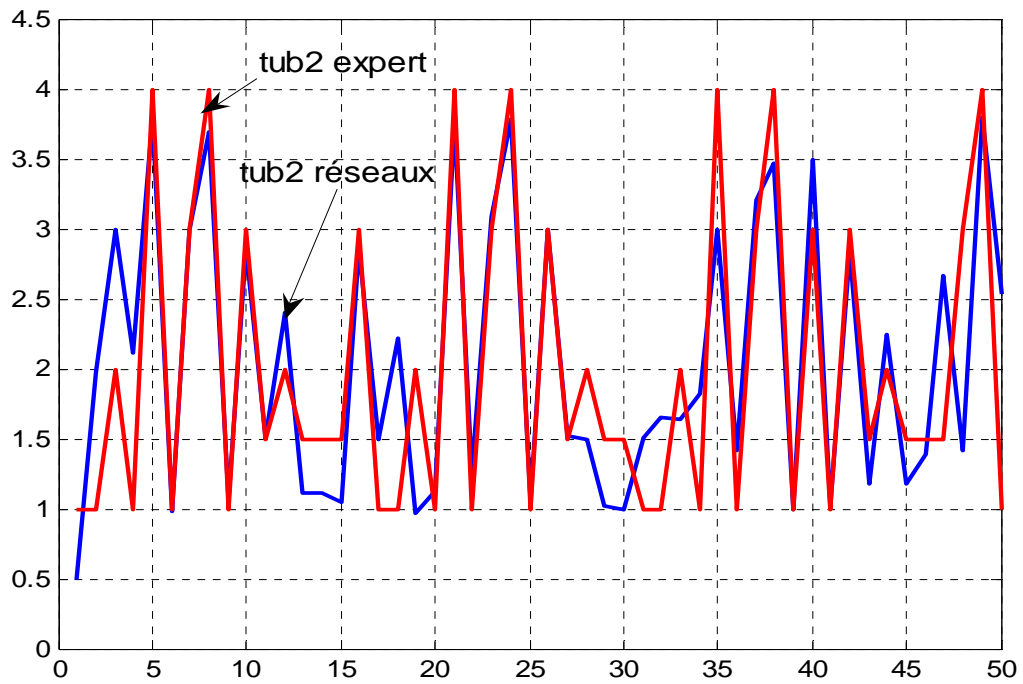


Figure 3.23 Confrontation des turbidités (mise en cascade).

La mise en cascade des deux réseaux a donné un résultat moins performant, l'indice de performance devient plus important, ceci est dû à la commande introduite par le réseau II (réseau opérateur).

3.14 Conclusion

Dans ce chapitre, des définitions essentielles relatives aux réseaux de neurones ont été introduites, ainsi qu'une distinction entre les réseaux de neurones non bouclés, qui réalisent des fonctions non linéaires et ceux qui sont bouclés, dynamiques, qui réalisent des équations aux différences non linéaires.

Nous avons aussi mis l'accent sur l'utilisation des réseaux de neurones comme outils de modélisation par apprentissage. Ces derniers permettent d'ajuster des fonctions non linéaires très générales à des ensembles de points. Comme toute méthode qui s'appuie sur des techniques statistiques, l'utilisation de réseaux de neurones nécessite que l'on dispose de données suffisamment nombreuses et représentatives.

L'étude des concepts fondamentaux de la modélisation neuronale, et leur application dans les procédés physico chimiques, a nécessité un type d'apprentissage supervisé.

L'application des réseaux de neurones dans la modélisation du processus d'épuration des eaux et celle des actions de contrôle du PAC par l'opérateur, a apporté une amélioration sur l'indice de performance relatif à chaque réseau, avec un temps de calcul réduit par rapport à la méthode précédente (ARMA).

Néanmoins, vu que le deuxième réseau neuronal modélise les actions floues, approximatives, de contrôle de la quantité de PAC à ajouter à l'eau non traitée, on a pensé à utiliser la technique de modélisation floue comme méthode de travail dans le chapitre suivant.

Une étude comparative des résultats obtenus par le modèle flou et par le modèle neuronal décidera du modèle final qui sera adopté.

Chapitre 4

Identification de modèles flous par la méthode de Takagi-Sugeno (TS) à partir de données d'entrées-sorties

4.1 Introduction

Dans ce chapitre, un traitement du problème d'identification est traité; construction des modèles flous à partir de données d'entrées-sorties.

Une autre approche de modélisation est définie, décrivant les techniques floues des systèmes, ceci consiste à établir leur description en termes mathématiques, c'est à dire donner le modèle mathématique du système physique.

Le développement des modèles mathématiques est un sujet central dans plusieurs disciplines des sciences et de l'ingénierie.

Traditionnellement, la modélisation est vue comme la double conjonction entre la compréhension de la nature et du comportement d'un système ainsi que le traitement mathématique approprié qui conduit à l'obtention d'un modèle utilisable.

Néanmoins, il est nécessaire de disposer d'une forte compréhension des phénomènes physiques de base quand on est confronté aux systèmes complexes. Ceci nécessite une modélisation qui consiste à proposer une structure appropriée pour le modèle et d'estimer ses paramètres, en utilisant habituellement ses données d'entrées-sorties représentatives du comportement, afin de d'identifier correctement les dynamiques et les non linéarités du système.

La techniques floues sur le modèle est de type Takagi-Sugeno, en utilisant ce formalisme, un comportement de la non linéarité du système est réalisé par une composition de règles du type « *Si-Alors* » est alors représenté.

Pour l'identification d'un tel système, utilisant une relation linéaire paramétrique entre ses entrées/sorties, l'identification de ce modèle flou est composée de deux parties :

- Identification de la structure du modèle flou.
- Identification des paramètres du modèle flou.

4.2 Techniques de modélisation floue et représentation du contrôleur flou

Classiquement, un contrôleur flou repose sur une structure [22], représentée à la figure 4.1, qui inclut :

- la fuzzification des variables d'entrée, avec éventuellement une normalisation,
- l'inférence à partir d'une base de règles,
- la défuzzification, avec éventuellement une dénormalisation.

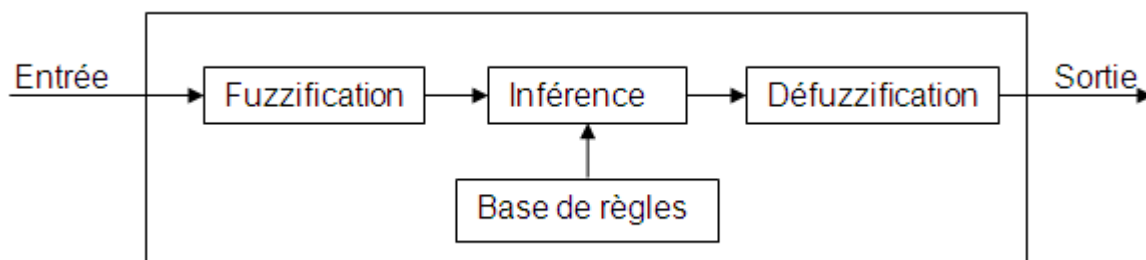


Figure 4.1 Représentation interne d'un contrôleur flou

La fuzzification est une étape dans laquelle, la grandeur d'entrée réelle est transformée en variable floue (fuzzifiée). L'inférence consiste à établir des règles floues qui permettent d'aboutir à la sortie désirée selon la base de connaissances.

Lors de l'étape de défuzzification (inverse de la fuzzification), les parties floues relatives aux sorties du mécanisme d'inférence sont converties en sorties numériques réelles.

4.3 Structure générale et différents types de modèles flous

En général les systèmes flous s'appuient sur une représentation de la connaissance sous forme de règles « Si-Alors » qui permettent de représenter les relations entre les variables d'entrée et de sortie dont l'expression générique est de la forme :

Si antécédent **Alors** conséquent

Dans un premier temps et afin de faciliter l'interprétation, l'antécédent (prémisse) est considéré comme une description linguistique qui indique les conditions de validité du phénomène représenté. Pour sa part, le conséquent (conclusion) représente le comportement associé aux conditions de validité décrites par l'antécédent.

Selon la structure particulière de la proposition « conséquent », trois types de modèles flous basés sur des règles sont définis [23].

Modèle flou linguistique (ou modèle Mamdani), dans lequel l'antécédent et le conséquent sont tous les deux des propositions floues qui utilisent des variables linguistiques [24][25].

Le modèle flou relationnel, peut être considéré comme une généralisation du modèle linguistique dans lequel il est possible d'associer une proposition «antécédent » spécifique avec plusieurs propositions de type « conséquent » différentes via une relation floue. Cette relation floue représente des associations entre les ensembles flous individuels définis dans les domaines d'entrées/sorties du modèle [26].

Le modèle flou Takagi-Sugeno (TS) [27], dans lequel le conséquent utilise des variables numériques plutôt que des variables linguistiques, sous la forme d'une

constante, d'un polynôme ou de manière plus générale d'une fonction ou d'une équation différentielle, dépendant des variables associées à la proposition «antécédent ».

Afin d'établir quelques éléments de base de la logique floue nécessaires, une règle linguistique R_i est de la forme générale de l'équation (4.1):

$$R_i: \text{Si } \mathbf{x} \text{ est } A_i \text{ Alors } \mathbf{y} \text{ est } B_i, i = 1, \dots, r \quad (4.1)$$

où r dénote le nombre de règles du modèle. $x \in X \subset R^p$ est la variable de l'*antécédent*, qui représente l'entrée du système et $y \in Y \subset R^q$ est la variable du *conséquent*, qui représente la sortie du système flou. Habituellement la dimension de l'espace du conséquent est prise comme $q = 1$. X et Y correspondent respectivement aux domaines (*univers de discours*) des variables d'entrées et de sorties. A_i et B_i sont des *termes linguistiques* qui correspondent aux valeurs qualitatives associées aux variables de base \mathbf{x} et \mathbf{y} . Ces termes linguistiques sont décrits par des ensembles flous définis par des *fonctions d'appartenance* μ , qui établissent une correspondance de chacun des univers de discours vers l'intervalle $[0, 1]$, ainsi :

$$\mu_{A_i}(x) : X \rightarrow [0, 1] , \mu_{B_i}(y) : Y \rightarrow [0, 1]$$

La théorie des ensembles flous [26] permet une appartenance graduelle d'un élément à un ensemble. Soit $\mu_{A_i}(\mathbf{x})$, le symbole de la valeur de la fonction d'appartenance (nommé le *degré d'appartenance*) de \mathbf{x} à l'ensemble caractérisé par A_i . Si le degré d'appartenance est égal à un, alors l'élément \mathbf{x} appartient complètement à l'ensemble (événement certain, réel).

Si celui-ci est égal à zéro, alors l'élément \mathbf{x} n'appartient pas à cet ensemble. Si le degré d'appartenance se trouve entre zéro et un, alors \mathbf{x} appartient partiellement à l'ensemble flou.

Le degré d'appartenance permet alors une gradation du degré de constatation des faits.

Typiquement, plusieurs formes de fonctions d'appartenance sont utilisées dans les applications. Elles peuvent être définies par le développeur du modèle (l'expert) en utilisant la connaissance préalable ou bien l'expérimentation.

Les contrôleurs flous s'appuient sur une représentation de la connaissance sous forme de règles dont l'expression générique est [28]:

SI conditions ALORS conclusions

Il est nécessaire de distinguer les contrôleurs de Mamdani et ceux de Takagi-Sugeno. Pour ces deux types de contrôleurs, la partie prémisse des règles utilise des variables linguistiques. Ce concept a été très largement développé par Zadeh. C'est par la partie conclusion que ces contrôleurs peuvent être distingués. Ainsi les contrôleurs de Mamdani utilisent des variables linguistiques dans les conclusions [28], tandis que celles des contrôleurs de Takagi-Sugeno sont numériques. Les exemples suivants correspondent respectivement aux deux types de contrôleurs.

4.3.1 Contrôleur de type Mamdani

SI température est basse **ET** luminosité est faible **ALORS** chauffage est grand

SI T est B **ET** L est F **ALORS** C est G

T , L et C sont les variables linguistiques représentant, respectivement la température, la luminosité et le chauffage.

B , F et G sont des valeurs linguistiques, basses de la température, faibles de la luminosité et grandes pour le chauffage [28].

4.3.2 Contrôleur de type Takagi-Sugeno

SI température est froide et luminosité est faible **ALORS** $u = 1.5 - 5 * T + 2 * L$

Les contrôleurs de Takagi-Sugeno ne comportent pas d'étape de défuzzification.

L'intérêt de cette modélisation par les techniques est tout particulièrement propre à ce type de contrôleurs.

4.4 Modélisation floue type Takagi-Sugeno

D'une manière générale les contrôleurs de types Takagi-Sugeno forment une famille particulière de contrôleurs flous, qui utilisent des règles de contrôle dont l'antécédent est une description d'un sous-espace flou d'entrée. La conséquence est une relation d'entrées/sorties linéaire paramétrique. La construction de tels modèles par les données d'entrées/sorties est caractérisée par :

1- l'instrumentation mathématique est nécessaire pour avoir la simplicité et la généralité.

2- la méthode d'identification est constituée de deux parties :

- l'identification de la structure du modèle flou.
- l'identification des paramètres du modèle flou.

4.5 Algorithme d'identification

Le modèle considéré est constitué d'un certain nombre de règles de production du format suivant :

If x_1 is A_1 and ... and x_k is A_k then $y = a_0 + a_1x_1 + \dots + a_kx_k$

Caractérisé par les connecteurs "and" et une équation linéaire.

L'identification des trois points suivants est nécessaire :

1. x_1, \dots, x_k : variables linguistiques dans l'antécédent.
2. A_1, \dots, A_k : valeurs linguistiques.
3. a_0, \dots, a_k : paramètres des conséquences.

Les variables d'entrée peuvent ne pas toutes apparaître dans l'antécédent.

Les points 1 et 2 sont relatifs à la partition de l'espace des variables d'entrée dans certains sous-espace flous.

Le point 3 est relatif à la description d'une relation d'entrée-sortie dans chaque sous espace flou.

De ce fait l'algorithme d'identification des règles de production est divisé en trois étapes correspondantes aux trois points ci-dessus.

4.5.1 Identification de la structure du modèle flou type Takagi Sugeno

Le modèle flou est constitué de deux parties :

- Détermination des variables linguistiques qui apparaissent dans la prémisse.
- Détermination de la partition optimale de l'espace d'entrée.

La détermination des variables qui apparaissent dans la prémisse est évaluée selon cet algorithme.

Soit un modèle flou de k entrées (x_1, \dots, x_k) et une seule sortie y

- Etape1:

Le sous-espace de x_1 est divisé en deux sous espaces flous « grand » et « petit » et les sous espaces des autres variables (x_2, \dots, x_k) ne sont pas divisés, ce qui veut dire que seul x_1 apparaît dans l'antécédent **[27] [29]**.

Le modèle constitué de deux règles de production est donc :

If x_1 is big 1 then ...

If x_1 is small 1 then ...

Ce modèle est appelé modèle 1-1, de même un modèle dans lequel le sous espace de x_2 est divisé, et les sous espaces des autres variables x_1, x_3, \dots, x_k ne le sont pas est appelé modèle 1-2.

De la même manière, on obtient les k modèles dont chacun est composé de deux règles de production.

En général le modèle 1-i est de la forme :

If x_i is big i then ...

If x_i is small i then ...

- Etape 2:

Dans chaque modèle, les paramètres optimaux de l'antécédent et de la conséquence sont trouvés par l'algorithme décrit dans les sections précédentes.

Le modèle avec le plus petit indice de performance est choisi en dehors de k modèles. Ce modèle est appelé état stable.

- Etape 3 :

Soit le modèle 1-i stable à l'étape 1, où seule la variable x_i apparaît dans l'antécédent.

Toutes les combinaisons de $x_i - x_j$ ($j = 1, 2, \dots, k$) sont présent en considération et le sous espace de chaque variable est divisé en deux sous espaces flous.

Pour la combinaison $x_i - x_j$, le sous espace de x_i est divisé en quatre sous espaces flous, par exemple « grand », « moyen grand », « moyen petit » et « petit » à la fin, les k modèles seront de la forme du modèle 2-j, chacun est constitué de 2*2 règles de productions, ensuite le modèle avec le plus petit indice de performance est déterminé, exactement comme dans l'étape 2, il sera nommé aussi l'état stable de cette étape.

- Etape 4 :

Répétez l'étape 3 de la même façon en ajoutant une autre variable dans l'antécédent (CF. Figure 4.2)

- Etape 5 :

La recherche est arrêtée si l'un des critères suivant est satisfait :

- L'indice de performance d'un état stable devient supérieur à une valeur prédéterminée.
- Le nombre de règles de productions d'un état stable dépasse le nombre prédéfini qui est 2^k

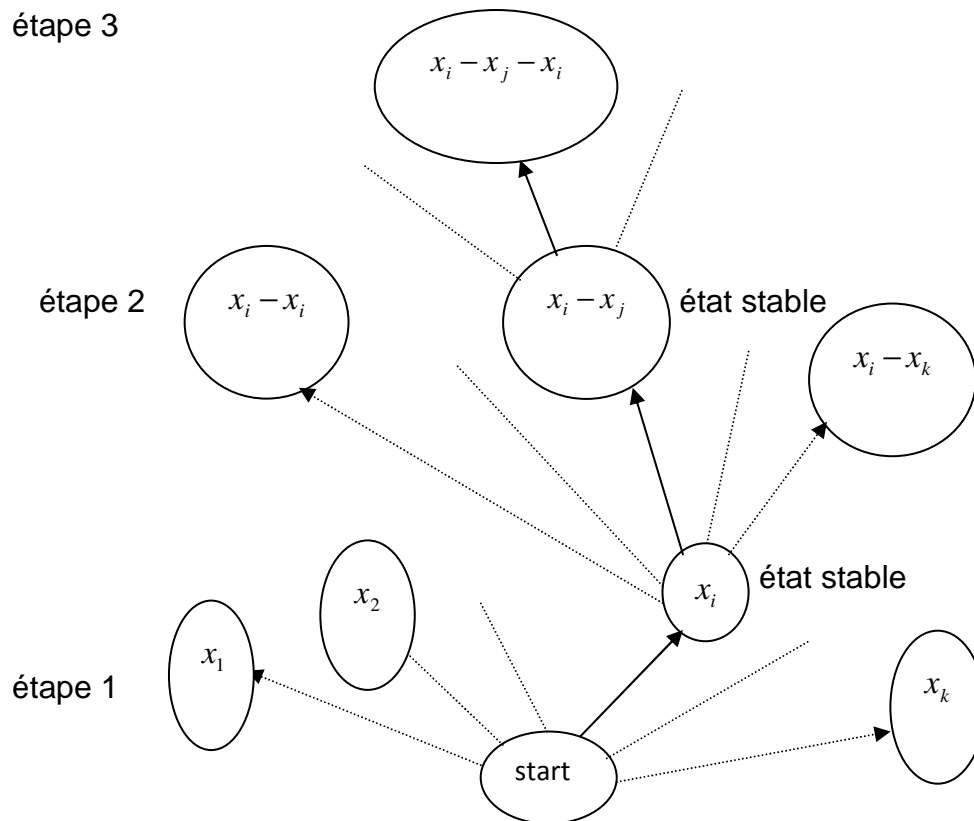


Figure 4.2 Choix des variables linguistiques

4.5.2 Identification des paramètres du modèle flou

Le modèle flou Takagi-Sugeno (TS), dans lequel le conséquent utilise des variables numériques plutôt que des variables linguistiques, sous la forme d'une constante, d'un polynôme ou de manière plus générale d'une fonction ou d'une équation différentielle, dépendant des variables associées à la proposition antécédent **[26][30]**.

Le modèle flou de type Takagi-Sugeno est construit à partir d'une base de règles « si...alors... », dans lesquelles la prémisse est exprimée linguistiquement et les conséquences peuvent s'exprimer, sous la forme d'une constante, d'un polynôme ou plus généralement sous la forme d'une équation différentielle dépendant des variables associées à l'antécédent.

D'une manière générale, un modèle de type Takagi-Sugeno (TS) est basé sur une collection des règles R_i du type :

$$R_i : \mathbf{Si} \ x_1 \text{ est } A_{i1} \text{ et } x_2 \text{ est } A_{i2} \text{ et...} x_p \text{ est } A_{ip} \ \mathbf{Alors} \ y_i = f_i(x)$$

Avec $i = 1, \dots, N$.

Typiquement, les fonctions f_i sont choisies comme des fonctions paramétrées appropriées, avec la même structure pour chaque règle où seuls les paramètres varient. Une forme de paramétrisation souvent utilisée est la forme affine:

$$y_i = a_i^T x + d_i$$

Où : $a_i \in \mathbb{R}^p$ est un vecteur de paramètres et d_i sont des scalaires. Ce modèle est appelé le modèle affine de Takagi-Sugeno.

Les conclusions des règles dans ce modèle sont alors des hyperplans (sous-espaces linéaires p -dimensionnels) dans l'espace \mathbb{R}^{p+1} [26].

Ainsi, en modélisation floue des systèmes, l'antécédent de chaque règle définit une région (floue) de validité pour le sous-modèle correspondant au conséquent.

Avant de pouvoir inférer la sortie, il faut calculer d'abord le degré d'appartenance $\mu(x)$ de l'antécédent.

$$R_i : \mathbf{Si} \ (x_1 \text{ est } X_{i1}) \text{ et...et } (x_n \text{ est } X_{in}) \ \mathbf{Alors} \ Y_i = a_{i0} + a_{i1}x_1 + \dots + a_{in}x_n \quad (4.2)$$

x_1, \dots, x_n : variables linguistiques de l'antécédent.

X_{i1}, \dots, X_{in} : valeurs linguistiques.

a_{i0}, \dots, a_{in} : paramètres des conséquences.

En appliquant la formule (4.2) à toutes les règles, on obtient :

$$\begin{cases} Y_1 = a_{10} + a_{11}x_1 + \dots + a_{1n}x_n \\ Y_2 = a_{20} + a_{21}x_1 + \dots + a_{2n}x_n \\ Y_3 = a_{30} + a_{31}x_1 + \dots + a_{3n}x_n \\ \vdots \\ Y_N = a_{N0} + a_{N1}x_1 + \dots + a_{Nn}x_n \end{cases}$$

Le degré d'appartenance **[31]**, correspondant à ces sorties est donné par la règle générale (Cf. Equation 4.3):

$$\mu_y(y_i) = \text{Sup}\{\mu_{x_1}(x_1) \wedge \dots \wedge \mu_{x_n}(x_n)\}, \quad (4.3)$$

$$Y_i = a_{i0} + a_{i1}x_1 + \dots + a_{in}x_n,$$

$$i = 1, \dots, N.$$

La sortie finale, se calcule comme la moyenne des sorties selon les règles R_i **[27]**, pondérées par l'expression (4.1):

$$y = \frac{\sum_{i=1}^N \mu_{Y_i}(y_i) \cdot y_i}{\sum_{i=1}^N \mu_{Y_i}(y_i)} \quad (4.4)$$

$$\begin{cases} R_1 : \text{Si } (x_1 \text{ est } X_{11}) \text{ et } \dots \text{ et } (x_n \text{ est } X_{1n}) \text{ Alors } Y = a_{10} + a_{11}x_1 + \dots + a_{1n}x_n, \\ \vdots \\ R_N : \text{Si } (x_1 \text{ est } X_{N1}) \text{ et } \dots \text{ et } (x_n \text{ est } X_{Nn}) \text{ Alors } Y = a_{N0} + a_{N1}x_1 + \dots + a_{Nn}x_n, \end{cases}$$

$$\text{avec } i=1, \dots, N.$$

La sortie finale pour les entrées x_1, \dots, x_n est donnée par l'équation (4.5):

$$y = \frac{\sum_{i=1}^N (\mu_{x_{i1}}(x_1) \wedge \dots \wedge \mu_{x_{in}}(x_n)) (a_{i0} + a_{i1}x_1 + \dots + a_{in}x_n)}{\sum_{i=1}^N (\mu_{x_{i1}}(x_1) \wedge \dots \wedge \mu_{x_{in}}(x_n))} \quad (4.5)$$

Où :

$$\mu_{x_{ij}}(x_j) = \mu_{x_j}(x_j) \text{ est le degré d'appartenance de } x_j \text{ obtenu dans la règle}$$

$$R_i, i=1, \dots, N, j=1, \dots, n.$$

Soit:

$$\beta_i = \frac{\mu_{xi1}(x_1) \wedge \dots \wedge \mu_{xin}(x_n)}{\sum_{i=1}^N (\mu_{xi1}(x_1) \wedge \dots \wedge \mu_{xin}(x_n))} \quad i = 1, \dots, N \quad (4.6)$$

Ce qui donne :

$$y = \sum_{i=1}^N \beta_{in} (a_{i0} + a_{i1}x_1 + \dots + a_{in}x_n)$$

$$= \sum_{i=1}^N [\beta_{in} \beta_{in} x_1 \dots \beta_{in} x_n] \begin{bmatrix} a_{i0} \\ a_{i1} \\ \vdots \\ a_{in} \end{bmatrix} \quad (4.7)$$

Si l'ensemble des données d'entrée/sortie est donné [30], [31] alors:

$$\begin{aligned} & x_1^1, x_2^1, \dots, x_n^1, y^1; \\ & x_1^2, x_2^2, \dots, x_n^2, y^2; \\ & \vdots \\ & x_1^M, x_2^M, \dots, x_n^M, y^M. \end{aligned} \quad (4.8)$$

Les paramètres de coefficients $\{a_{i0}, a_{i1}, \dots, a_{in} / i = 1, \dots, N\}$ sont identifiés par la méthode des moindres carrés [30], [31] à partir de l'ensemble de données (Cf. Equation 4.8).

$$\Lambda \theta = b \quad (4.9)$$

$$\Lambda = \Lambda_{M * (N * (n+1))}$$

Après identification des paramètres, la sortie finale correspond à l'expression donnée par l'équation (4.5).

$$\begin{bmatrix} \beta_{11} \dots \beta_{N1} x_1^1 \beta_{11} \dots x_1^1 \beta_{N1} \dots n_n^1 \beta_{11} \dots x_n^1 \beta_{N1} \\ \vdots \\ \vdots \\ \vdots \\ \beta_{1M} \dots \beta_{NM} x_1^M \beta_{1M} \dots x_1^M \beta_{NM} \dots n_n^M \beta_{1M} \dots x_n^M \beta_{NM} \end{bmatrix} \quad (4.10)$$

$$\theta = [a_{10} \dots a_{N0} \quad a_{11} \dots a_{N1} \dots a_{1n} \dots a_{Nn}]_{(N*(n+1))*1}^T \quad (4.11)$$

$$b = [y^1 \quad y^2 \quad \dots \quad y^M]_{M*1}^T \quad (4.12)$$

$$\beta_{kj} = \frac{\mu_{xk1}(x_1^j) \wedge \dots \wedge \mu_{xkn}(x_n^j)}{\sum_{k=1}^N (\mu_{xk1}(x_1^j) \wedge \dots \wedge \mu_{xkn}(x_n^j))} \quad k = 1, \dots, N; j = 1, \dots, M$$

L'équation (4.6) permet d'écrire: $b = \Lambda \theta$

Donc l'estimation du vecteur paramètre θ , par la méthode des moindres carrés, est donnée par :

$$\hat{\theta} = [\Lambda^T * \Lambda]^{-1} \Lambda^T b. \quad (4.13)$$

4.6 Application à la modélisation floue

4.6.1 Application au processus d'épuration des eaux

La technique précédemment présentée, sera appliquée dans ce qui suit, pour la modélisation des actions de contrôle de la quantité du coagulant (Polychlorure basique d'aluminium : PAC) à ajouter à l'eau de surface, non traitée pour que la turbidité de l'eau soit maintenue au dessous d'un certain niveau de tolérance.

Il est toute fois difficile de mettre la quantité adéquate du coagulant sachant que la turbidité de l'eau dépend de plusieurs facteurs pouvant la favoriser.

Par exemple, si TE est faible, alors plus de PAC est nécessaire, par contre pH et AL affectent de façon non linéaire la quantité du PAC nécessaire, le PAC optimal dépend de ces variables.

La relation entre ces variables n'est pas claire. Notons qu'il existe d'autres variables qui influent sur le processus mais qui ne sont pas toujours mesurables; par exemple le nombre de planctons contenus dans l'eau de surface qui augmente au printemps et qui ne peut être mesuré les autres saisons.

Selon l'algorithme d'identification de la structure du modèle flou discuté précédemment, l'univers de discours de chaque variable qui apparaît dans la prémisse sera exprimé linguistiquement par rapport aux mesures prises avant le traitement.

Étape1 :

Le sous-espace de *PH* est divisé en deux sous espaces flous « grand » et « petit » (Cf. Figure 4.3), alors que les sous espaces des autres variables (*TE, AL, TB1, TB2*) ne sont pas divisés, ce qui veut dire que seul *PH* apparaît dans l'antécédent.

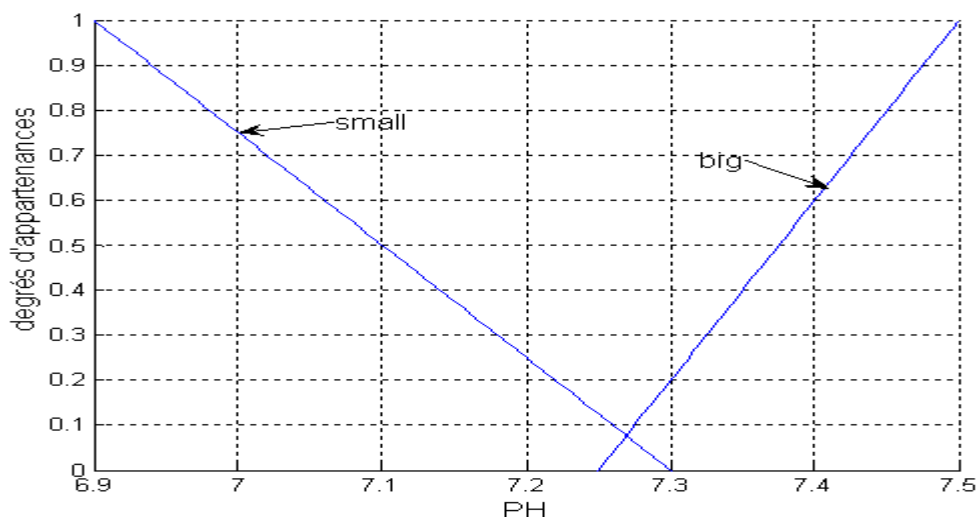


Figure 4.3 Fonction d'appartenance de la variable PH

Le modèle constitué de deux règles de production est donc :

If PH is small then $PAC = a_{10} + a_{11} * PH + a_{12} * TE + a_{13} * AL + a_{14} * TB1 + a_{15} * TB2$

If PH is big then $PAC = a_{20} + a_{21} * PH + a_{22} * TE + a_{23} * AL + a_{24} * TB1 + a_{25} * TB2$

Ce modèle est appelé modèle 1-1, de même un modèle dans lequel le sous espace de TE est divisé, et les sous espaces des autres variables ($PH, AL, TB1, TB2$) ne le sont pas, est appelé modèle 1-2.

De la même manière, les 5 modèles sont obtenus, dont chacun est composé de deux règles de production.

Etape 2 :

Pour chaque modèle, les paramètres de la conséquence sont obtenus par l'algorithme des moindres carrés décrit dans les sections précédentes.

Le modèle avec le plus petit indice de performance est choisi en dehors des 5 modèles. Ce modèle est appelé état stable.

Le tableau 4.1, illustre une comparaison de l'indice de performance ainsi que la variance entre les 5 modèles.

	Indice de performance	variance
Modèle 1-1(PH)	78.926	10957
Modèle 1-2(TE)	75.94	10355
Modèle 1-3(AL)	87.038	13568
Modèle 1-4(TB1)	87.06	11711
Modèle 1-5(TB2)	90.71	13655

Tableau 4.1 Indice de performance par modèle.

Le modèle 1-2 dont TE seule apparaît dans la prémisse possède un indice de performance le plus faible (Cf. Tableau 4.1), il sera donc l'état stable.

La figure 4.4 montre la sortie du modèle par rapport à celle de l'expert.

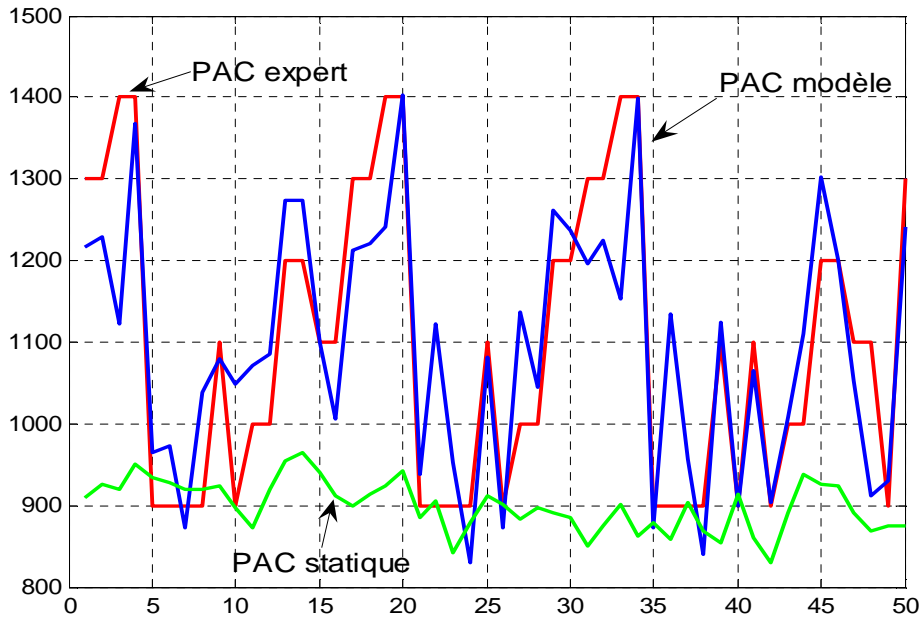


Figure 4.4 Sortie du modèle flou avec la variable (TE) dans la prémisse

L'apparition de la variable TE dans la prémisse indique une sortie donnant un indice de performance égal à 75.94, et une variance de 10355.

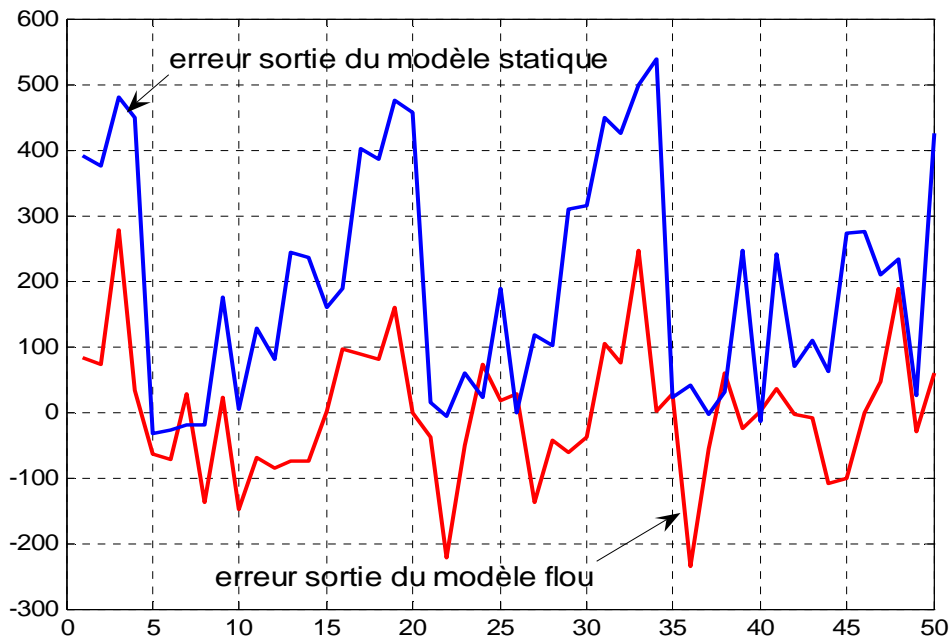


Figure 4.5 Comparaison des erreurs du modèle flou et celui du modèle statique avec la variable (TE) dans la prémisse

L'erreur de la sortie du modèle flou par rapport au modèle ARMA, ainsi que celle donnée par le modèle empirique est plus faible (CF. Figure 4.5), alors qu'il reste d'autres variables à introduire dans la partie prémisse. Justement dans l'étape 3, une introduction d'une deuxième variable aura lieu.

Etape 3 :

Toutes les combinaisons de (TE, TE) , (TE, PH) , (TE, AL) , $(TE, TB1)$ et $(TE, TB2)$ sont prises en compte, et le sous espace de chaque variable est divisé en deux sous espaces flous.

Pour la combinaison (TE, TE) , le sous espace de TE est divisé en quatre sous espaces flous, par exemple « grand », « moyen grand », « moyen petit » et « petit ».

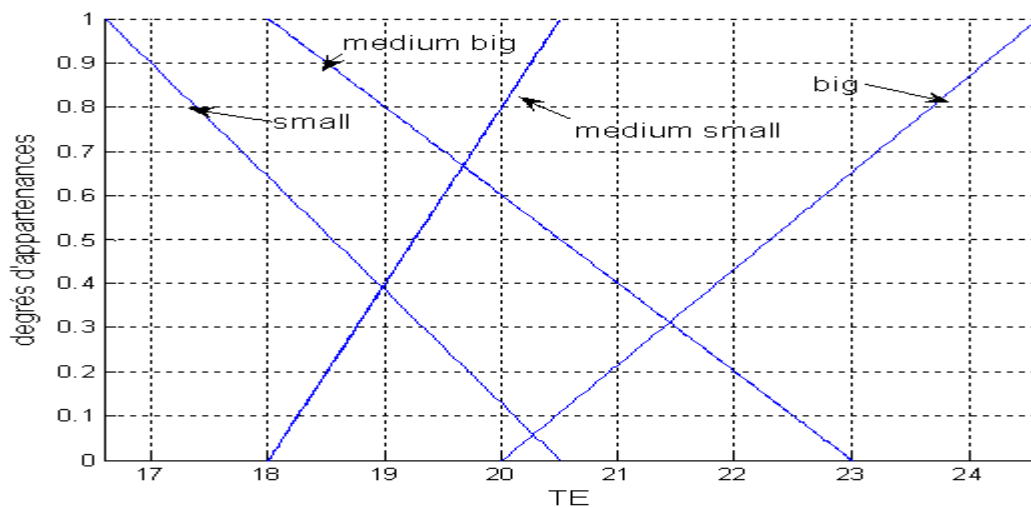


Figure 4.6 Fonction d'appartenance de la variable TE
divisée en 4 sous espaces flous

La figure (4.6), montre le partage du sous-espace de la variable en 4 sous-espaces flous, puisque cette variable est combinée avec elle-même.

Le modèle constitué aura alors quatre règles de production:

If TE is small then $PAC = a_{10} + a_{11} * PH + a_{12} * TE + a_{13} * AL + a_{14} * TB1 + a_{15} * TB2$

If TE is medium small then

$$PAC = a_{20} + a_{21} * PH + a_{22} * TE + a_{23} * AL + a_{24} * TB1 + a_{25} * TB2$$

If TE is medium big then

$$PAC = a_{30} + a_{31} * PH + a_{32} * TE + a_{33} * AL + a_{34} * TB1 + a_{35} * TB2$$

If TE is big then $PAC = a_{40} + a_{41} * PH + a_{42} * TE + a_{43} * AL + a_{44} * TB1 + a_{45} * TB2$

A la fin on aura 5 modèles de la forme du modèle 2-j, chacun est constitué de 2*2 règles de productions.

Le tableau 4.2, illustre l'indice de performance, la variance ainsi que l'erreur en moyenne pour les 5 modèles :

	Indice de performance	Variance
Modèle 2-1(TE-TE)	67.74	7811.6
Modèle 2-2(TE-PH)	41.86	2853.5
Modèle 2-3(TE-AL)	63.15	7839.1
Modèle 2-4(TE-TB1)	130.47	38664
Modèle 2-4(TE-TB2)	54.49	5850.2

Tableau 4.2 Indices de performance par modèle.

Le modèle 2-2 possède le plus petit indice de performance ce qui veut dire que les variables TE et PH apparaissent dans la prémisse et c'est l'état stable.

Chacune des variables est divisée en deux sous ensembles flous.

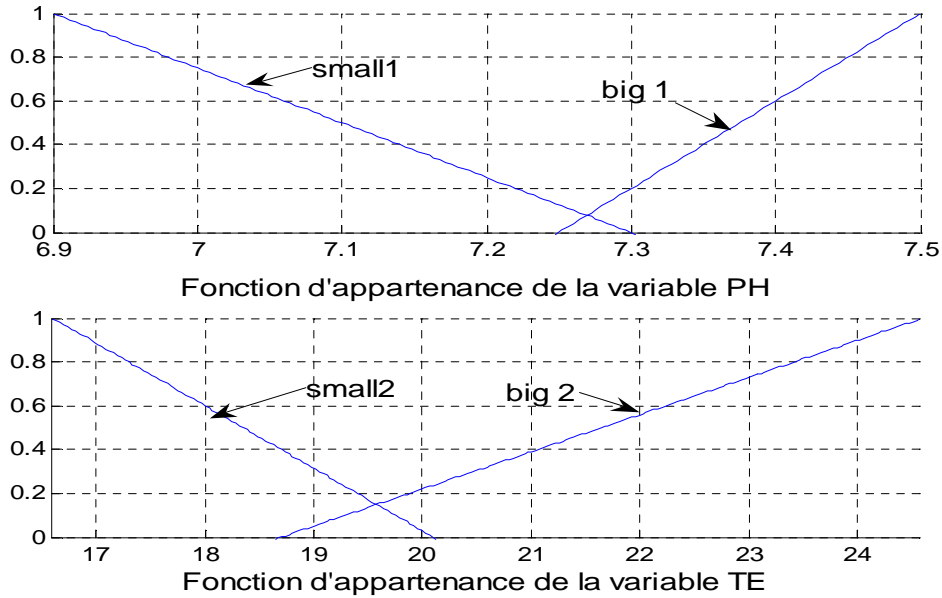


Figure 4.7 Fonctions d'appartenance des variables (TE, PH) dans l'antécédent.

Les règles de productions sont de la forme :

If PH is small1 and TE is small2 then

$$PAC = 1839.5 - 369.07 * PH + 104.56 * TE + 3.94 * AL + 3.38 * TB1 - 149.99 * TB2$$

If PH is small1 and TE is big2 then

$$PAC = 7647.9 - 1144.1 * PH + 104.05 * TE - 15.37 * AL + 10.1 * TB1 - 55.78 * TB2$$

If PH is big1 and TE is small2 then

$$PAC = 85893 - 12403 * PH + 204.21 * TE + 40.07 * AL + 5.85 * TB1 + 22.95 * TB2$$

If PH is big1 and TE is big2 then

$$PAC = -13867 + 2695.3 * PH - 28.01 * TE - 82.63 * AL + 9.01 * TB1 - 88.8 * TB2$$

La figure suivante montre la sortie de l'expert ainsi que celle issue du modèle.

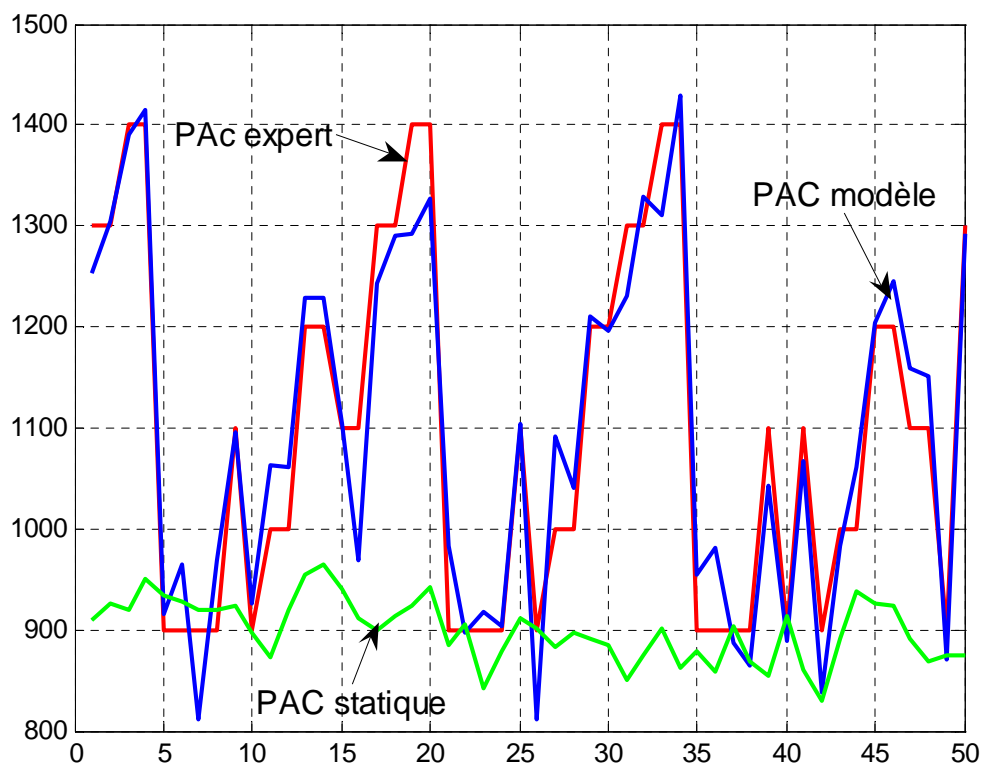


Figure 4.8 Sortie du modèle flou avec les variables (TE, PH) dans la prémisse donnant $I_p = 41.86$

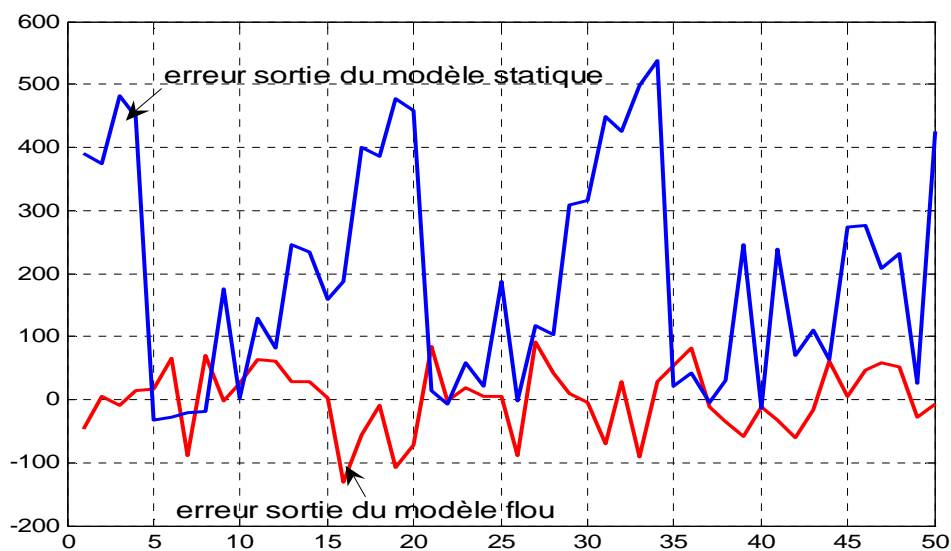


Figure 4.9 Comparaison des erreurs du modèle flou et celui du modèle statique avec les variables (TE, PH) dans la prémisse

L'erreur entre le PAC donné par l'expert et celui obtenu par le modèle flou (variance= 2853.5 ; erreur en moyenne = $1.92 \cdot 10^{-9}$) est plus centré et possède une variance plus faible que celle du modèle statique.

Nous devons maintenant passer à impliquer 3 variables dans la prémisse

On prend maintenant toutes les combinaisons

$(TE, PH, TE), (TE, PH, PH), (TE, PH, AL), (TE, PH, TB1)$ et $(TE, PH, TB2)$ et on divise le sous espace de chaque variable en deux sous espaces flous.

Pour (TE, PH, TE) : l'espace de TE est divisé en quatre sous-espaces flous et PH en deux, donc on aura 8 règles de production de la forme :

If TE is small and PH is small then

$$PAC = a_{10} + a_{11} * PH + a_{12} * TE + a_{13} * AL + a_{14} * TB1 + a_{15} * TB2$$

If TE is small and PH is big then

$$PAC = a_{20} + a_{21} * PH + a_{22} * TE + a_{23} * AL + a_{24} * TB1 + a_{25} * TB2$$

If TE is medium small and PH is small then

$$PAC = a_{30} + a_{31} * PH + a_{32} * TE + a_{33} * AL + a_{34} * TB1 + a_{35} * TB2$$

If TE is medium small and PH is big then

$$PAC = a_{40} + a_{41} * PH + a_{42} * TE + a_{43} * AL + a_{44} * TB1 + a_{45} * TB2$$

If TE is medium big and PH is small then

$$PAC = a_{50} + a_{51} * PH + a_{52} * TE + a_{53} * AL + a_{54} * TB1 + a_{55} * TB2$$

If TE is medium big and PH is big then

$$PAC = a_{60} + a_{61} * PH + a_{62} * TE + a_{63} * AL + a_{64} * TB1 + a_{65} * TB2$$

If TE is big and PH is small then

$$PAC = a_{70} + a_{71} * PH + a_{72} * TE + a_{73} * AL + a_{74} * TB1 + a_{75} * TB2$$

If TE is big and PH is big then

$$PAC = a_{80} + a_{81} * PH + a_{82} * TE + a_{83} * AL + a_{84} * TB1 + a_{85} * TB2$$

De même, pour la combinaison (TE, PH, PH) où l'espace de PH est divisé en quatre sous espaces flous et TE en deux.

Le tableau 4.3 illustre les différences quant à l'indice de performance.

	Indice de performance	Variance	Erreur en moyenne
Modèle 3-1(TE-PH-TE)	175.13	$1.09 \cdot 10^5$	101.79
Modèle 3-2(TE-PH-PH)	117.85	22485	28.06
Modèle 3-3(TE-PH-AL)	1.95	26.51	1.18
Modèle 3-4(TE-PH-TB1)	444.3	$1.05 \cdot 10^6$	397.7
Modèle 3-5(TE-PH-TB2)	472.25	$1.07 \cdot 10^6$	317.42

Tableau 4.3 Indice de performance par modèle

Comme ça était déjà cité dans les sections précédentes, qu'afin d'obtenir un indice de performance optimal il faut déterminer les fonctions d'appartenance optimales minimisant cet indice, et donc en changeant les sous-espaces des fonctions d'appartenances, et c'est ce qui a été fait avant d'obtenir l'indice de performance égal à 1.95, sachant qu'on avait d'abord 18.08 ensuite 14.3, etc.

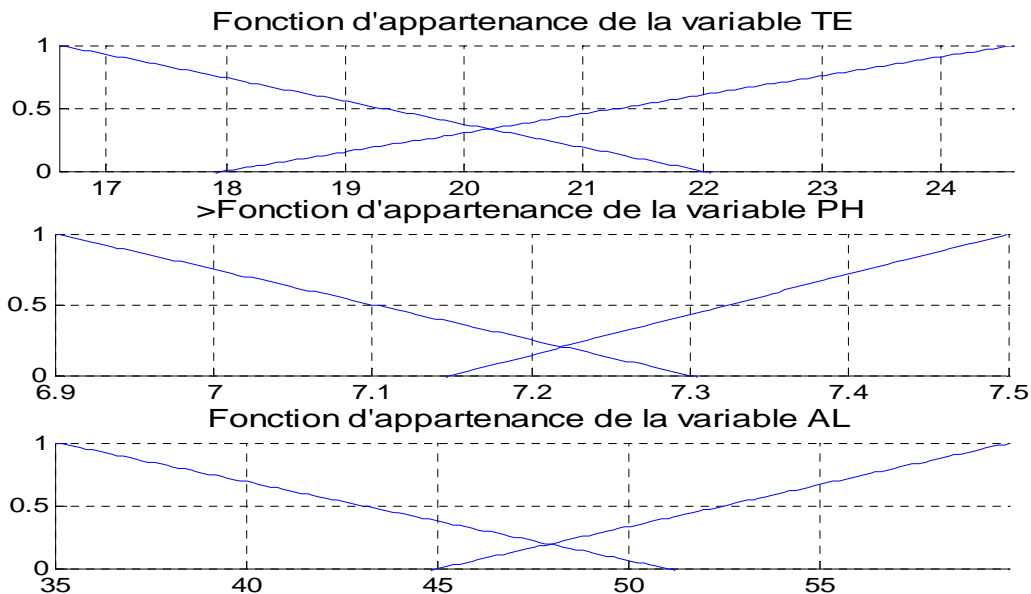


Figure 4.10 Fonction d'appartenance des variables TE, PH et AL

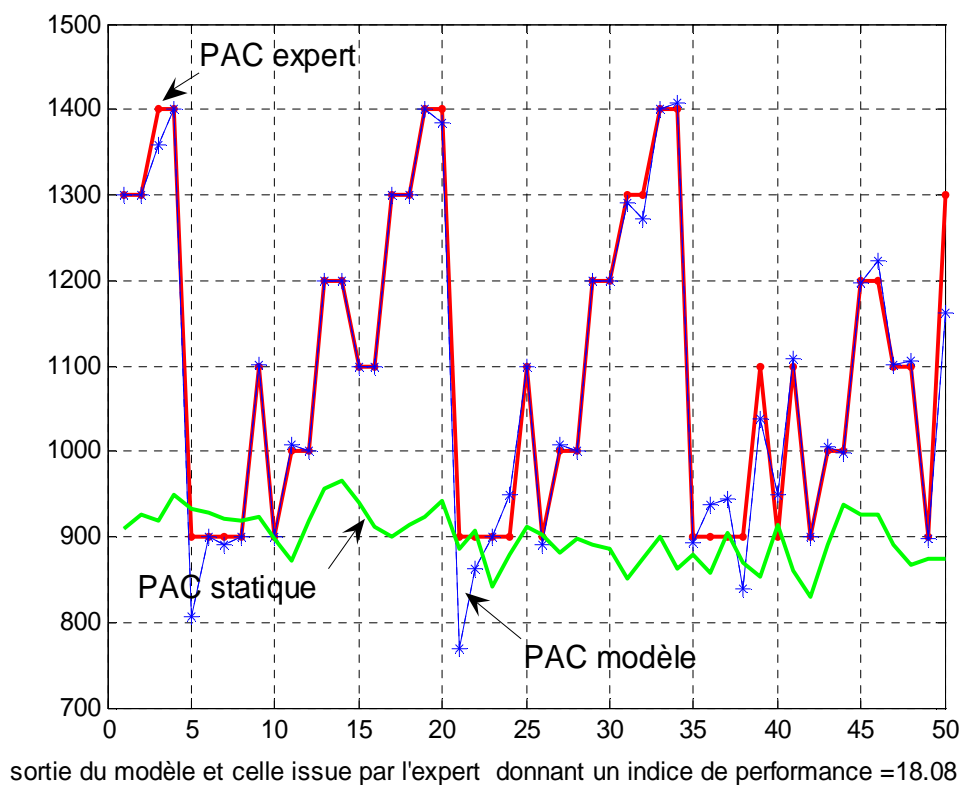


Figure 4.11 Sortie du modèle flou avec les variables (TE, PH, AL) dans la prémisse donnant $I_p=18.08$.

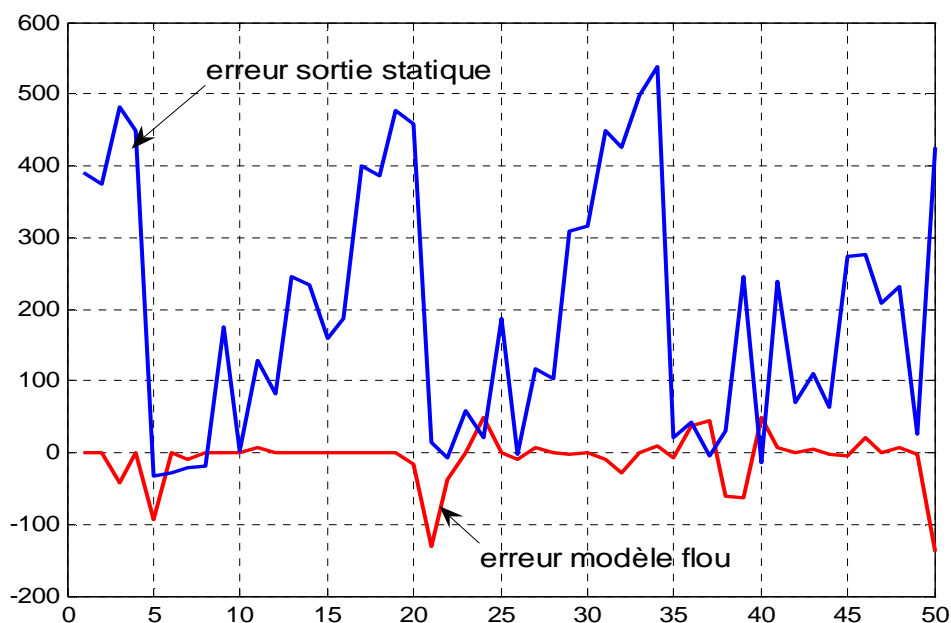


Figure 4.12 Comparaison des erreurs du modèle flou et celui du modèle statique avec les variables (TE, PH, AL) dans la prémisse avec $I_p=18.08$

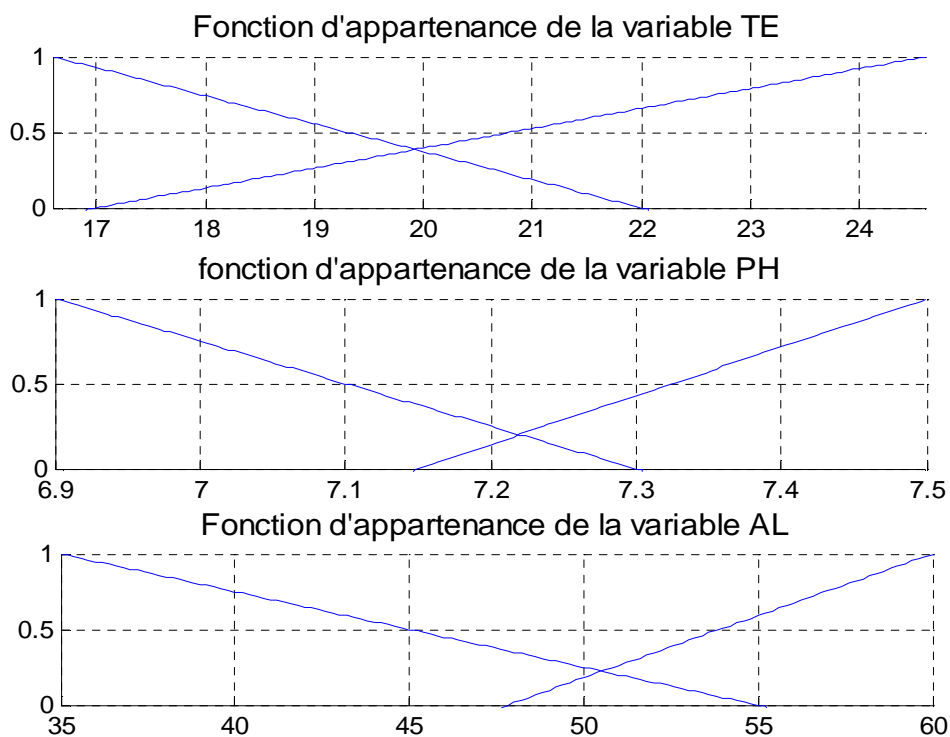
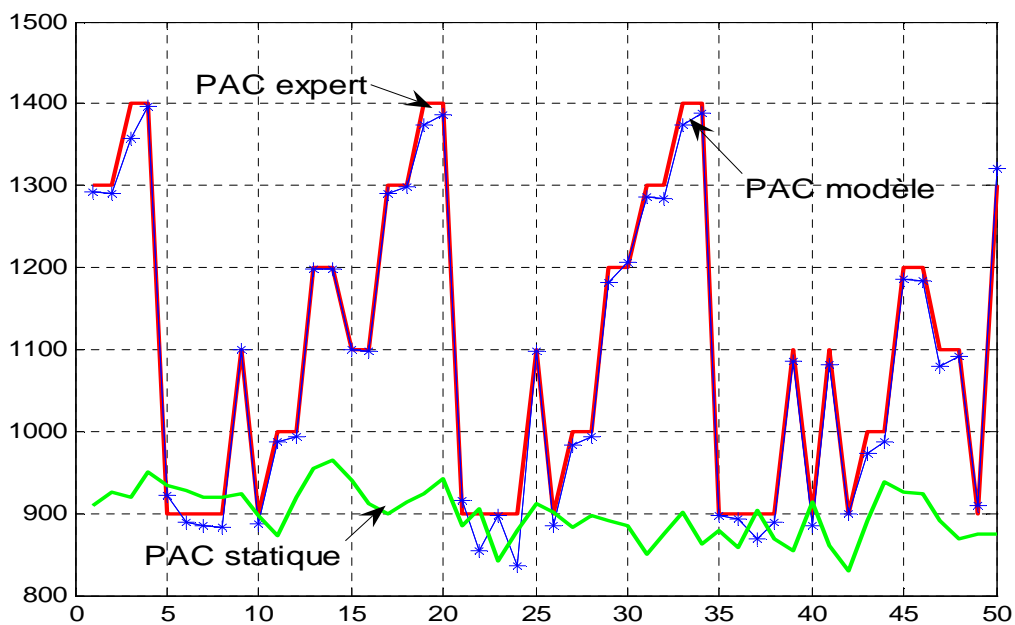


Figure 4.13 Fonctions d'appartenances

Figure 4.14 Sortie du modèle flou avec les variables (TE, PH, AL) dans la prémisse donnant $l_p=14.3$

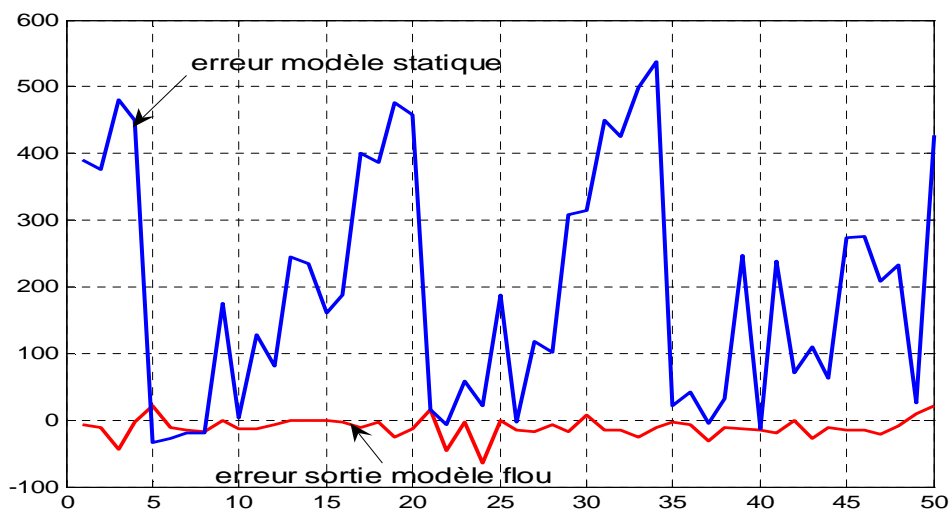


Figure 4.15 Comparaison des erreurs du modèle flou et celui du modèle statique avec les variables (TE, PH, AL) dans la prémisses avec $I_p=14.3$

Une autre partition de l'univers de discours est considérée (Cf. Figure 4.16), ce qui améliore encore plus l'indice de performance.

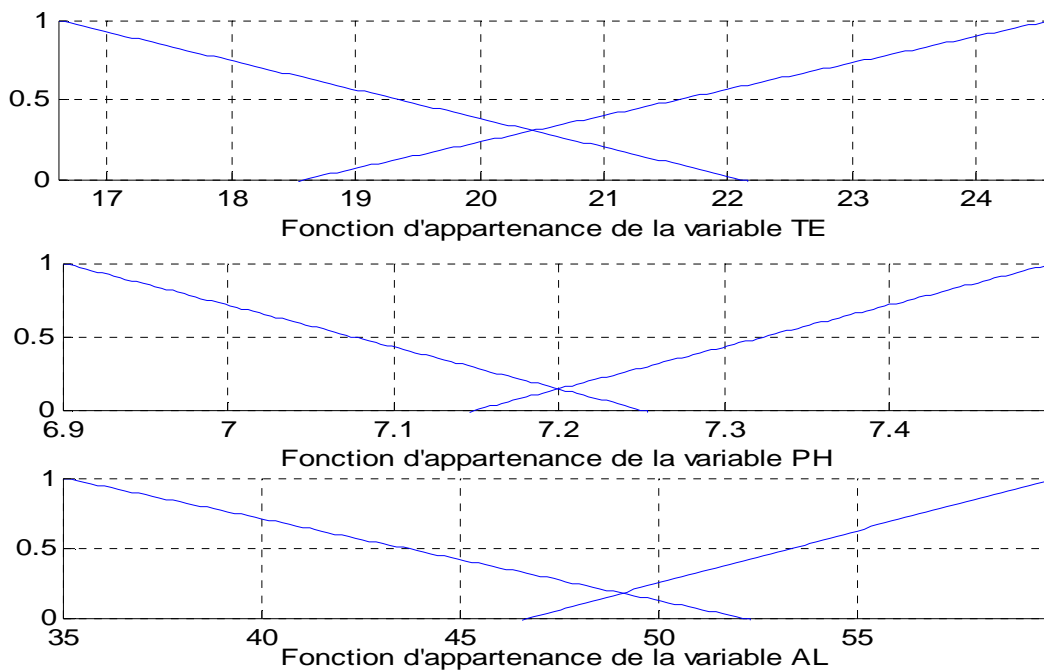


Figure 4.16 Fonctions d'appartenances

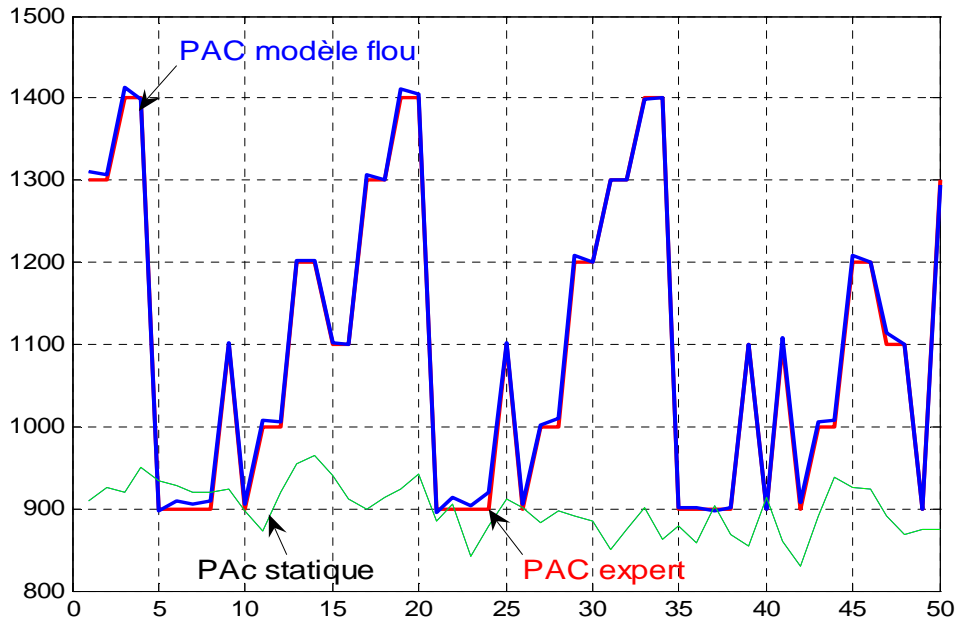


Figure 4.17 Sortie du modèle flou avec les variables (TE, PH, AL) dans la prémisse donnant $I_p=4.89$

La figure 4.17 montre la sortie correspondante à la partition de l'univers de discours (Cf. Figure 4.16) indiquant $I_p=4.89$.

Et de là on peut déduire que le modèle avec le plus petit indice de performance est le modèle 3-3, avec 8 règles de production :

If PH is small1 and TE is small2 and AL is small3 then

$$PAC = -3.81 + 4.43 * 10^5 * PH + 33354 * TE - 0.014 * AL - 1162.4 * TB1 + 1.16 * 10^5 * TB2$$

If PH is small1 and TE is small2 and AL is big3 then

$$PAC = -4.86 + 7.26 * 10^7 * PH + 1.17 * 10^6 * TE - 7.85 * AL - 3.1 * TB1 - 5.26 * TB2$$

If PH is small1 and TE is big2 and AL is small3 then

$$PAC = -2.12 + 3.38 * 10^6 * PH - 1.89 * TE + 7955.5 * AL + 47074 * TB1 + 1.98 * TB2$$

If PH is small1 and TE is big2 and AL is big3 then

$$PAC = 1.6 * 10^8 - 1.7 * PH - 2.77 * TE - 6.83 * AL + 82465 * TB1 + 2.32 * 10^6 * TB2$$

If PH is big1 and TE is small2 and AL is small3 then

$$PAC = -6.87 * 10^7 + 7.38 * 10^6 * PH + 1.54 * 10^5 * TE + 2.26 * 10^5 * AL - 26192 * TB1 + 7.5 * 10^5 * TB2$$

If PH is big1 and TE is small2 and AL is big3 then

$$PAC = -2.63 + 3.79 * 10^7 * PH - 1.21 * TE - 2.5 * AL + 83058 * TB1 + 96250 * TB2$$

If PH is big1 and TE is big2 and AL is small3 then

$$PAC = 2.11 * 10^6 - 1.03 * PH - 10042 * TE - 22600 * AL + 13.71 * TB1 + 1034 * TB2$$

If PH is big1 and TE is big2 and AL is big3 then

$$PAC = -3.03 + 2.91 * 10^5 * PH - 44359 * TE + 35196 * AL - 613.61 * TB1 - 4005.4 * 10^6 * TB2$$

4.6.2 Résultats de la modélisation floue

Les résultats présentés dans le tableau 4.4 montrent les performances de notre modèle flou.

Les résultats obtenus ($PAC_{\text{modèle_flou}}$) sont très proches de ceux obtenus par l'habile et expérimenté opérateur humain (PAC_{expert}) et sont nettement meilleurs que ceux obtenus à l'aide du modèle statique (PAC_{statique}), utilisé jusqu'à nos jours dans la majorité des stations qui est représenté par la formule empirique suivante :

$$PAC_{\text{statique}} = 9.11 \sqrt{TB1} - 79.8 PH + 12.7 CL + 1255.6$$

Le tableau ci-dessous, illustre les résultats obtenus.

PAC expert	PAC modèle flou	PAC statique			
1300	1300.9	909.09	900	898.7	901.79
1300	1300.6	918.99	1000	999.1	882.05
1400	1395	950.29	1000	1000.8	897.48
1400	1401.8	933.68	1200	1200.6	891.48
900	883.9	927.9	1200	1200.1	884.82
900	902.3	919.99	1300	1300	851.12
900	898.7	919.85	1300	1299.7	874.38
900	900.9	923.77	1400	1399.8	900.98
1100	1099.9	896.75	1400	1400.1	862.93
900	900.4	896.75	900	900.2	878.61
1000	1000	872.16	900	900.1	858.58

1000	1001.5	919.01	900	899.8	904.03
1200	1199.9	955.47	900	900	868.52
1200	1199.9	965.13	1100	1100	854.03
1100	1099.9	939.94	900	900.1	913.75
1100	1101.2	911.9	1100	1100	860.28
1300	1300.6	899.82	900	900.7	830.07
1300	1299.9	913.5	1000	999.8	890.25
1400	1404.1	923.77	1000	1001.1	937.9
1400	1399.3	942.69	1200	1199.8	926.62
900	877.4	884.89	1200	1199.7	924.7
900	895.3	906.36	1100	1101.1	891.2
900	900.2	841.93	1100	1100.1	868.05
900	899.2	878.17	900	899.5	874.12
1100	1099.9	911.86	1300	1277.3	873.89

Tableau 4.4 Illustration des résultats flous, statiques et mesurés.

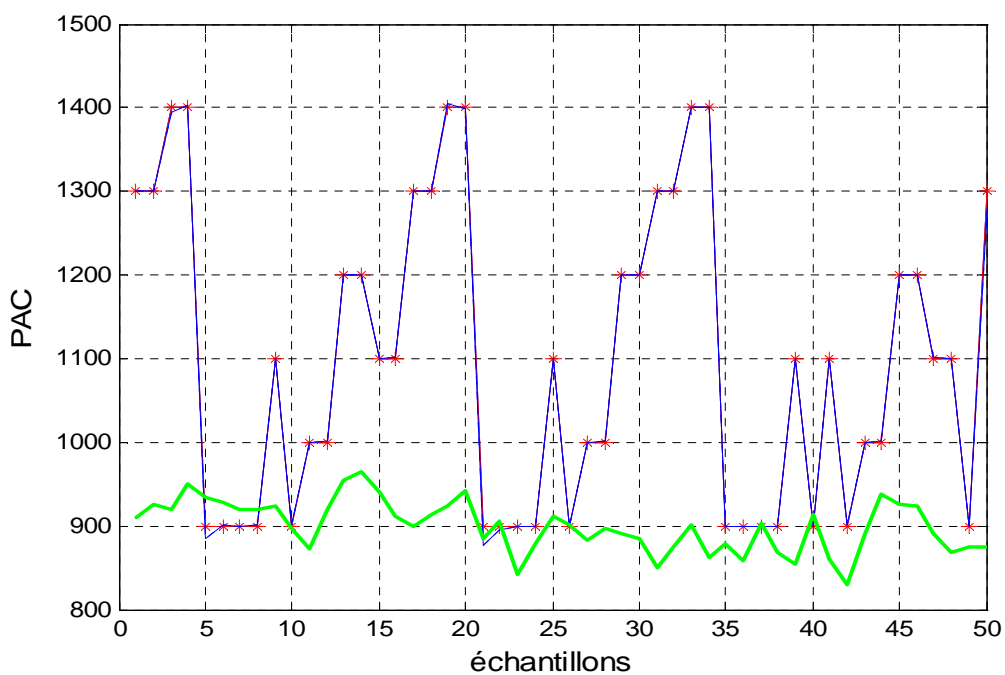


Figure 4.18 PAC statique, PAC modèle flou et PAC de l'expert.

La figure 4.18 illustre la sortie du modèle donnant un $I_p=1.95$, ce modèle permet une automatisation du procédé d'adjonction du produit PAC dans le processus d'épuration (sans la présence de l'expert).

Alors que les valeurs obtenues par la formule empirique ne sont pas performantes; le calcul de l'indice de performance donne une valeur de 201.22.

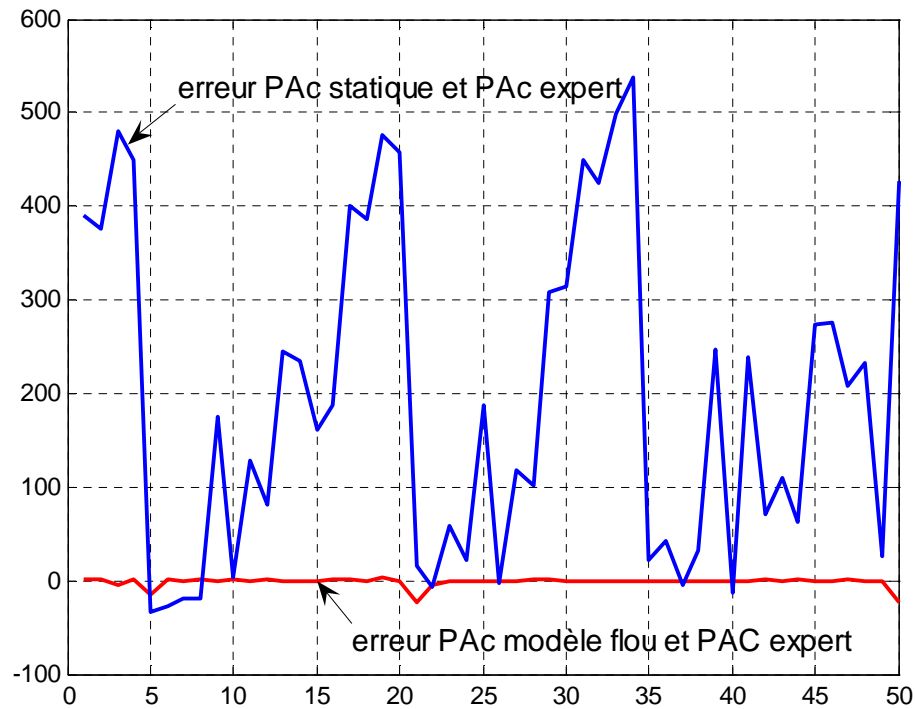


Figure 4.19 Erreur PAC expert/ PAC modèle flou et celle PAC expert/PAC statique.

On voit bien en figure 4.19 que l'erreur moyenne entre PAC expert et PAC modèle flou est très faible devant celle qui existe entre le PAC expert et le PAC statique (environ 196.12)

4.7 Résultats et discussions :

Le modèle flou a bien imité les actions de l'expert, une nette amélioration de l'indice de performance a été obtenue relativement à l'utilisation du modèle neuronal et ARMA.

L'architecture du contrôle de la station d'épuration de l'eau peut-être représentée selon le schéma-bloc suivant, dans lequel l'expert est représenté par un modèle neuronal tan disque le processus d'épuration de l'eau est schématisé par un modèle flou de type Takagi-Sugeno.

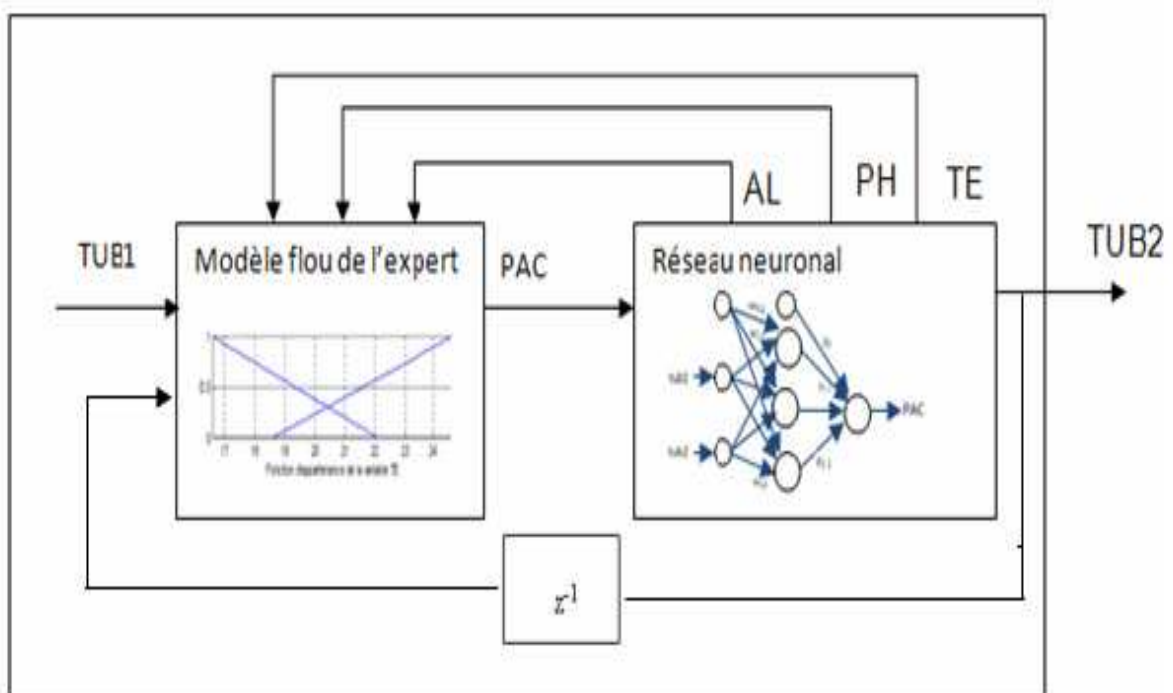


Figure 4.20 Architecture du processus d'épuration de l'eau

Le modèle flou du processus « opérateur » délivrera la quantité du PAC au modèle neuronal du processus d'épuration de l'eau.

Celui-ci nécessite l'acquisition des paramètres AL, PH et TE. De plus, comme le modèle flou possède comme deuxième entrée, la variable TUB2, on utilisera celle obtenue lors du traitement précédent.

Cette architecture a permis d'obtenir le résultat suivant, le plus performant de tous :

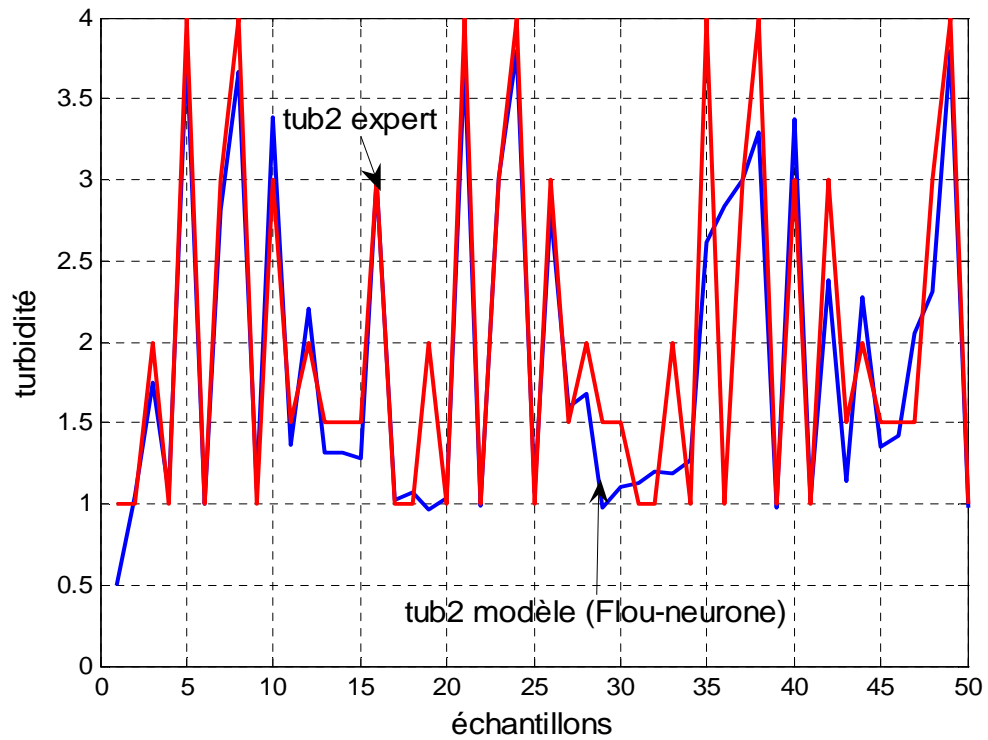


Figure 4.21 Modèle neuro-flou

Cette modélisation a permis d'obtenir un $I_p = 0.29$ (meilleur indice de performance), et une variance de 0.2.

4.8 Conclusion

Dans un processus d'épuration de l'eau de surface, la décision sur la quantité de PAC à adjoindre à l'eau non traitée est sans doute, la décision la plus importante.

Cette quantité dépend de plusieurs facteurs dont certains ne sont même pas mesurables (température, pH, Alcalinité, quantité de plancton, etc.).

Il est très difficile d'établir une relation mathématique qui permet de déterminer d'une manière systématique la quantité du PAC nécessaire.

Un modèle statique expérimental, représenté par une formule empirique est utilisé dans les stations de traitement, mais les résultats obtenus sont loin de ceux obtenus en présence d'un habile et expérimenté opérateur humain qui prendrait les décisions plus adéquates sur la quantité de PAC à ajouter.

Dans cette contribution, on a élaboré un modèle utilisant les techniques floues type Takagi-Sugeno, en essayant de reproduire les décisions de l'opérateur humain sur la quantité de PAC dans le processus d'épuration de l'eau. Des données (relevés) réels sur une station d'épuration, sur une durée de 50 jours ont été utilisées.

Le modèle obtenu s'est avéré très efficace (par rapport au modèle neuronal) et permet une automatisation du procédé d'adjonction du produit PAC dans le processus d'épuration (sans la présence de l'expert). Ce modèle flou pourrait être une excellente alternative en remplacement du modèle empirique statique utilisé jusqu'à nos jours dans nos stations et dont les résultats sont très insatisfaisants.

CONCLUSION

Les travaux présentés dans ce mémoire s'articulent autour des thèmes principaux de la modélisation neuronale et de la commande floue de type Takagi-Sugeno (TS) pour une application à une station d'épuration des eaux de surface.

Dans la première partie, il est supposé que le système (station d'épuration) possède un comportement linéaire. L'équation aux récurrences des entrées-sorties, a permis l'utilisation de la modélisation statique, pour limiter la turbidité à un certain seuil de tolérance et déterminer ainsi, la quantité du réactif PAC à introduire.

Dans la deuxième partie une modélisation neuronale, a aussi permis d'améliorer les résultats obtenus par le premier modèle (ARMA) proposé. Plusieurs architectures, ont été élaborées avec un certain nombre de neurones à l'entrée des réseaux, inférieur au nombre de neurones dans la couche cachée et un neurone, dans la couche de sortie représentant respectivement, le PAC et la turbidité par rapport à chaque réseau. La mise à jour des poids et des biais, est réalisée par l'algorithme d'optimisation «rétropropagation du gradient», utilisant un apprentissage supervisé (à chaque entrée correspond une sortie connue).

La station a été modélisée par deux réseaux de neurones montés en cascade, le premier décrivant les actions de l'expert, le deuxième modélisant le processus d'épuration de l'eau, pour la détermination de la quantité de coagulant (PAC) à ajouter à l'eau non traitée. Les résultats obtenus ayant montré qu'une amélioration du rendement du premier réseau était nécessaire, on s'est proposé d'utiliser la modélisation floue comme technique de simulation des actions de contrôle de la quantité de PAC par l'opérateur humain.

Dans la troisième partie, nous avons présenté d'abord une approche structurale de la modélisation et de l'identification floues en nous focalisant particulièrement sur le modèle de type Takagi-Sugeno (TS), à partir des données

entrées-sorties. En effet, au cours de ces dernières années la discipline a évolué d'une façon graduelle, vers une utilisation pratiquement exclusive des systèmes flous dans lesquels le conséquent des règles utilise des variables numériques sous la forme de fonctions (modèle de type TS) plutôt que des variables linguistiques (modèle de type Mamdani).

Dans le cas de l'identification floue des systèmes, le formalisme Takagi-Sugeno est mieux adapté à une démarche plus systématique pour la construction de modèles non linéaires multivariables, grâce à leur bonne capacité d'interpolation numérique et d'apprentissage à partir de données.

La construction de ce type de modèle flou, est divisée en deux parties :

- Une identification de la structure du modèle flou, a été réalisée par la détermination des variables linguistiques qui apparaissent dans l'antécédent avec la partition optimale de l'univers de discours.
- Une identification, des paramètres de conséquence des règles de production suivant le modèle employé, pour déterminer la sortie inférée.

L'indice de performance a été très largement réduit par rapport aux deux approches précédentes (ARMA et neuronale). En outre les résultats obtenus sont meilleurs que les résultats obtenus par utilisation de la formule empirique utilisée jusqu'à nos jours dans les stations d'épuration des eaux dans notre pays. D'où l'intérêt de ce travail.

En perspective nous proposons :

- de récupérer l'historique des mesures d'entrées-sorties sur le comportement de la dynamique, et de les stocker sous forme d'entrepôt (data warehouse). Dans ce contexte, il sera utilisé une modélisation par chaîne de markov ;
- d'Identifier les paramètres du modèle flou, par l'algorithme de colonies de fourmis ;
- d'implémenter le modèle étudié, dans la salle de contrôle pour automatiser le système.

REFERENCES

1. Hadj sadok, Z.K., « Modélisation et estimation dans les bioréacteurs ; prise en compte des incertitudes : application au traitement de l'eau » thèse de Doctorat de l'Université de Nice – Sophia Antipolis, France, 1999.
2. Pierre. M., « Traitement des eaux avant utilisation » technique de l'ingénieur, G 1 172,p 1-12,1993.
3. Sibony. J.,Bigot. B., « Traitement des eaux urbaines » technique de l'ingénieur, C 5220, p 2-28,1993
4. Salghi. R., « Différents Filières de traitement des eaux », sciences Appliquées d'Agadir, ENSA, Agadir, 2008.
5. Gilles. O., « L'eau » école Industrielle et commerciale, Namur, 2008.
6. Bassom. C., « Procédé a boues actives pour le traitement d'effluents papetiers » ; thèse de Doctorat, Automatique, Grenoble, 2007.
7. Couture. I, « Principaux critères pour évaluer la quantité de l'eau en micro-irrigation », colloque Saint-Hyacinthe, Québec, 2006
8. Desjardins. R, « Le traitement des eaux », Editions de l'Ecole Polytechnique de Montréal, 1997.
9. Mouchet. P., « Traitement des eaux avant utilisation », « Matières particulières », technique de l'ingénieur, Strasbourg, G1 170, p 4-18,1993.
10. Abdelazim.T., Malik.O.P., « Identification of nonlinear systems by Takagi-Sugeno logic grey box modeling for real-time control » science direct control engineering practice,13 (2005) 1489-1498.

11. Ben Abdennour. R., Borne. P., Ksouri. M., M'sahli.F, « Identification et commande numérique des procédés industriels », édition technip, avril 2001.
12. Maddi. A., Guessoum. A., Berkani.D., Belkine.O « Etude de la méthode des moindres carrés récursive et application au signal de parole », 3rd International conference sciences of electronic, Technologies of Information and telecommunications, SETIT, 2005.
13. Mokhtari.M., Mesbah.A., « Apprendre et maîtriser Matlab », 540-62773, Allemagne, p 667-669, 1997.
14. Period. P., Pivonka, V., Veleba, P., Osmera., « Adaptive Controllers by Using Neural Network Based Identification for Short Sampling » International Journal of Circuits, Systems and Signal Processing, vol. Vol. 1, No. 1., pp. 62-67, 2008.
15. Kara. K., Hadjili. M.L., Benmohamed. K., « Modélisation neuronale des systèmes non-linéaire en présence de perturbation », setif, pp 1,6, 2008.
16. St-Hilaire. A., Ouarda. T.B « Utilisation des réseaux de neurones et de la régularisation bayésienne en modélisation de la température de l'eau en rivière », Revue des sciences de l'eau, Journal of Water Science, vol. 21, n° 3, 2008, p. 373-382.
17. Hebb.D.O., «The organisation of behavior », John Wiley and sons, New-york, 1949.
18. Rumelhart, D.E., McClelland.J.L., « Parallel Distributed Processing: Exploration in the MicroStructure of Cognition » MIT Press, Cambridge, 1986.
19. Marie. M., Mokhtari. M., « Application de Matlab5 et Simulink », 2-287-59651-8, verlag France, p 95-139, p 149-152, 1998
20. Hadj sadok. M., Guessoum.A., Salhi, H., Tidjani. N, « Modélisation neuronale des actions de contrôle de la quantité de PAC dans un processus d'épuration des eaux » 4^{ème} journée sur le traitement et la réutilisation des eaux, USDB, jstre 2010.
21. Belaidi.I., Tahmi.R., Mohammedi. K., « Configuration optimale d'un réseau de neurones adapté à la surveillance en ligne de l'usure des outils de tournage » Grenoble, 2007.
22. Hornik. K., « Approximation capabilities of multilayer feedforward networks », vol.4, pp 251-257, 1991.

23. M. Hadj sadok « Méthodologie de contrôle basée sur la logique floue »
Thèse de Magister, USDB, 1995.
24. Babuska.R., « Fuzzy Modeling for Control ». Kluwer Academic Publishers,
USA, 1998.
25. ZADEH. L., « Outline of a new approach to the analysis of complex systems
and decision processes », IEEE Trans. on Systems, Man, and Cybernetics,
vol.1, pp.28-44, 1973.
26. Mamdani. E., « Application of fuzzy logic to approximate reasoning using
linguistic systems », Fuzzy Sets and Systems, vol.26, pp.1182-1191, 1977.
27. Grisales. P ., « Modélisation et commande floues de type Takagi-Sugeno
appliquées à un bioprocédé de traitement des eaux » thèse de doctorat,
Université Paul Sabatier, 2007.
28. Takagi. T., Sugeno. M., « Fuzzy identification of systems and its application to
modelling and control », IEEE Trans. On systems, Man and Cybernetics, 15, p.
116-132, 1985.
29. Foulloy. L., Galichet. S « La commande floue – De l'approximation à
l'apprentissage » Hermès IC2, 2003.
30. Sugeno, M., Kang, G.T., « Structure identification of fuzzy model » Fuzzy sets
and systems, vol.28, pp.15-33, 1988
31. Hadj sadok.M., Guessoum.A., Salhi.H., Tidjani. N., « modélisation floue des
actions de contrôle de la quantité de PAC dans un processus d'épuration des
eaux », 3^{ème} journée sur le traitement et la réutilisation des eaux, USDB, jstre
2009.
32. Boca. R., « introduction to fuzzy sets, fuzzy logic, and fuzzy control systems »,
New York, 2001.