MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE FERHAT ABBAS - SETIF U.F.A.S. (ALGERIE)

MEMOIRE

Présenté à la Faculté des Sciences de l'Ingénieur Département d'électronique

Pour l'Obtention du Diplôme de

MAGISTERE

Option: CONTRÔLE

Par

OUAREM MOHAMED

THEME

Implémentation en temps réel d'une technique d'analyse spectrale paramétrique à l'aide de l'approche neuro-floue optimisée par les algorithmes génétiques

·		
Président : N. BOUZIT ;	PR	UFAS
Examinateur : L. BERRETIL ;	MC	UFAS
Examinateur : K. BENMAHAMMED ;	PR	UFAS
Rapporteur : F. KRIM ;	PR	UFAS

Soutenu le:....., devant la commission d'examen:

Implémentation en temps réel d'une technique d'analyse spectrale paramétrique à l'aide de l'approche neuro-floue optimisée par les algorithmes génétiques

OUAREM Mohamed

Mémoire présenté à la Faculté des Sciences de l'Ingénieur Département d'Electronique pour l'obtention du diplôme de Magistère option Contrôle.

ABSTRACT

La surveillance en temps réel d'harmoniques est devenue un souci important dans beaucoup d'applications, comme dans les systèmes de puissance qui sont soumis à une pollution harmonique due aux charges non linéaires. La limitation en temps et la complexité de calcul informatique liés aux techniques conventionnelles incitent l'utilisation de techniques alternatives pour l'extraction harmonique. D'autre part, la majorité des applications, telles que la conception des filtres nécessitent l'extraction rapide des différents harmoniques. Ainsi ce travail concerne une technique basée sur les réseaux neurones flous optimisée par les algorithmes génétiques pour l'extraction rapide d'harmoniques. Celle-ci utilise un type de fusion entre les réseaux de neurones artificiels et la commande par la logique floue dont les fonctions d'appartenance sont ajustées convenablement, grâce à la capacité d'apprentissage des réseaux de neurones, pour l'estimation, en temps réel, les différents harmoniques d'un signal distordu ou bruité. La technique proposée est réalisée et simulée avec le logiciel de simulation. Les résultats montrent que la technique proposée est nettement plus rapide et précise que les techniques conventionnelles.

Mots clés: les réseaux neurones, les algorithmes génétiques, la logique floue, les réseaux neurones flous, les recuits simulés.

REMERCIEMENTS

Tout d'abord, tous les éloges et remerciements sont à **DIEU**.

Je voudrais en premier lieu exprimer tous mes remerciements à monsieur le Professeur Fateh KRIM, pour m'avoir donné l'opportunité d'effectuer ce travail dans un domaine qui m'a toujours passionné, les algorithmes génétiques, la logique floue et les réseaux de neurones.

Mes remerciements vont également à tous ceux qui ont, de près ou de loin, participé à l'élaboration de ce document, et à tous ceux qui par un simple mot ou geste, m'ont aidé et encouragé tout au long de mes études. Je pense à mes parents, mes sœurs, mon frère, mes cousins, et à tous mes amis.

Tous mes remerciements vont également aux membres du jury pour m'avoir fait l'honneur d'examiner ce travail avec beaucoup d'intérêt.

ABREVIATIONS

RNA : Réseau de Neurones Artificiel.

RNF: Réseau de Neurones Flou.

AG: Algorithme Génétique.

RS: Recuit Simulé.

RNFOAG : Réseau de Neurones Flou Optimisé par les Algorithme Génétiques.

LF: Logique Floue.

in : input (entrée).

TABLE DES MATIERES

REMERCIEMENTS	
ABREVIATIONS	
LISTE DES FIGURES	
LISTE DES TABLES	
LISTE DES TABLES	
INTRODUCTION GENERALE	
CHAPITRE I	
NOTIONS FONDAMENTALES DANS LE TRAITEMENT D	
LES METHODES D'IDENTIFICATIONS	
1. Introduction	
2. Analyse de Fourier	
2.1. Les séries de Fourier	
2.1.3. Spectre de Fourier d'une fonction	
2.1.5. Exemples de décomposition en série de Fourie	
2.1.5. Exemples de décomposition en serie de Fourier 2.2. Les transformées de Fourier	
2.3. La transformée de Fourier à temps discret (TFTD	
2.3.1. Propriétés de la TFTD	
2.4. Transformée de Fourier Discrète (TFD)	
2.4.1. Propriétés de la TFD	
2.5. Transformée de Fourier Rapide (TFR, FFT)	
2.5.1. Algorithme de Cooley-Tuckey	
La Transformée de Laplace	
3.1. Définition	
3.2. Propriétés de la Transformée de Laplace	
4. Transformée en Z	و 11
4.1. Définition de la transformée en Z	
4.2. Propriétés de la transformée en Z	
4.3. Tableau de quelques transformées en Z	
5. Les méthodes d'identification	
5.1. Approche d'identification	
5.2. Les types d'identification	
5.2.1. Approche en boucle ouverte	
5.2.2. Approche en boucle fermée	
5.3. Type de modèles utilisés pour l'identification	
5.3.1. Les modèles non paramétriques	
5.3.2. Les modèles paramétriques	
6. Conclusion	
CHAPITRE II	
LES ALGORITHMES GENETIQUES, LA LOGIQUE FLOUI	E ET LES
RESEAUX DE NEURONES	
1. Les algorithmes génétiques	20
1.1. Introduction	
1.2. Principe de base	
1.3. Opérateurs génétiques	
1.3.1. Codage d'un individu	
1.3.2. Génération de la population initiale	
1.3.3. Evaluation	21

1.3.4.	Sélection	21
1.3.5.	Croisement	24
1.3.6.	Mutation	24
1.3.7.	Recuit Simulé	25
2. La logio	ue floue	
	roduction	
2.2. No	tions fondamentales de la logique floue	26
2.2.1.	Ensembles flous	
2.2.2.	Opérations sur les ensembles flous	
2.2.3.	Relations floues	
2.2.4.	Composition Sup-Star	
2.3. Re	présentation et raisonnement linguistique	
2.3.1.	Variables linguistiques	28
2.3.2.	Proposition floue	
2.3.3.	Conjonction et disjonction des propositions floues .	
2.3.4.	Raisonnement flou	
2.3.5.	Mécanisme d'inférence	
2.4. Co	ntrôle par la logique floue	
2.4.1.	Fuzzification	
2.4.2.	Règles et mécanisme d'inférence	
2.4.3.	Défuzzification	
	eaux de neurones artificiels (RNAs)	
	roduction	
	s neurones biologiques	
	ructure des neurones	
3.3.1.	Le corps cellulaire	
3.3.2.	Les dendrites	
3.3.3.	L'axone	
3.4. Ne	urone formel	
	tilisation des RNA	
3.5.1.	Classification	
3.5.2.	Catégorisation	
3.5.3.	Approximation de fonctions	
3.5.4.	Prédiction	
3.5.5.	Optimisation	
3.5.6.	Mémoire associative	
3.5.7.	Contrôle	
3.6. Ap	prentissage	36
3.6.1.	L'apprentissage supervisé	36
3.6.2.	L'apprentissage nonsupervisé	
3.6.3.	L'apprentissage hybride	
3.7. Qu	elques topologies des RNA	
	Le réseau de Hopfield	
3.7.2.	Le réseau de neurones de Hamming	
3.7.3.	Le Classificateur de Carpenter et Grossberg	
3.7.4.	Le perceptron multicouche	
3.7.5.	L'algorithme de rétropropagation	
3.7.6.	Carte auto-organisatrice de Kohonen	
4. Conclus	ion	
	III	
ANALYSE S	PECTRALE PAR LES AG, LES RNA ET LES RESEA	UX DE
	FLOUES (RNF)	

1.	Intro	oduction	41
2.	L'ide	ntification des paramètres des harmoniques par les AG	41
2.	1.	Initialisation de l'algorithme	42
2.	2.	Codage des chromosomes	43
2.	3.	Génération de la population initiale	43
2.	4.	L'évaluation	
2.	5.	Sélection	43
2.	6.	Croisement	44
2.	7.	Mutation	44
2.	8.	Evolution génétique	44
2.	9.	Meilleurs individus	44
2.	10.	Evolution d'extraction	44
3.	Iden	tification des paramètres des harmoniques par les recuits	
sim	ulés (RS)	45
3.	1.	Génération de voisinage	46
3.	2.	Diminution de la température	
3.	3.	L'intensification	46
3.	4.	Sélection du meilleur individu	46
4.	Extra	action on-line des harmoniques utilisant les RNA	47
4.	1.	Formulation du problème	47
4.	2.	Architecture du RNA	
4.	3.	L'apprentissage	48
4.	4.	Normalisation des entrées/sorties	
4.	5.	La sortie	49
4.	6.	Optimisation par les AG	
5.	L'ext	traction on-line des harmoniques utilisant les RNF	
	1.	Architecture du RNF utilisé	
5.	2.	L'apprentissage	52
5.	3.	Normalisation des entrées/sorties	
5.	4.	La sortie	
5.	5.	Optimisation par les AG	
6.	Cond	clusion	
CHA	APITE	RE IV	53
TES	T, SI	MULATION ET RESULTATS	53
1.		oduction	53
2.		al test/simulation et critères de performance	
3.		nation des paramètres des harmoniques par un AGAG	
	1.	Caractéristiques de l'AG utilisé	
4.	Vale	urs réelles, calculées par l'AG, par FFT et par RS	58
	1.	Propriétés du RS utilisé	
5.	Extra	action des harmoniques on-line par les RNA	
	1.	Propriétés du réseau utilisé	
5.	2.	Modèle de simulation SIMULINK	61
6.	Extra	action des harmoniques on-line par les RNFOAG	62
	1.	Propriétés du RNFOAG utilisé	
6.	2.	Modèle de simulation SIMULINK	
7.	Com	paraison entre RNA et RNFOAG	65
		luence du nombre d'entrées	
		luence de la fréquence	
		luence de l'amplitude	
		rprétation et discussion	
		clusion	

CONCLUSION GENERALE	72
PERSPECTIVE	73
BIBLIOGRAPHIE	74

LISTE DES FIGURES

Figure I:1: TFD d'ordre N à partir de 2 TFD d'ordre N/2	8
Figure I:2: Représentation schématique de la TRF	8
Figure I:3: Procédure d'identification.	. 13
Figure I:4: Principe de l'analyse par corrélation.	
Figure I:5: Réponse d'un procédé continu et discrétisé	. 17
Figure I:6: Réponse indicielle d'un système du deuxième ordre	
Figure I:7: Système en boucle fermée.	
Figure I:8: Principe de l'estimation des paramètres d'un modèle	
Figure II:1: Codage des individus.	
Figure II:2: Roulette proportionnel.	
Figure II:3: 2 Exemples de croissement.	
Figure II:4: Exemple d'opérateur de mutation.	
Figure II:5: Exemple de fonctions d'appartenance.	
Figure II:6: Schéma block d'un contrôleur floue.	
Figure II:7: Exemple de fuzzification.	
Figure II:8: Defuzzification par centre de gravité.	
Figure II:9: Le modèle d'un neurone biologique	
Figure II:10: Le modèle de McCulloch-Pitts.	
Figure II:11: Quelques types de RNAs.	
Figure II:12: Perceptron multicouche.	
Figure III:1: L'algorithme d'extraction par les AG.	
Figure III:2: L'algorithme d'extraction par les RS.	
Figure III:3: L'architecture du réseau de neurones utilisé	
Figure III:4: La fonction de transfert des neurones (logsig)	
Figure III : 5 : L'architecture résumée du Neuro-like Fuzzy.	
Figure III :6 : L'architecture du neurone floue utilisé	
Figure VI:1: Signal s(t) - fenêtre périodique (3 périodes)	
Figure VI:2: Spectre par FFT.	
Figure VI:3: Spectre par les AG.	
Figure VI:4: Les trois spectres.	
Figure VI:5: Signal s(t) - fenêtre non périodique (3.5 périodes)	
Figure VI:6: Spectre par FFT.	
Figure VI:7: Spectre par les AG.	
Figure VI:8: Les trois spectres.	
Figure VI:9: Quatre spectres : fenêtre périodique	
Figure VI:10: Quatre spectres : fenêtre non périodique	
Figure VI:11: Signal s(t)	
Figure VI:12: Réelles et extraies des 5 harmoniques par RNA.	
Figure VI:12: Neelles et extrales des 5 harmoniques par KNA	
Figure VI:14: Model de simulation pour 5 harmoniques	
Figure VI:15: Réelles et extraies des 5 harmoniques par RNF	
Figure VI:16: Model de simulation pour un seul harmonique	
Figure VI:17: Model de simulation pour 5 harmoniques	
Figure VI:18: Un quart de période du fondamental : réel, extrait par F	
(31 in) et par RNFOAG (6 in)Figure VI:19: Le RMSE en pourcentage pour un RNA de 64 entrées	. UO
Figure VI:19: Le RMSE en pourcentage pour un RNFOAG de 8 entrées	
· · · · · · · · · · · · · · · · · · ·	
Figure VI:22: Le RMSE en pourcentage pour un RNA de 64 entrées	
Figure VI:23: Le RMSE en pourcentage pour un RNFOAG de 8 entrées.	. / U

LISTE DES TABLES

Table I:1: Quelques transformées en Z	12
Table II:1: Les implications les plus utilisées.	29
Table II:2: Exemple d'une matrice d'inférence	32
Table II:3: L'analogie entre les neurones biologiques et les neuro	nes
artificiels	35
Table VI:1: Les différentes valeurs (fenêtre périodique)	54
Table VI:2: L'EAR et l'EFR (fenêtre périodique)	54
Table VI:3: Les différentes valeurs (fenêtre non périodique)	55
Table VI:4: L'EAR et l'EFR (fenêtre non périodique)	55
Table VI:5: L'effet des paramètres : TP, PS, PC, PM et NPC	57
Table VI:6: Comparaison entre les erreurs relatives (fenêtre périodique)	58
Table VI:7: Comparaison entre l'erreurs relatives (fenêtre non périodique) 58
Table VI:8: Comparaison entre des RNFOAGs et RNAs	66
Table VI:9: L'effet de nombre d'entrée sur la précision d'extraction	66
Table VI:10: L'effet de variation de fréquence sur la précision d'extractior	า 67
Table VI:11: L'effet de variation d'amplitude sur la précision d'extraction.	69

INTRODUCTION GENERALE

Il a été démontré que tout signal périodique peut être décomposé en une somme de signaux sinusoïdaux d'amplitudes, de fréquences et de phases différentes. L'analyse spectrale est issue de cette théorie et cherche à déterminer la composition fréquentielle d'un signal, ce qui veut dire le calcul des différentes amplitudes, fréquences et phases. Cela va conduire à une autre représentation, autre que celle du domaine temporelle, avec laquelle on peut interpréter l'information incluse dans le signal considéré plus efficacement.

Une autre exigence dans ce domaine, ce qui devient très important, est la détermination en temps réel des harmoniques constituant un signal périodique. Cela est devenu un souci important dans beaucoup d'applications telles les systèmes de puissance et la conception des filtres.

Pour ce faire, plusieurs méthodes ont étés établies parmi lesquelles, des techniques dans le domaine temporel tels que les filtres IIR (Infinite impulse response) et d'autres techniques dans le domaine fréquentiel tels que la mesure récursive numérique pour la détermination on-line des harmoniques de système d'alimentation proposés par Girgis et autres, à base du filtre de Kalman (KF) et de la transformée de Fourier rapide (FFT).

Dans le cadre de ce mémoire de magister, nos travaux ont porté sur le développement d'une nouvelle méthode d'extraction des harmoniques basée sur les RNFOAG. Le mémoire est organisé comme suit :

Après une introduction générale,

Au premier chapitre, on rappelle les principales notions de traitement de signal et les différentes méthodes d'identification.

Au deuxième chapitre, on présente les points les plus importants concernant la logique floue, les RNAs et les AGs.

Le troisième chapitre concerne les différentes méthodes d'indentification pour l'analyse fréquentielle, utilisées lors de ce travail.

Finalement, au quatrième chapitre, on présente les différents résultats obtenus avec leur interprétation et leur comparaison avec d'autres méthodes.

Nous terminerons le mémoire par une conclusion générale.

Chapitre I:

NOTIONS FONDAMENTALES DANS LE TRAITEMENT DU SIGNAL ET LES METHODES D'IDENTIFICATIONS

1. Introduction:

Deux principaux types de signaux existent : les signaux déterministes et les signaux aléatoires. Les fonctions du traitement du signal peuvent être divisées en deux grandes catégories : l'élaboration des signaux et l'interprétation des signaux.

L'élaboration des signaux est l'intégration de l'information sous forme d'un signal par :

- Synthèse : par exemple, la création de signaux de forme appropriée en procédant à la combinaison de signaux élémentaires ;
- Modulation : changement d'amplitude, de fréquence, ou de phase permettant d'adapter un signal aux caractéristiques d'une voie de transmission;
- Codage : par exemple, la traduction de signaux en code binaire.

L'interprétation de signaux est l'extraction de l'information intégrée dans un signal par :

- Filtrage : c'est l'élimination de certaines composantes indésirables ;
- Détection : c'est l'extraction d'un signal bruité ;
- Identification : c'est le classement d'un signal dans des catégories préalablement définies ;
- Analyse : c'est isolement des composantes essentielles d'un signal de forme complexe ;
- Mesure : c'est l'estimation d'une grandeur caractéristique d'un signal avec un certain degré de confiance.

Les domaines de traitement du signal existants sont le traitement analogique et le traitement numérique. L'essor du numérique est survenu à cause d'un grand nombre d'avantages par rapport aux méthodes analogiques :

- Reproductibilité des systèmes ;
- Stabilité : pas de dérive en temps et en température ;
- Adaptabilité et souplesse d'emploi (modification du programme) ;
- Rapidité : en MHz et GHz.

On abordera dans ce chapitre quelques notions principales sur les signaux, les traitements effectués (transformée de Fourier, transformée de Laplace, transformée en z) aussi les méthodes d'identification des systèmes existants.

2. Analyse de Fourier :

2.1. Les séries de Fourier :

2.1.1. Le théorème de Fourier :

Sous certaines conditions de régularité qui seront toujours satisfaites en physique, une fonction f(t), périodique de période T et de pulsation

 $w=2\pi/T$ est décomposable en une série de fonctions sinusoïdales de pulsations n.w, (avec n entier naturel) et d'amplitudes déterminées, sous la forme :

$$f(t) = a_0 + \sum_{n=1}^{\infty} (a_n \cos(nwt) + b_n \sin(nwt)) \dots$$
 (I:1)

Ce développement est unique et appelé développement en série de Fourier de la fonction $f\left(t\right)$.

Les coefficients a_n et b_n sont les coefficients de Fourier de la fonction f(t). Le terme de pulsation w est le fondamental. Les termes de pulsation n.w avec n>2 sont les harmoniques de range n.

La série de Fourier converge vers f(t) partout où f est continue et

$$\operatorname{vers} \frac{f(t^+) + f(t^-)}{2}$$
 en chaque point où f admet une discontinuité. [1]

2.1.2. Propriétés des coefficients de Fourier :

• Signification de a_0 : a_0 représente la valeur moyenne de f:

$$a_0 = \frac{1}{T} \int_0^T f(t)dt$$
 (I:2)

 \bullet Calculs des coefficients $a_{\scriptscriptstyle n}$ et $b_{\scriptscriptstyle n}$: Pour tout $n \geq 1$, on a :

$$a_n = \frac{2}{T} \int_{t_0}^{t_0+T} f(t) \cdot \cos(nwt) \cdot dt$$
 (I:3)

Et:
$$b_n = \frac{2}{T} \int_{t}^{t_0+T} f(t).\sin(nwt).dt$$
 (I:4)

• Considérations de parité :

Si
$$f$$
 est paire, tous les b_n sont nuls et : $a_n = \frac{4}{T} \int_0^{T/2} f(t) \cdot \cos(nwt) \cdot dt \dots$ (I : 5)

Si
$$f$$
 est impaire, tous les a_n sont nuls et : $b_n = \frac{4}{T} \int_{0}^{T/2} f(t) \cdot \sin(nwt) dt$... (I : 6)

• Symétrie de glissement :

Lorsqu'une fonction de période T est changée en son opposé par translation d'une demi-période, son développement en séries de Fourier ne comporte que des harmoniques impaires.

• Décroissance des coefficients de Fourier :

Plus une fonction est régulière, plus ses coefficients de Fourier tendent rapidement vers zéro à l'infini. Inversement, plus la fonction périodique varie brutalement, plus les harmoniques élevées jouent un rôle important dans son développement en séries de Fourier. [2]

2.1.3. Spectre de Fourier d'une fonction :

Il est commode de représenter les séries a_n et b_n par des graphes où l'on porte en abscisse le rang n de l'harmonique (ou sa pulsation n.w) et verticalement un segment de hauteur a_n ou b_n , égal à l'amplitude de l'harmonique.

Chacune des séries de segments obtenues constitue un spectre de fréquence (ou spectre de Fourier) de la fonction f(t). L'utilité de ce spectre est qu'il montre immédiatement l'importance relative des harmoniques. Le spectre d'une fonction sinusoïdale ne comporte évidemment qu'une seule raie à la fréquence de la sinusoïde.

Considérant que :
$$a_n \cos(nx) + b_n \sin(nx) = S_n \cos(nx - \phi_n)$$
 avec : $S_n = \sqrt{a_n^2 + b_n^2}$ et $\tan \phi_n = \frac{b_n}{a_n}$

La connaissance séparée de a_n et b_n devient équivalente à celle de l'amplitude S_n et de la phase ϕ_n de l'harmonique de rang n. Dans de tels cas, on pourra représenter le développement par deux spectres : S_n et ϕ_n , appelés respectivement spectres d'amplitude et de phase [2]

2.1.4. L'énergie moyenne d'un signal :

L'énergie moyenne associée à une fonction périodique est égale à la somme des énergies moyennes associées à chacune de ses composantes de Fourier :

$$\langle f^2(t) \rangle = a_0^2 + \frac{1}{2} \sum_{n=1}^{\infty} (a_n^2 + b_n^2) \dots$$
 (I:7)

Cette égalité, reliant la valeur moyenne du carré du module de f à ses coefficients de Fourier est appelée égalité de Bessel-Parseval [1].

2.1.5. Exemples de décomposition en série de Fourier :

• Signal carré symétrique :

Ici
$$f(t)$$
 est impaire : $\forall n : a_n = 0, b_{2n} = 0$ et : $b_{2n+1} = \frac{4}{(2n+1)\pi}$

Le développement en séries de Fourier de f(t) s'écrit :

$$f(t) = \frac{4}{\pi} \left[\sin(wt) + \frac{\sin(3wt)}{3} + \frac{\sin(5wt)}{5} + \dots \right]$$
 (I:8)

• Signal triangulaire symétrique :

Ici
$$f(t)$$
 est paire, $\forall n : a_{2n} = 0, b_n = 0$ et : $a_{2n+1} = \frac{8}{((2n+1)\pi)^2}$

Le développement en séries de Fourier de f(t) s'écrit :

$$f(t) = \frac{8}{\pi^2} \left[\cos(wt) + \frac{\cos(3wt)}{9} + \frac{\cos(5wt)}{25} + \dots \right]$$
 (I:9)

2.2. Les transformées de Fourier :

Il reste à transposer la théorie concernant les signaux périodiques précédente au cas d'une fonction f(t) non périodique. Cela est possible à condition de prendre une durée T_1 très grande, jouant pour la fonction f(t) le rôle de la période des fonctions T_1 -périodiques. La pulsation fondamentale w_1 est alors très faible et les harmoniques nw_1 sont très rapprochés. On détermine alors les coefficients a_n et b_n en faisant :

$$a_n = \frac{1}{T_1} \int_{t_0}^{t_0+T_1} f(t) \cos(nw_1 t) dt \dots$$
 (I:10)

Posant :
$$w = nw_1$$
 alors : $a_n = \frac{1}{T_1} \int_{t_0}^{t_0+T_1} f(t) \cos(wt) dt$ (I : 11)

En faisant tendre T_1 vers l'infini, on fait la conclusion qualitative importante :

L'intégrale $\int_{-\infty}^{+\infty} f(t)\cos(wt)dt$ est proportionnelle à la partie sinusoïdale de pulsation w de la fonction f(t). Idem pour les termes b_n . Finalement, en

utilisant la représentation complexe, on définit la transformée de Fourier d'une fonction f(t) par :

$$F(w) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} f(t) \exp(jwt) dt \dots$$
 (I:12)

Par le théorème de réciprocité on démontre le résultat suivant :

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} F(w) \exp(-jwt) dw \dots \qquad (I:13)$$

En pratique et plus particulièrement pour les signaux acquis via un ordinateur ou un oscilloscope numérique, le spectre résulte d'un calcul fait sur des nombres obtenus par échantillonnage et quantification du signal analogique d'origine.

Le calcul de la transformée de Fourier sur la série échantillonnée conduit à la définition de la Transformée de Fourier en temps discrète (TFTD) [3].

2.3. La transformée de Fourier à temps discret (TFTD) :

Lorsqu'on échantillonne un signal s(t) à temps continu, on obtient une

suite
$$s_e(n) = s(nT_e)$$
 et $F_e = \frac{1}{T_e}$ représente la fréquence d'échantillonnage

La TFTD d'une suite $\{s(n)\}$, issue de l'échantillonnage d'un signal s(t), est la fonction périodique de période 1 de la variable réelle f définie par :

$$S(f) = \sum_{n=-\infty}^{+\infty} s(n) \exp(-2j\pi f n)$$
 (I: 14)

A partir de S(f), la TFTD inverse permet de revenir à la suite $\{s(n)\}$ et par définition : Si S(f) est une fonction périodique de période 1 et

$$\sin \int_{0}^{1} |S(f)|^{2} df < +\infty$$
, alors [4]:

$$s(n) = \int_{-1/2}^{1/2} S(f) \exp(2j\pi f n) df \qquad (I:15)$$

2.3.1. Propriétés de la TFTD :

Considérons S(f) la TFTD de la suite $\{s(n)\}$ et V(f) la TFTD de la suite $\{v(n)\}$:

- Linéarité : $a.s(n) + b.v(n) \rightarrow a.S(f) + b.V(f)$
- Translation en temps : $s(n-n_0) \rightarrow S(f) \cdot \exp(-2j\pi f n_0)$
- Modulation : $s(n) \cdot \exp(2j\pi f_0 n) \rightarrow S(f f_0)$

- Renversement temporel : $s(-n) \rightarrow S(-f)$
- Conjugaison : $s^*(n) \rightarrow S^*(f)$
- Suite réelle : s(n) réelle $\to S(f) = S^*(-f)$ On dit que S(f) possède la propriété de symétrie hermitienne. Aussi, |S(f)| et $\mathrm{Re}[S(f)]$ sont des fonctions paires $\mathrm{etarg}[S(f)]$ et $\mathrm{Im}[S(f)]$ sont des fonctions impaires. Dans ce cas, nous pourrons limiter la représentation à $f \in [0,0.5]$
- Produit de convolution : $s(n) * v(n) = \sum_{k=-\infty}^{+\infty} s(k).v(n-k) \rightarrow S(f).V(f)$
- Formule de Parseval : $\sum_{n=-\infty}^{+\infty} \left| s(n) \right|^2 = \int_{-1/2}^{1/2} \left| S(f) \right|^2 df$

Le calcul par ordinateur de la TFTD, à partir des échantillons s(n) impose une charge de calcul infinie car la suite est infinie et la fréquence f varie continûment sur l'intervalle [0,1]. Cette transformée n'est donc pas réalisable d'un point de vue purement numérique. C'est pourquoi, la TFTD cède le pas à la TFD (Transformée de Fourier Discrète) [4]

2.4. Transformée de Fourier Discrète (TFD) :

Elle se calcule sur la série de nombres réels ou complexes s(n), correspondant aux valeurs de s(t) prélevées aux instants nT_e où n est un entier appelé indice temps et T_e est la période d'échantillonnage.

Par définition, on appelle TFD de la suite $s(n), n \in \{0,1,...,N-1\}$ la suite $S(k), k \in \{0,1,...,N-1\}$, définie par :

$$S(k) = \sum_{n=0}^{N-1} s(n) \cdot \exp\left[-2j\pi n(\frac{k}{N})\right] = \sum_{n=0}^{N-1} s(n) \cdot W_N^{nk}$$
 (I: 16)

Avec : $W_N = \exp(-2j\pi/N)$ est une racine nième de l'unité, c.à.d. $W_N^N = 1$ La transformée inverse [2] :

$$s(n) = \frac{1}{N} \sum_{k=0}^{N-1} S(k) \cdot W_N^{-nk}$$
 (I:17)

2.4.1.Propriétés de la TFD :

Les propriétés de la TFD présentent des grandes similitudes avec la TFTD, cependant, dans le cas de la TFD, tous les calculs se font modulo N.

A partir de l'expression de la transformée de Fourier, il est clair qu'il y a plusieurs opérations arithmétiques à effectuer lors du calcul de la TFD par un calculateur numérique. Ce qui impose le problème de temps. Pour cela, il est nécessaire de trouver une manière rapide de calcul. Ce qui est fait par la transformée de Fourier rapide (TFR) [2].

2.5. Transformée de Fourier Rapide (TFR, FFT) :

C'est un algorithme permettant de réduire le temps nécessaire pour effectuer un nombre d'opérations élémentaires MAC (Multiplication Addition aCcumulation). Ce temps de calcul est primordial pour réaliser des systèmes temps réel.

La TFD se calcule par :

$$S_n = \sum_{k=0}^{N-1} S_k W_N^{nk} \quad ; n \in \{0,1,...,N-1\}.$$
 (I: 18)

Aussi pour un n donné, il y a N multiplications et N-1 additions complexes et pour les N points de la transformée, nous obtenons :

- N.N Multiplications complexes;
- N.(N−1) Additions complexes.

Le temps de calcul d'une addition est négligeable devant celui de la multiplication. Aussi le coût de calcul de TFD est N^2 .

Différents algorithmes ont été développés pour réduire le coût de calcul. Le plus connu, celui de Cooley-Tuckey (1965) appelé méthode d'entrelacement temporel qui a réduit le nombre de multiplication complexe à $\frac{N}{2}\ln(N)$ [4].

2.5.1. Algorithme de Cooley-Tuckey:

Aussi appelé algorithme de réduction à base 2 dans le domaine temporel. Il s'applique dans le cas où $N=2^k$. Il permet alors une décomposition dichotomique du calcul.

On peut déduire que :

•
$$W_N^{2nk} = W_{N/2}^{nk}$$

• Et pour
$$n < N/2$$
: $W_N^{(nk+N/2)} = -W_N^{nk}$

Pour calculer les N échantillons de la $\mathsf{TFD}\{S_0, S_1, ..., S_{N-1}\}$, on utilise l'expression de base (I : 18).

En séparant les échantillons d'ordre pair et d'ordre impair, on obtient :

$$S_{k} = \sum_{i=0}^{\frac{N}{2}-1} s_{2i} W_{N}^{2ik} + \sum_{i=0}^{\frac{N}{2}-1} s_{2i+1} W_{N}^{(2i+1)k} = \sum_{i=0}^{\frac{N}{2}-1} s_{2i} W_{N/2}^{ki} + W_{N/2}^{k} \sum_{i=0}^{\frac{N}{2}-1} s_{2i+1} W_{N/2}^{ki}$$

$$S_{k} = S_{1(N/2)k} + W_{N}^{k} S_{2(N/2)k}$$
(I: 19)

Avec:

 $S_{\mathrm{l}(N/2)k}$: Transformée d'ordre N/2 effectuée sur les échantillons pairs.

 $S_{2(N/2)k}\,$: Transformée d'ordre $N/2\,$ sur les échantillons impairs.

Ainsi nous avons à calculer 2 transformées d'ordre N/2 et N/2 multiplications complexes.

Nous pouvons constater que le calcul de la TFD a été divisé en deux. Or,

$$S_{k+\frac{N}{2}} = S_{1(N/2)k} + W_N^{k+\frac{N}{2}} S_{2(N/2)k} = S_{1(N/2)k} - W_N^k S_{2(N/2)k}$$
 (I: 20)

Donc pour:

$$0 \le k \le \frac{N}{2} - 1$$
 , on a: $S_k = S_{1(N/2)k} + W_N^k . S_{2(N/2)k}$ (I: 21)

$$N/2 \le k \le N-1$$
, on a : $S_k = S_{1(N/2)k} - W_N^k . S_{2(N/2)k}$ (I : 22)

Le calcul d'une TFD d'ordre N conduit à effectuer deux TFD d'ordre N/2. Il est schématisé en figure suivante :

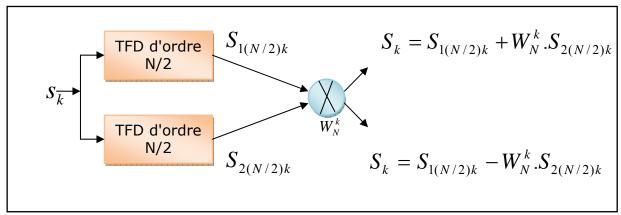


Figure 0:1: TFD d'ordre N à partir de 2 TFD d'ordre N/2.

Par ce schéma, on peut calculer les transformée d'ordre N/2, N/4, N/i avec $i=2^L$.

Cette analyse peut donc être réalisée L fois si $N=2^L$. On obtient à la fin de ce processus dichotomique une transformée d'ordre 2 qui ne demande pas de multiplication car : $W_2^1=-1;W_2^0=1$ Ainsi, l'algorithme complet peut être représenté par L étapes de calcul de type papillon commençant par les N/2 calculs d'ordre 2.La figure suivante illustre les décompositions nécessaires pour obtenir une transformée de Fourier d'une série s; $n \in \{0,1,...N-1\}$ avec N=8 échantillons.

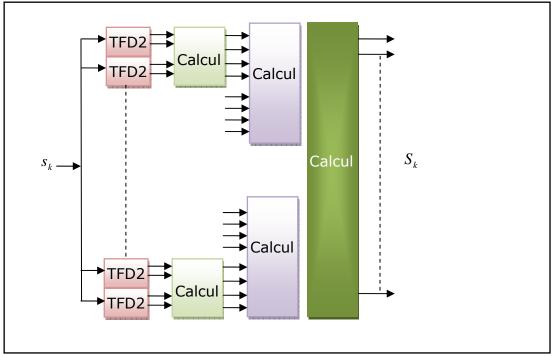


Figure 0:2: Représentation schématique de la TRF.

Par cet algorithme, on trouve au lieu de N^2 multiplications complexes et N.(N-1) additions complexes pour la TFD calculée directement par l'équation (I : 16) : $\frac{N}{2}.\log_2(N)$ multiplications complexes et $N.\log_2(N)$ addition complexes [4].

3. La Transformée de Laplace :

3.1. Définition :

Si f(t) désigne une fonction à valeurs réelles ou complexes de la variable réelle t, définie sur le domaine $t \in]0,+\infty[$ et nulle pour t < 0; on appelle Transformée de Laplace de f(t) la fonction :

$$F(p) = L[f(t)] = \int_{0}^{+\infty} f(t) \cdot \exp(-pt) dt$$
 (I:23)

Où p est complexe.

L'existence de F(p) suppose la convergence de l'intégrale On dit que F(p) est " l'image " de f(t)

Conditions d'existence de F(p):

- f(t) Doit être continue par morceau ;
- $\exists \alpha \text{ avec } 0 < \alpha < 1 \text{ tel que } : \lim_{t \to 0} t^{\alpha} \left| f(t) \right| \to 0$
- f(t) doit être d'ordre exponentiel.

Conséquences:

Certaines fonctions f(t) ne possèdent pas de transformée de Laplace, par exemple $f(t) = \frac{1}{t}$ qui ne respecte pas la deuxième condition d'existence ou $f(t) = \exp(t^2)$ qui ne respecte pas la troisième condition. Inversement, toutes les fonctions F(p) ne sont pas des transformées de Laplace. On montre que, si F(p) est une transformée de Laplace, alors $\lim_{n \to \infty} F(p) \to 0$ [5].

3.2. Propriétés de la Transformée de Laplace :

On note $F(p) = L[f(t)] = \int\limits_0^{+\infty} f(t) . \exp(-pt) dt$ la fonction dite image de f(t) et $f(t) = L^{-1}[F(p)]$ la fonction dite original $\det F(p)$. Sous réserve des conditions d'existence, la correspondance $F(p) \leftrightarrow f(t)$ est unique.

Linéarité :

Si
$$F_1(p) = L[f_1(t)]; F_2(p) = L[f_2(t)]$$
 alors:
 $L[a.f_1(t) + b.f_2(t)] = a.F_1(p) + b.F_2(p)$

• Transformée de Laplace de la fonction dérivée :

Si
$$f(t)$$
 est continue, alors $L[f'(t)] = p.F(p) - f(0^+)$ où $f(0^+) = \lim_{t \to 0} f(t)$;

Plus généralement, si f(t) est discontinue aux points $\{t_1, t_2, ..., t_n\}$

$$L[f'(t)] = p.F(p) - f(0^{+}) - \sum_{k=1}^{n} \exp(-pt_{k})[f(t_{k}^{+}) - f(t_{k}^{-})]$$

Les expressions ci-dessus se généralisent aux dérivées d'ordre n et, en particulier, si la fonction f(t) et ses dérivées sont continues, on obtient :

$$L[f^{(n)}(t)] = p^{n}.F(p) - p^{n-1}f(0^{+}) - p^{n-2}f'(0^{+}) - \dots - pf^{(n-2)}(0^{+}) - f^{(n-1)}(0^{+})$$

• Transformée de l'intégrale :

$$h(t) = \int_{0}^{t} f(u) du$$

$$h'(t) = f(t)$$
; On pose $H(p) = L[h(t)]$ et $F(p) = L[f(t)]$

$$L[h'(t)] = p.H(p) - h(0^+) = pH(p) = L[f(t)] \Rightarrow H(p) = \frac{F(p)}{p}$$

En généralisant, on obtient $L[\int_{0}^{t} \int_{0}^{t} ... \int_{0}^{t} f(u) du] = \frac{F(p)}{p}$

• Dérivation et intégration par rapport à un paramètre :

Soit
$$F(p,a) = L[f(t,a)] = \int_{0}^{+\infty} f(t,a) \cdot \exp(-pt) dt$$

$$L\left[\frac{\partial f(t,a)}{\partial a}\right] = \int_{0}^{+\infty} \frac{\partial f(t,a)}{\partial a} \cdot \exp(-pt)dt = \frac{\partial}{\partial a} \int_{0}^{+\infty} f(t,a) \cdot \exp(-pt)dt \Rightarrow L\left[\frac{\partial f(t,a)}{\partial a}\right] = \frac{\partial F(p,a)}{\partial a}$$

Soit
$$g(t) = \int_{0}^{\beta} f(t, a) da$$

$$L[g(t)] = \int_{0}^{+\infty} \int_{\alpha}^{\beta} f(t,a) da \cdot \exp(-pt) dt = \int_{\alpha}^{\beta} \int_{0}^{+\infty} f(t,a) \cdot \exp(-pt) dt da = \int_{\alpha}^{\beta} F(p,a) da$$

D'où
$$L[\int_{\alpha}^{\beta} f(t,a)da] = \int_{\alpha}^{\beta} F(p,a)da$$

• Théorème du retard :

Soit
$$f(t) = \begin{cases} 0; t < \tau \\ f(t-\tau); t > \tau \end{cases}$$

alors
$$L[f(t-\tau)] = \int_{0}^{+\infty} f(t-\tau) \cdot \exp(-pt) dt = \int_{\tau}^{+\infty} f(t-\tau) \cdot \exp(-pt) dt$$

En faisant le changement de variable $v=t-\tau$, on montre que $L[f(t-\tau)]=F(p).\exp(-p\,\tau)$

- Théorème de l'amortissement : L[f(t).exp(-kt)] = F(p+k)
- Changement d'échelle de temps : $L[f(kt)] = \frac{1}{k}F(\frac{p}{k})$ Avec k positif.
- Dérivation et intégration de la fonction symbolique F(p):

$$F^{(n)}(p) = L[(-1)^n t^n f(t)]$$

- Théorème de la valeur initiale : $\lim_{t \to 0^+} f(t) = \lim_{p \to +\infty} pF(p)$
- Théorème de la valeur finale : $\lim_{t \to +\infty} f(t) = \lim_{p \to 0} pF(p)$
- Théorème de convolution : $h(t) = g(t) * f(t) = \int_{0}^{t} f(u)g(t-u)du$

$$H(p) = F(p).G(p)$$

4. Transformée en Z [6]:

4.1. Définition de la transformée en Z :

Pour les suites discrètes f(k); $k \in \mathbb{Z}$ il existe une transformation appelée transformée en \mathbb{Z} . Cette transformation s'applique aux équations récurrentes. Elle est définie par :

$$F(z) = Z[f(k)] = \sum_{k=0}^{+\infty} f(k).z^{-k}$$
 (I: 24)

4.2. Propriétés de la transformée en Z :

• Linéarité :

$$Z[\lambda_1.f_1 + \lambda_2f_2] = \lambda_1Z(f_1) + \lambda_2Z(f_2)$$

- Translation temporelle :
- 1. Cas du retard : $Z[f(t-kT)] = z^{-k} \cdot F(z)$

Démonstration:

$$Z[f(t-kT)] = \sum_{0}^{\infty} f((n-k)T)z^{-n} = z^{-k} \sum_{0}^{\infty} f((n-k)T)z^{-(n-k)}$$

$$Z[f(t-kT)] = z^{-k} \cdot \sum_{m=-k}^{\infty} f(mT)z^{-m}$$

Comme: f(mT) = 0 pour m < 0 on obtient:

$$Z[f(t-kT)] = z^{-k} \cdot \sum_{m=0}^{\infty} f(mT)z^{-m} = z^{-k}.F(z)$$

2. Cas de l'avance :

$$Z[f(t+kT)] = z^{k}.F(z) - z^{k}.f(0.T) - z^{k-1}.f(T)... - z.f((k-1)T)$$

En effet:

$$Z[f(t+kT)] = \sum_{n=0}^{\infty} f((n+k)T)z^{-n} = z^{k} \sum_{n=0}^{\infty} f((n+k)T)z^{-(n+k)}$$

$$= z^{k}.(F(z) - \sum_{m=0}^{k-1} f(mT) \cdot z^{-m})$$

- Théorème de la valeur initiale : $\lim_{n \to \infty} f(nT) = \lim_{z \to \infty} F(z)$
- Valeur finale : $\lim_{n \to \infty} f(nT) = \lim_{z \to \infty} f(nT) = \lim_{z \to \infty} f(z-1) \cdot F(z) = \lim_{z \to \infty} f(z-$
- Théorème de sommation : $Z\left[\sum_{k=0}^{n} f(kT)\right] = \frac{z}{z-1} \cdot F(z)$

Pour la démonstration, posons : $g(nT) = \sum_{n=0}^{\infty} f(kT) = g((n-1)T) + f(nT)$

$$Z[g(nT)] = Z[\sum_{n=0}^{\infty} f(kT) = g((n-1)T) + f(nT)]$$

$$G(z)=z^{-1}.G(z)+F(z)$$

D'ou :
$$G(z) = \frac{1}{1 - z^{-1}} \cdot F(z)$$

4.3. Tableau de quelques transformées en Z :

On donnera les transformées en Z que l'on utilisera dans les équations récurrentes.

Fonction Récursive	Transformée en Z
f(k)	F(z)
f(k-1)	z.F(z)
f(k+1)	$z^{-1}F(z)$
u(t) échelon	$\left \frac{z}{z-1}\right $; pour $ z > 1$
$\sin(wt)$	$\frac{z\sin(wT)}{z^2 - 2z\cos(wT) + 1} \text{ ; pour } z > 1$
$\cos(wt)$	$\frac{z(z-\cos(wT))}{z^2-2z\cos(wT)+1} \text{ ; pour } z >1$
$\delta(t)$	1; pour $ z > 1$
t	$\frac{Tz}{(z-1)^2} ; pour z > 1$
$\exp(-a.t)$	$\frac{z}{z - \exp(-aT)} \text{ pour } z > \exp(-aT)$

Table 0:1: Quelques transformées en Z [7].

En particulier, on a :
$$\sum_{k=0}^{\infty} f(k+1)z^n = z^{-1} \big[f(z) - f(0) \big]$$

5. Les méthodes d'identification [8] :

Un système est un objet dans lequel des variables de différents types interagissent et produisent des signaux observables. Lorsque le modèle du système n'est pas connu, il est nécessaire de procéder à son identification. Cette identification est réalisée par l'obtention d'un modèle mathématique du système réel en réalisant une modélisation.

La modélisation réalisée à partir d'un comportement du système et / ou de lois physiques, consiste à déterminer la structure des équations qui régissent le comportement de ce système, et aussi, à fixer, a priori la valeur de ses paramètres (longueurs, masses, inerties, capacités, résistances, frottements...). Mais, il est souvent impossible d'obtenir une connaissance à priori complète et précise de tous les paramètres du modèle. Pour affiner et compléter cette connaissance, il est alors nécessaire de procéder à une identification du système à partir des réactions de celui-ci à des sollicitations données et connues. On peut, si le système est observable, identifier les paramètres encore inconnus.

5.1. Approche d'identification:

La procédure d'identification se compose principalement de quatre étapes schématisées par la figure suivante [9] :

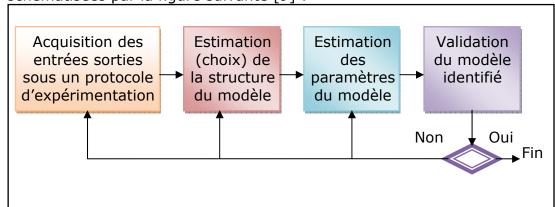


Figure 0:3: Procédure d'identification.

5.2. Les types d'identification :

On distingue deux approches pour l'identification des systèmes :

- Approche en boucle ouverte ;
- Approche en boucle fermée.

5.2.1. Approche en boucle ouverte :

Majoritairement utilisée car beaucoup de procédés peuvent fonctionner en boucle ouverte. Un exemple concret est celui de la diffusion vidéo numérique. Le codage de l'image à transmettre, utilise des algorithmes de compression faisant appel aux techniques d'identification. Il s'agit en particulier de prédire et estimer les mouvements de l'image afin de transmettre uniquement les informations nouvelles par rapport à l'image précédente. Cette approche :

- est utilisée lorsque le système est naturellement stable, et si les contraintes d'exploitation et de sécurité le permettent
- permet d'appliquer des signaux de test en entrée d'un modèle en observant la réponse en sortie, afin d'identifier le plus directement

possible les paramètres du procédé. Déterminer les paramètres du processus considéré en boucle ouverte est un des objectifs premiers de l'identification

- est souvent préférée à l'approche en boucle fermée : la boucle de retour pose des problèmes dans l'identification
- pour des systèmes multivariables, la structure de chaque transfert est personnalisable (retard, pôles, zéros)
- le concept n'est pas limité aux modèles dynamiques linéaires et s'applique de la même manière aux systèmes décrits par des équations différentielles non linéaires. L'identification des paramètres électriques d'un moteur asynchrone en est un exemple.

5.2.2. Approche en boucle fermée [10] :

C'est une alternative souvent incontournable en milieu industriel. Elle y est utilisée depuis longtemps. Beaucoup de procédés ne peuvent en fait, fonctionner qu'en boucle fermée. C'est le cas pour :

- des impératifs de production ou de sécurité, ou pour des raisons économiques.
- si le processus est instable en boucle ouverte ou à un comportement intégrateur.
- il peut arriver aussi pour certains systèmes, qu'une dérive importante du point de fonctionnement en boucle ouverte ait lieu. La capacité d'un régulateur à linéariser le comportement du procédé autour d'un point de fonctionnement en boucle fermée constitue un point fort.

Ce constat s'applique à de nombreux systèmes de production (fabrication du papier, du verre...), mais aussi à des systèmes tels que des robots manipulateurs ou les servomécanismes (exemple : positionnement de disques CD, DVD, CD-ROM...) et également aux domaines non techniques, tels que dans les systèmes biologiques ou économiques. Ses avantages et caractéristiques :

- sont motivés par des contraintes pratiques. Exemple : réaliser la maintenance d'un contrôleur existant sans ouvrir la boucle ou réréglage sur site. Avec le vieillissement, les paramètres et le modèle d'un processus peuvent être modifiés,
- permet d'obtenir de meilleurs modèles pour la commande des systèmes (une identification en basse fréquence (BF) et donne de meilleurs résultats pour un système fonctionnant normalement en BF.).
- permet la validation de la commande d'un système : comparaison entre performances souhaitées et réelles dans les domaines temporels et fréquentiels, calcul de pôles et tests statistiques,
- elle est particulièrement favorable à la synthèse d'un régulateur robuste,
- permet un meilleur contrôle de la puissance d'entrée (et de sortie) pendant l'expérience,
- permet de réduire l'ordre du modèle, afin de reconcevoir la commande (modification du contrôleur) en conservant les performances souhaitées en boucle fermée,
- permet l'identification en temps réel en boucle fermée et l'adaptation de la commande.

5.3. Type de modèles utilisés pour l'identification [11] :

La notion de modèle mathématique d'un système, d'un processus ou d'un phénomène, est un concept fondamental. Il existe une multitude de types de modèles, chacun étant destiné à une application particulière. Nous pouvons les décliner en deux grandes catégories :

- Les modèles de connaissance (basés sur les lois de la physique, de la chimie...), donnent une description complète des systèmes et sont utilisés pour la simulation et la conception des procédés. Ce sont souvent des modèles complexes;
- Les modèles dynamiques de commande, qui donnent la relation entre les variations des entrées d'un système et les variations de la sortie.

Les modèles dynamiques sont de deux sortes :

- 1. Modèles non paramétriques (réponse fréquentielle, réponse à un échelon);
- 2. Modèles paramétriques (fonction de transfert, équations différentielles).

5.3.1. Les modèles non paramétriques :

• Analyse fréquentielle :

Les essais expérimentaux harmoniques sont rarement employés, car ils sont longs et fastidieux. Des essais réalisés avec une pulsation comprise entre $0.1\omega_c$ et $10\omega_c$, avec une grande constate de temps pour le processus, peuvent durer des heures, d'autant plus qu'il est parfois nécessaire de recommencer deux à trois fois l'expérimentation pour chaque pulsation (on doit attendre le temps d'établissement pour chaque mesure). Quelques outils classiques utilisés pour l'analyse de la réponse en fréquence du système à identifier sont : Bode, Nyquist et Black.

• Réponse impulsionnelle :

Un système linéaire invariant est entièrement caractérisé par sa réponse à une impulsion de Dirac, dont la transformée de Fourier est constante sur tout le spectre. La transformée de Fourier de la réponse impulsionnelle du système est sa fonction de transfert. Un Dirac « réel », peut être approché, du point de vue de ses effets, par une impulsion brève, de durée finie et d'énergie limitée. La réponse du système, est en conséquence faible et bruitée, donc difficilement exploitable. Ainsi, pour atteindre la réponse impulsionnelle h(t) par voie indirecte. Il est possible d'appliquer les techniques de corrélation. Le processus est alors excité par une Séquence Binaire Pseudo Aléatoire (SBPA.). Notons que l'utilisation d'un générateur de S.B.P.A. pour l'essai ou la simulation d'un système, est aussi connue sous le nom de techniques de Monte-Carlo (très largement utilisées par les physiciens lors de simulations). Ajoutons, que les signaux aléatoires sont caractérisés par leurs propriétés statistiques (alors que les signaux d'excitation utilisés précédemment étaient de nature déterministe).

La fonction d'auto corrélation est définie :
$$\varphi_{xx}(\tau) = \lim_{t \to \infty} \frac{1}{T} \int_{t_0 - \frac{T}{2}}^{t_0 + \frac{T}{2}} x(t)x(t+\tau)dt \dots \qquad (I:25)$$

I.e., si on compare dans l'intervalle défini, un signal x(t) avec lui-même décalé $\det T$, on conçoit que cela permette de voir en quoi le signal à un instant donne est lié à ce qui s'est passé un instant τ avant.

La fonction d'inter corrélation est donnée par :
$$\varphi_{xy}(\tau) = \lim_{t \to \infty} \frac{1}{T} \int_{t_0 - \frac{T}{2}}^{t_0 + \frac{T}{2}} x(t) y(t + \tau) dt \dots (I:26)$$
Le si en sempare deux signaux $x(t)$ et $y(t)$ par exemple l'excitation et

I.e., si on compare deux signaux x(t) et y(t), par exemple l'excitation et la réponse d'un système linéaire, on peut se demander, si de la connaissance de l'histoire de x(t) et de y(t) pendant le même intervalle de temps T, on peut déduire une relation entre les deux signaux. Si le signal x(t) est tel que sa fonction de transfert est assimilable à une impulsion de Dirac, alors $\varphi_{xy}(T) = h(T)$.

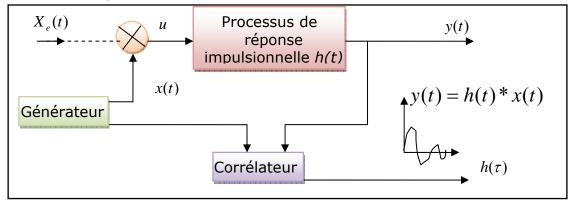


Figure 0:4: Principe de l'analyse par corrélation.

Notons que la transformée de Fourier d'une corrélation temporelle, a pour résultat une corrélation fréquentielle (densité spectrale). La corrélation peut être utilisée, par exemple, pour des études vibratoires (en mécanique). Cette méthode est toutefois sensible aux perturbations. Les diverses méthodes non paramétriques (analyse fréquentielle, réponse indicielle, impulsionnelle...) peuvent aider à choisir la structure du modèle. Néanmoins, les méthodes qui prennent en compte un modèle paramétrique du processus, sont préférées, car elles sont plus simples à mettre en œuvre et à utiliser.

5.3.2.Les modèles paramétriques :

Il existe deux principales classes de méthodes paramétriques :

- Les méthodes paramétriques graphiques ou déterministes ;
- Les méthodes paramétriques statistiques (études stochastiques).

• Les méthodes paramétriques graphiques :

L'objectif est d'obtenir des modèles paramétriques à partir de modèles non paramétriques, type « réponse en échelon » ou « réponse en fréquence ».

a. Identification d'un système du premier ordre :

Citons trois principaux modèles :

- Modèle apériodique de Strejc : $G(s) = \frac{[k \cdot \exp(-Ts)]}{(1+Ts)^n}$ (I : 27)
- Modèle de Broida : $G(s) = [k.\exp(-Ts)]/(1+Ts)$ (I : 28)

• Modèle de Ziegler et Nichols : $G(s) = (R/s).\exp(-Ls)$ avec :

$$\hat{L} = \tau \text{ et } \hat{R} = K/T_1...$$
 (I : 29)

Nous illustrons ces méthodes ci-dessous :

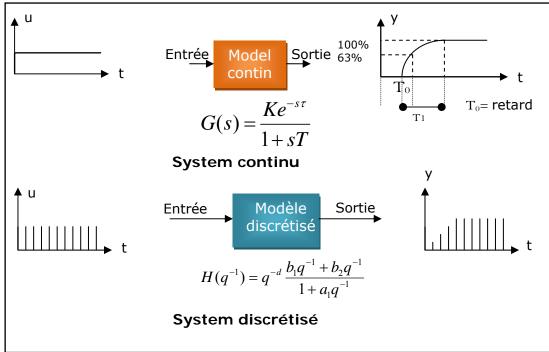


Figure 0:5: Réponse d'un procédé continu et discrétisé.

b. Identification d'un système du deuxième ordre :

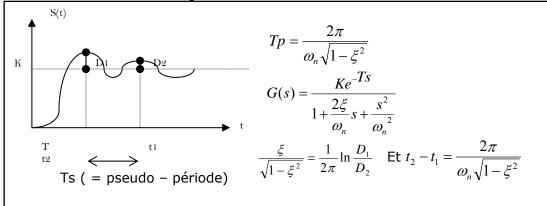


Figure 0:6: Réponse indicielle d'un système du deuxième ordre.

A l'aide des éléments graphiques on peut ainsi déterminer la fonction de transfert G(p)

Notons que ces essais, utilisés pour la boucle ouverte, peuvent être appliqués pour la boucle fermée :

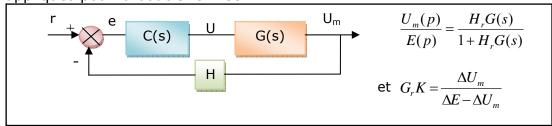


Figure 0:7: Système en boucle fermée.

Les méthodes que nous venons de décrire sont simples, et ne nécessitent que peu de matériel, en revanche :

- elles sont limitées dans le choix des modèles,
- elles nécessitent des signaux tests d'amplitude importante (rarement tolérés par les installations industrielles),
- elles sont sensibles aux perturbations (et il n'est pas possible ici de modéliser les perturbations),
- elles sont longues, et il n'y a pas de procédures explicites de validation des modèles,
- en boucle ouverte, les procédures d'identification que nous venons de décrire, sont assez imprécises,
- la méthode proposée à partir du modèle de Strejc, ou de celui de Broida, permet de déduire l'ensemble des paramètres du modèle. En boucle fermée le mode opératoire utilisé consiste à placer le système à la juste instabilité (déphasage de $-\pi$ et gain de boucle de -1: ce sont des valeurs exactes. Si la boucle fermée comporte un intégrateur, il doit en être tenu compte).

• Les méthodes paramétriques statistiques :

Il s'agit de mettre en place des algorithmes d'estimation automatique des paramètres des modèles échantillonnés des procédés en utilisant un calculateur numérique. Le traitement des données entrées / sorties peut se faire à l'aide d'algorithmes non récursifs (Traitement en bloc des fichiers de données obtenues sur un horizon de temps) ou récursifs (Traitement pas à pas des données), opérant avec des signaux d'excitation extrêmement faibles. Le second type d'algorithme, présente l'avantage d'être « en temps réel » et nécessite moins de ressources mémoire en terme de calculs.

L'identification paramétrique statistique est généralement basée sur l'erreur de prédiction, qui est l'erreur entre la sortie du procédé y(t), et la sortie $\hat{y}(t)$ prédite par le modèle. Cette erreur de prédiction est utilisée par un Algorithme d'Adaptation Paramétrique (AAP.), qui à chaque instant d'échantillonnage, va modifier les paramètres du modèle afin d'en minimiser l'erreur. L'entrée appliquée au système est généralement une Séquence Binaire Pseudo Aléatoire (S.B.P.A.) de très faible niveau, engendrée par le calculateur.

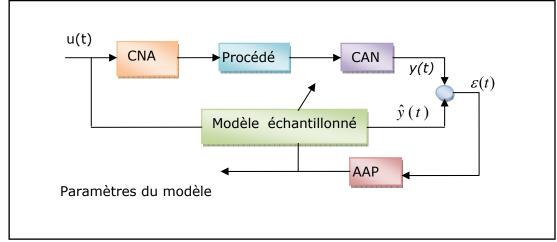


Figure 0:8: Principe de l'estimation des paramètres d'un modèle.

Cette approche moderne de l'identification :

- élimine les défauts mentionnés précédemment et permet un suivi des paramètres en temps réel,
- permet de modéliser (et supprimer si besoin) les perturbations et bruits capteurs, et permet un traitement aisé des signaux (analyse spectrale des signaux).

Les algorithmes non récursifs ont pour objectif la minimisation de l'erreur entre la sortie du procédé et la sortie de prédiction au sens des moindres carrés. Nous décrivons dans ce qui suit, seulement les algorithmes récursifs.

6. Conclusion:

L'analyse spectrale par TFD est une technique utilisée largement dans plusieurs applications implémentant des algorithmes de calcul rapides (TFR ou FFT). Ces algorithmes sont suffisamment rapides pour être utilisés dans des applications en temps réel. Cependant, cette technique d'analyse spectrale a un très grand inconvénient qui est sa faible résolution spectrale, particulièrement si l'échantillon du signal est court et avec un faible rapport signal sur bruit. Par conséquent d'autres techniques sont nécessaires pour une utilisation efficace, surtout en temps réel.

Chapitre II:

LES ALGORITHMES GENETIQUES, LA LOGIQUE FLOUE ET LES RESEAUX DE NEURONES

1. Les algorithmes génétiques :

1.1. Introduction:

Les algorithmes génétiques (AG) ont été proposés par JOHN HOLLAND dans les années 70 [12] [13]. Ils s'inspirent de l'évolution génétique des espèces, plus précisément du principe de sélection naturelle. La nature ayant conduit à des organismes particulièrement robustes (capacité de survive dans des conditions parfois extrêmement difficiles) et adaptables. On essaie de la copier pour obtenir des algorithmes de recherche d'optimum.

La fonction dont on recherche l'optimum est dite fonction objectif. On peut remarquer à présent que l'on ne fait aucune hypothèse sur cette fonction, en particulier elle n'a pas besoin d'être dérivable, ce qui représente un avantage sur certaines méthodes de recherche d'extrémum comme les méthodes du premier et du second ordre (gradient, Newton,...). Pour résoudre un problème quelconque, on devra l'exprimer sous la forme d'une fonction objective, ce qui est parfois difficile. Par ailleurs des dispositions particulières devront être prises lorsque le problème à traiter est avec contraintes.

1.2. Principe de base :

Les AG manipulent une population d'individus de taille constante, cette population est soumise à une compétition entre les individus. Chaque individu est donné sous forme d'une seule chaîne de caractères appelée chromosome et qui représente un point de l'espace de recherche [14]. L'AG fait évoluer cette population d'individus au cours des générations à l'aide d'opérateurs de sélection, de croisement et de mutation qui s'inspirent des phénomènes naturels. Le croisement et la mutation sont chargés d'explorer l'espace de recherche en construisant de nouveaux individus à partir de la génération précédente, alors que la sélection

La structure générale d'un algorithme génétique est la suivante :

favorise les individus qui possèdent une adaptation élevée.

- (1). Générer aléatoirement une population d'individus de taille donnée M;
- (2).Faire:
- Sélectionner et apparier les individus ;
- Recombiner deux individus appariés pour créer deux nouveaux individus;
- Faire muter certains individus ;
- Créer une nouvelle génération en remplaçant tout ou une partie des anciens individus par les nouveaux ;
 - (3). Y a t-il un individu qui peut être l'optimum?
 - (4).Oui: aller en 6;
 - (5).Non: retour à 2;
 - (6).Fin.

1.3. Opérateurs génétiques :

1.3.1. Codage d'un individu :

Un individu représente un point de l'espace de recherche du problème traité, c'est-à-dire une évaluation de l'ensemble des variables du problème. Mises bout à bout, ces variables, appelées gènes, forment le chromosome ou l'individu, et leurs différentes valeurs possibles sont les allèles. Les AG utilisent traditionnellement des chaînes de bits binaires, entiers ou réels de taille finie pour représenter les chromosomes comme illustré dans la figure suivante :



Figure 0:1: Codage des individus.

1.3.2. Génération de la population initiale :

L'étape d'initialisation d'un \overline{AG} nécessite la génération de M éléments de population où M est un nombre préalablement fixé, et constant au cours des générations successives. Néanmoins, si on a une idée de la forme de la (des) solution (s), on peut biaiser cette initialisation en regroupant les individus autour de la région considérée, ce qui est susceptible d'accélérer la convergence de l'algorithme.

1.3.3. Evaluation:

La fonction d'adéquation d'un individu est calculée par la fonction d'évaluation; elle mesure son adaptation à un environnement donné. Le but de l'AG est d'optimiser (maximiser ou minimiser) cette fonction. Pour résoudre un problème d'optimisation, on prend naturellement la fonction de coût comme fonction d'évaluation, mais dans le cas d'un problème de minimisation il faut évidemment considérer l'opposé de la fonction de coût. Seule la fonction de coût est utilisée par un AG lors de l'évaluation, et non pas ses dérivées successives comme dans des méthodes de calculs classiques du type méthode de gradient, ce qui confère aux AG une très bonne robustesse (aucune hypothèse n'est requise sur la fonction de coût).

1.3.4. Sélection:

La sélection est le processus selon lequel on effectue une transposition informatique de la sélection naturelle. La fonction sélective joue le rôle de l'environnement (prédateur, maladies,....). La sélection est le premier arbitre décidant de la vie et de la mort des individus, c'est pourquoi elle est un élément primordial du bon fonctionnement d'un AG. On utilise parfois le terme reproduction à la place de sélection.

• Mise en œuvre possible de la sélection proportionnelle :

Le mécanisme de sélection est constitué de deux phases. Dans la première phase, on assigne à chaque individu un nombre réel qui

représente le nombre espéré d'occurrences de cet individu dans la génération suivante. Dans la seconde phase, on crée un nombre entier d'occurrences de chaque individu, ce nombre étant calculé à partir des nombres réels de la première phase. La deuxième phase doit être conçue avec soin car elle introduit nécessairement des effets stochastiques en convertissant un nombre espéré réel en un nombre effectif entier.

• Roulette proportionnelle :

Pour chaque individu, on reproduit sur un disque de périmètre unité un arc de cercle proportionnel à la fonction d'adéquation, puis on fait tourner la roulette aléatoirement i. e on tire au hasard un nombre dans $[0\ 1]$ et on reproduit l'individu sélectionné, on recommence n fois pour obtenir la nouvelle population. Le nombre de descendants d'un individu est ainsi statistiquement proportionnel à la fonction d'adéquation comme l'illustre la figure suivante :

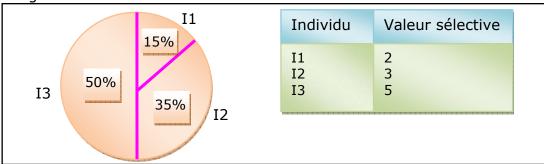


Figure 0:2: Roulette proportionnel.

Il peut être intéressant de garder intact le meilleur individu de la population lors du passage d'une génération à la suivante, la stratégie élitiste concrétise cette idée en remplaçant le meilleur individu d'une population dans la population suivante et ce, sans aucune altération (mutation), cet individu reste toutefois candidat pour la phase de reproduction habituelle.

Si tous les individus de la population ont une valeur sélective identique, leur sélection est équiprobable. La probabilité de sélection d'un individu est alors 1/M. Lorsque les valeurs sélectives des individus sont différentes, il faut modifier les probabilités de sélection de manière à favoriser les individus de plus forte valeur sélective. Pour cela, en notant f_i la valeur sélective de l'individui, on fixe la probabilité de sélection Ps_i à :

à:
$$Ps_i = \frac{f_i}{\sum\limits_{j=1}^{j=\lambda} f_j} (II:1)$$

Le mécanisme de sélection peut être appliqué dans une forme préservatrice ou dans une forme extinctrice. Dans le premier cas, chaque individu se voit attribuer une valeur sélective non nulle de manière à ce que sa sélection soit possible. Au contraire, dans le second cas, on empêche la sélection de certains individus en leur assignant une valeur sélective nulle.

Pour mettre en ouvre la forme extinctrice, les M individus sont triés par ordre décroissant de valeur sélective. Soit μ le nombre d'individus dont

on souhaite tenir compte pour la sélection (on décide par exemple de tenir compte de 60% des individus, les meilleurs). Soit A_i^t l'individu i de la génération t. Soit f la fonction sélective, la probabilité de sélection de cet individu, soit $Ps(A_i^t)$, est donnée par :

$$Ps_{i} = \begin{cases} 0 & pour \quad i \in [1 + \mu, \lambda] \\ \frac{f(A_{i}^{t})}{\sum_{j=1}^{j=\mu} f(A_{j}^{t})} & pour \quad i \in [1, \mu] \end{cases}$$
 (II : 2)

Dans sa forme la plus répandue, la sélection est préservatrice, i.e. $\mu = \lambda$

• Croissance des individus :

Nous nous intéressons à la forme préservatrice, la plus courante. Soit m_i^t le nombre d'occurrences de l'individui, de valeur sélective f_i , à la génération t. On s'intéresse au nombre espéré de ses occurrences à la génération (t+1), sous l'action d'une sélection proportionnelle (M tirages d'un nombre i compris entre 1 et M ont été effectués), la probabilité d'obtenir i étant $Ps(A_i^t)$

$$m_i^{t+1} = m_i^t \cdot \lambda \cdot Ps(A_i^t)$$
 (II:3)

$$m_i^{t+1} = m_i^t \cdot \lambda \cdot \frac{f(A_i^t)}{\sum_{j=1}^{j=\lambda} f(A_j^t)} Ps(A_i^t) \dots$$
 (II:4)

Or:
$$\frac{\sum_{j=1}^{j=\lambda} f(A_j^t)}{\lambda} = \overline{f^t}$$
 (II:5)

Est la valeur sélective moyenne des M individus de la génération t. On a donc :

$$m_i^{t+1} = m_i^t \lambda \cdot \frac{f(A_i^t)}{\overline{f^t}} Ps(A_i^t) \dots$$
 (II:6)

Par conséquent, seuls les individus dont la valeur sélective est supérieure à la moyenne, voient leur nombre d'occurrences augmenter dans la génération suivante. Ce résultat n'est valable que pour une sélection préservatrice.

• Choix de la fonction sélective :

La fonction sélective permet de calculer les valeurs sélectives des individus de la population. Elle doit rendre des résultats positifs ou nuls pour que l'opérateur de sélection fonctionne convenablement. Le problème à résoudre ayant été traduit sous la forme d'une fonction objective, la fonction sélective f est formée par la composition de deux fonctions :

$$f(x) = g(o(x))$$
.....(II:7)

Où o représente la fonction objectif et g transforme la valeur o(x) en une valeur positive.

1.3.5. Croisement:

Le croisement est un opérateur de recombinaison qui fournit un couple d'enfants à partir d'un couple de parents de la génération précédente. Le problème à résoudre consiste à choisir deux individus parmi ceux qui ont été sélectionnés (les parents) pour en créer deux autres (les enfants ou descendants). Ce choix s'appelle l'appariement. HOLLSTIEN a expérimenté diverses méthodes d'appariement parmi lesquelles :

- L'appariement aléatoire ;
- La consanguinité;
- Le croisement entre lignées ou inter fécondation.

Le croisement combine deux individus appariés avec une probabilité Pc. La version classique de cet opérateur est appelée le croisement en un point ou croisement simple [15]. Il consiste à tirer au sort une valeur (position de croisement) comprise entre 1 et l-1 (l étant la taille d'un chromosome) et à générer deux individus en inversant les gènes des deux parents dont le lieu est compris entre le site de croisement +1 et l, (Figure (II : 3)). Les valeurs de Pc sont classiquement comprises entre 0.6 et 0.9. Cet opérateur permet de générer deux individus nouveaux à partir de deux individus ayant jusqu'alors survécu, donc dont la valeur sélective est bonne.

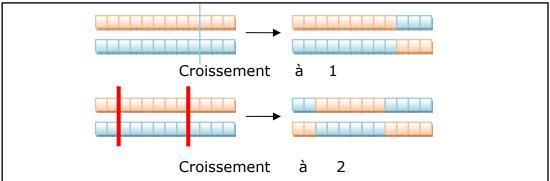


Figure 0:3: 2 Exemples de croissement.

Le croisement des nombres réels se fait très fréquemment sur la base du modèle barycentrique, avec α choisi aléatoirement entre -0.5 et 1.5.

$$X_{1}^{'} = \alpha X_{1} + (1 - \alpha) X_{2}$$
 (II: 8)

$$X_{2}' = \alpha X_{2} + (1 - \alpha)X_{1}$$
 (II: 9)

Avec X_1 et X_2 les parents et X_1 et X_2 les fils. Afin de ne plus s'éloigner des parents, le croisement s'effectue sur une partie des chromosomes.

1.3.6. Mutation:

La mutation consiste à modifier, avec une probabilité Pm, faible, les valeurs des gènes des chromosomes. Lorsque l'alphabet est binaire, la mutation consiste à inverser la valeur du gène (0 devient 1 et 1 devient 0, voir figure suivante :

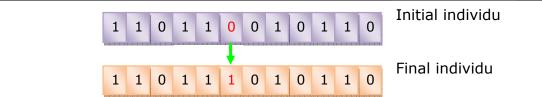


Figure 0:4: Exemple d'opérateur de mutation.

Cet opérateur permet un déplacement aléatoire dans l'espace des solutions autorisant ainsi l'exploration de régions dans lesquelles se trouve peut-être un point intéressant. Sans la mutation, il y a un risque de convergence prématurée de l'algorithme vers un extrémum local.

1.3.7. Recuit Simulé:

Le recuit simulé ne fait pas à proprement parler partie des opérateurs de croisement mais il opère après que le croisement se soit effectué, on dispose alors de deux chromosomes parents et de deux chromosomes fils. Nous allons déterminer quels seront les deux descendants qui vont être effectivement conservés lors d'un tournoi. Si les chromosomes fils sont meilleurs que leurs parents, c'est à dire si leurs adéquations est plus élevée alors ils seront conservés, s'ils sont moins bons, ils gardent néanmoins une chance d'être conservés. On effectue un tirage aléatoire et on conserve ces chromosomes avec une probabilité d'admission P(T) qui décroît au fur et à mesure que le nombre de générations augmente. [16].

[16].
$$P(t) = \frac{1}{1 - e^{\left(\frac{-|E_i - E_f|}{T}\right)}}$$
 (II: 10)

 E_i L'adéquation du parent et E_f l'adéquation de l'enfant.

Plus la différence entre E_i et E_f est grande plus la probabilité P(T) de garder le parent est grand. En fonction des générations successives, T va diminuer, donc la probabilité de garder les parents va augmenter. Donc au début, si la différence de l'adéquation entre le parent et l'enfant n'est pas trop élevée, la probabilité de garder les enfants est élevée, ce qui favorise l'exploration.

En fin d'exécution de l'algorithme la probabilité de garder les meilleurs individus, c'est à dire les parents, est dominante.

T Va donc décroître en fonction du nombre k de générations. La formule suivante est souvent utilisée, avec à l'initialisation $T(0) = T_i$:

Le recuit simulé nous permet donc de conserver une disparité au sein de la population. Le but étant de chercher à conserver les chromosomes ayant la meilleure adéquation autant que possible mais en gardant une chance de permettre aux chromosomes moins bien adaptés de survivre malgré tout pendant un certain temps. Cependant les possibilités de mouvement se réduisent quand nous nous rapprochons de la fin de l'exécution. En effet les déplacements importants autorisés en début d'exécution ne sont plus intéressants car en fin d'algorithme nous sommes proches d'une solution optimale, et il n'est pas bon de permettre à l'adéquation du chromosome de chuter. Voilà pourquoi la probabilité d'admission d'un chromosome d'adéquation inférieure à celle de ses parents décroît en fonction des générations.

2. La logique floue:

2.1. Introduction:

La majorité des systèmes industriels complexes sont difficiles à contrôler. Cette difficulté découle généralement de la complexité du système dynamique dont il faut assurer la conduite non linéaire, objectifs et contraintes de commande multiples, erreurs de modélisation, bruit de mesures, perturbations...etc.

Pour faire face à ces difficultés, de nouvelles stratégies de commande, basées sur l'expertise de l'opérateur humain ont été élaborées. Le point clef de ces stratégies réside dans l'utilisation du calcul symbolique pour l'élaboration de l'algorithme de commande. Cette méthodologie ouvre une nouvelle voie pour l'utilisation d'heuristiques en remplaçant la modélisation du régulateur par une loi de commande basée sur des connaissances linquistiques acquises d'un opérateur du procédé [17].

Bien que des tentatives aient été menées depuis très longtemps pour la formalisation des connaissances subjectives, le véritable mérité de la réussite de cette nouvelle stratégie revient à Lotfi Zadeh [18] qui a exposé pour la première fois à l'université de Berkeley (Californie) en 1965 ce qui est connu de nos jours sous le nom de logique floue.

2.2. Notions fondamentales de la logique floue :

2.2.1. Ensembles flous:

La logique floue et ses applications sont basées sur la notion d'ensembles flous. Dans la théorie classique des ensembles, un ensemble est caractérisé par les éléments qui lui appartiennent. Pour chaque élément, il n'existe que deux états : l'élément appartient à l'ensemble ou il n'appartient pas. La théorie des ensembles flous, par contre, permet une appartenance partielle d'un élément à l'ensemble.

Cette notion d'appartenance partielle d'un élément à un ensemble donné est pondérée par un coefficient dit degré d'appartenance. De ce fait, chaque ensemble est caractérisé par une fonction $\mu_F(x)$, dite " fonction d'appartenance".

Chaque ensemble flou peut être noté comme suit :

$$F = \{(x, \mu_F(x)) / x \in X\}...$$
 (II: 12)

 $\mu_{\scriptscriptstyle F}(x)$: Fonction d'appartenance ;

X : Ensemble de référence ;

x : Valeur prise par une variable floue donnée.

L'ensemble flou F est définie par l'ensemble des couples (x, $\mu_F(x)$). Donc

Parmi les fonctions d'appartenance les plus utilisées, on cite [19] : les fonctions triangulaires, trapézoïdales, gaussiennes ...etc.

La figure (II: 5) montre l'allure de ces fonctions

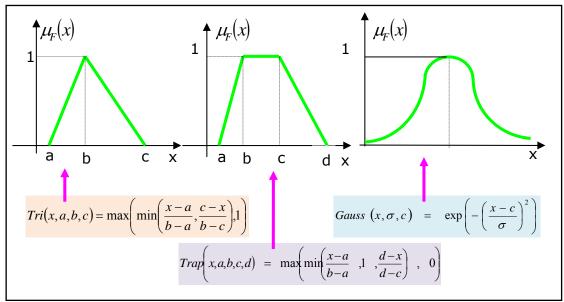


Figure 0:5: Exemple de fonctions d'appartenance.

2.2.2. Opérations sur les ensembles flous :

Afin de manipuler les ensembles flous, il est nécessaire de définir certaines opérations élémentaires.

Soit, les nombres réels : a, b, c et $d \in [0 \ 1]$.

• <u>Définition de norme triangulaire (T-norme) [20] :</u>

La norme triangulaire de a et b, notée $a\otimes b$, dénote une classe de fonctions binaires qui peuvent représenter l'opération d'intersection entre ensembles. Cette opération est commutative, associative, l'élément neutre est 1.

• <u>Définition de norme co-triangulaire (T-conorme ou S-norme)</u>:

La conorme triangulaire de a et b, notée $a \oplus b$, dénote une classe de fonctions binaires qui peuvent représenter l'opération d'union entre ensembles. Cette opération est commutative, associative, l'élément neutre est le zéro.

Soit, A et B deux ensembles dans X caractérisés respectivement par les deux fonctions d'appartenance $\mu_A(x)$ et $\mu_B(x)$. L'union, l'intersection et la complémentation des ensembles flous sont définies à l'aide de leurs fonctions d'appartenance [21].

• Union :

La fonction d'appartenance $\mu_{A \cup B}$ de l'union $A \cup B$ est définie, pour tout $x \in X$, par : $\mu_{A \cup B}(x) = \max \mu_A(x), \mu_B(x)$ }.....(II : 14)

• Intersection:

La fonction d'appartenance $\mu_{A\cap B}$ de l'intersection $A\cap B$ est définie, pour tout $x{\in}X$, par :

$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x)\}.$$
 (II: 15)

• Complémentation :

La fonction d'appartenance $\mu_{\overline{A}}$ du complément \overline{A} d'un ensemble flou A est définie pour tout $x \in X$, par :

$$\mu_{\overline{A}}(x) = 1 - \mu_A(x)$$
..... (II: 16)

2.2.3. Relations floues:

Soient U_1 , U_2 , ..., U_n des univers de discours. Une relation floue est un ensemble flou dans $U_1*U_2*...*U_n$ exprimé comme suit :

$$R_{U_1^*...^*U_n} = \{ ((x_1, ..., x_n), \mu_R(x_1, ..., x_n)) / (x_1, ..., x_n) \in U_1^*...^*U_n \}$$
 (II: 17)

2.2.4. Composition Sup-Star:

Si R et S sont deux relations floues dans respectivement U^*V et V^*W , la composition de ces deux relations est une relation floue dénotée par $R \circ S$ et elle définie par :

$$R \circ S = \{ [(u, v), \sup(\mu_R(u, v) \otimes \mu_S(v, w))], u \in U, v \in V, w \in W \}......$$
 (II: 18)

2.3. Représentation et raisonnement linguistique :

Dans le but de représenter les expressions linguistiques ou des mots du langage naturel à l'aide de la logique floue, on introduit une représentation symbolique des différentes notions de ses expressions. L'association d'ensembles et de relations floues à la représentation symbolique permettra de quantifier les expressions.

2.3.1. Variables linguistiques:

L'utilisation des ensembles flous constitue une voie privilégiée pour la manipulation de concepts vagues et imprécis : petit, grand, chaud...etc. En particulier, on peut employer les ensembles flous pour représenter des variables linguistiques. Une variable linguistique peut être vue comme une variable dont la valeur est un nombre flou.

Une variable linguistique est déterminée par quatre caractéristiques (X,LX,U_X,M_X) où :

X: Nom de la variable (vitesse par exemple);

LX: Ensemble des valeurs linguistiques de X par exemple : $L(vitesse) = \{rapide, moyenne, lente...\}$

 U_X : Univers de discours de X

Mx: Règle sémantique associé à chaque valeur

2.3.2. Proposition floue:

Une proposition floue est une description de la variable floue, donnée sous la forme suivante : " XestA "

2.3.3. Conjonction et disjonction des propositions floues :

La conjonction des propositions $P_1, P_2, ..., P_n$ définie par $P_1 : "XestF_1 ", ..., P_n$: " $XestF_n$ " est la proposition composée P_1 telle que :

 $P: "XestF_1 \text{ Et } XestF_2 \text{ et ... } XestF_n".$

La disjonction des propositions $P_1, P_2, ..., P_n$ définie par $P_1 : "XestF_1 : ", ..., P_n$:

" $XestF_n$ " est la proposition composée P telle que :

 $P: "XestF_1 Ou XestF_2 ou... XestF_n".$

2.3.4. Raisonnement flou:

En général, plusieurs valeurs de variables linguistiques convenablement définies par des fonctions d'appartenances sont liées entre elles par des règles afin de tirer des conclusions.

Ces règles utilisent souvent les connecteurs SI-ALORS. La procédure utilisée pour déduire des conclusions à partir d'un ensemble de règles et de conditions, représentés par des ensembles et relations floues, est appelée raisonnement flou [22] [19].

Dans le raisonnement flou intervient trois niveaux fondamentaux :

- Implication floue;
- Inférence floue ;
- Agrégation floue.

• Implication floue [23] [24] :

En logique classique, une relation d'implication possède plusieurs représentations dont les résultats sont identiques. Considérons les deux variables logiques binaires R et R. La relation d'implication $R \Rightarrow R$ est équivalente aux deux relations $\overline{R} \vee P_2$ ou $(R \wedge P_2) \vee \overline{R}$, avec $(\wedge, \vee, \overline{})$, désignent respectivement, les opérateurs ou, et, complément de la logique classique.

Sur la base de ces deux relations, nous avons introduit plusieurs

définitions de l'implication floue, parmi lesquelles nous citons :

Norme	Valeur de vérité $\mu_{F\to B}(x_1x_n,y)$
KLANE-DIENES	$\max \left(1-\mu_F(x_1x_n),\mu_B(y)\right)$
LUKASIEWICZ	$\min(1,1-\mu_F(x_1x_n) + \mu_B(y))$
ZADEH	$\max\left(\min(\mu_F(x_1x_n),\mu_B(y)),1-\mu_F(x_1x_n)\right)$
MAMDANI	$\min\left(\mu_F(x_1x_n),\mu_B(y)\right)$
LARSEN	$\mu_F(x_1x_n) \times \mu_B(y)$

Table 0:1: Les implications les plus utilisées.

• Inférence Floue :

La procédure utilisée pour déduire des conclusions à partir d'un ensemble de règles floues et d'une ou de plusieurs conditions est appelée Inférence floue (raisonnement flou, ou encore raisonnement approximatif) [19]. En logique floue, il y a deux importantes règles d'inférence : Le Modus Tollens Généralisé "MTG", et le Modus Ponens Généralisé "MPG" proposé

par Zadeh [25].

	MTG	MPG
Condition	Y est B'	X est F''
Règle floue	Si X est F Alors Y est B	Si X est F Alors Y est B
Ensemble de résultat	X est F'	Y est B [']

En commande à base de logique floue, seul le MPG est utilisé afin de respecter le principe de causalité, alors que dans les systèmes experts, on peut utiliser les deux modes.

Dans MPG, l'ensemble inféré B' est le résultat de l'inférence (ou la composition) de l'ensemble flou F' et de la relation de l'implication floue, notée R. On aura :

notée R. On aura :
$$B' = F' \circ R = \int_{y \in U, x \in U_x} \sup \left(\mu_F(x) \otimes \mu_R(x, y) \right) / y \dots$$
 (II : 19)

• Agrégation des règles :

Lorsque la base de connaissance comporte plusieurs règles liées par un opérateur "ou", l'ensemble flou inféré B' est obtenu après une opération appelée agrégation des règles. En d'autres termes, l'agrégation des règles consiste à considérer des ensembles flous inférés par chaque règle pour en créer un autre, représentatif de l'ensemble des règles.

2.3.5. Mécanisme d'inférence :

Les mécanismes les plus utilisés en commande à base de logique floue, sont [26] :

- Méthode d'inférence somme-produit "Som-pro";
- Méthode d'inférence maximum-produit "Max-pro";
- Méthode d'inférence maximum-minimum "Max-min".

• Méthode d'inférence Som-prod :

Cette méthode réalise, au niveau de la condition, l'opérateur OU par la formation de la somme (la valeur moyenne), tandis que l'opérateur ET est réalisé par la formation de la produit. La conclusion de chaque règle, précédée par ALORS, est réalisée par la formation du produit [27]. L'expression de l'ensemble inféré s'écrit :

$$\mu_{B}(y) = \sum_{l=1}^{K} \sup \left[\mu_{F_{l}^{i}}(x_{1}) \times ... \times \mu_{F_{l}^{i}}(x_{1}) \right] \times \left[\left(\mu_{F_{l}^{i}}(x_{1}) \times ... \times \mu_{F_{l}^{i}}(x_{1}) \right) \times \mu_{B^{(l)}}(y) \right] / K \cdot (II : 20)$$

Dans le cas de fonctions singletons :

$$F_{i}'(x) = \left\{x_{i}^{*}\right\} = \begin{cases} 1 & si & x = x_{i}^{*} \\ 0 & si & x \neq x_{i}^{*} \end{cases} \qquad i = 1, ..., n \dots$$
 (II: 21)

On obtient :
$$\mu_{B^{(l)}}(y) = \frac{1}{K} \sum_{l=1}^{K} \mu_{B^{(l)}}(y) \times \left[\prod_{i=1}^{n} \mu_{F_{i}^{(l)}}(x_{i}^{*}) \right]$$
 (II : 22)

2.4. Contrôle par la logique floue :

La majeure application dans la pratique est la régulation par un régulateur flou qui fait apparaître trois grands modules : module de fuzzification, module de règles et d'inférence et module de défuzzification. Le module de fuzzification permet de passer de variables réelles à des variables floues. Le module d'inférence permet la déduction de la valeur floue de la sortie à partir des règles floues prédéfinies. Enfin il y a le module de défuzzification qui sert à rendre la valeur floue de sortie une valeur numérique.

Les trois étapes sont schématisées dans la figure suivante [28] :

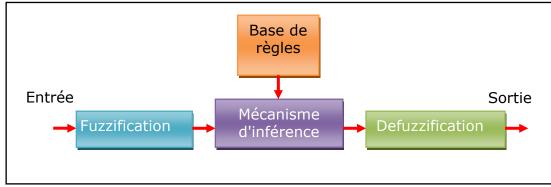


Figure 0:6: Schéma block d'un contrôleur floue.

2.4.1. Fuzzification:

Comme les opérateurs utilisés dans la commande floue agissent sur des sous ensembles flous, il est nécessaire de transformer les valeurs réelles des variables d'entrée provenant du monde extérieur en des ensembles flous. Pour cela, le module de fuzzification sert à associer à une mesure x_{0i} d'une variable d'entrée x_i une fonction d'appartenance particulière $\mu_{x_i}(x_0)$. Les variables d'entrée $X = [x_1, x_2, ..., x_n]$, avant de les faire entrer, sont réduites à des grandeurs normalisées X_n dans une plage de variation, souvent [-1, 1] appelée univers de discours, qui

plage de variation, souvent [-1, 1] appelée univers de discours, qui peut-être soit discret, soit continu.

Dans le cas d'un univers de discours continu, les valeurs linguistiques d'une variable d'entrée (Exemple : faible, moyen, élevé) sont représentées par des fonctions d'appartenance dont le nombre peut varier d'une variable d'entrée à une autre et leur valeur est dans l'intervalle [0, 1].

Un exemple de fuzzification continu est illustré dans la figure suivante avec des fonctions d'appartenance gaussiennes :

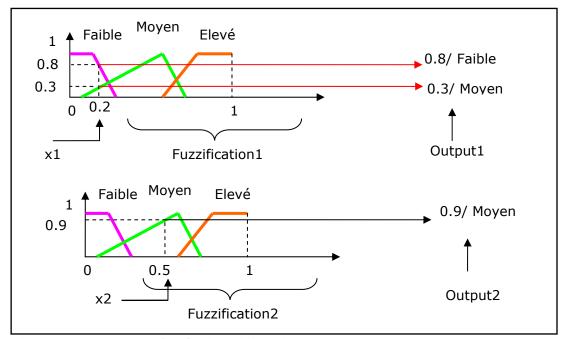


Figure 0:7: Exemple de fuzzification.

2.4.2. Règles et mécanisme d'inférence :

Le mécanisme d'inférence sert à relier les variables d'entrée, qui sont transformées en variables linguistiques à l'aide de la fuzzification, aux variables de sortie qui sont également exprimées comme variables linguistiques. Cette liaison est réalisée à l'aide des règles dite d'inférence prédéfinies par l'expert ayant la forme (Si....Et..., Alors).

Souvent, les règles d'inférence sont exprimées sous la forme d'une matrice comme l'illustre l'exemple suivant :

x1	x2	Faible	Moyen	Elevé
Faible		Faible	Faible	Elevé
Moyen		Faible	Moyen	Elevé
Elevé		Faible	Elevé	Elevé

Table 0:2: Exemple d'une matrice d'inférence.

Suivant l'exemple précédent, l'entrée du module d'inférence dans ce cas est ((Faible, Moyen); Moyen) et à partir de la matrice d'inférence, la sortie est (Faible; Moyen) calculé comme suit :

- x1 est Faible et x2 est Moyen, alors : la sortie est : Faible ;
- x1 est Moyen et x2 est Moyen, alors : la sortie est : Moyen.

2.4.3. Défuzzification:

C'est l'opération totalement inverse de la fuzzification qui sert à transformer une variable, (la sortie) ayant une valeur linguistique, en une valeur numérique, pour être exploitée. La sortie du module d'inférence, qui est la sortie calculée par raisonnement flou a une valeur linguistique (Faible; Moyen) qui n'est pas exploitable comme sortie, donc sa transformation en une valeur numérique est nécessaire en utilisant, par exemple:

La méthode du centre de gravité qui est la méthode la plus utilisée sert à déterminer le centre de gravité de la fonction d'appartenance résultante. La méthode du maximum qui consiste à prendre la moyenne des abscisses correspondant à la valeur maximale de la fonction d'appartenance résultante comme valeur de sortie du régulateur. Dans le cas où il y a plusieurs abscisses correspondantes, on prend alors leur moyenne.

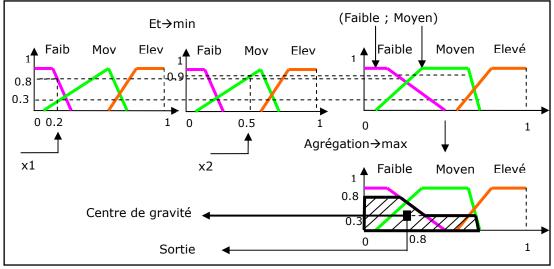


Figure 0:8: Defuzzification par centre de gravité.

3. Les réseaux de neurones artificiels (RNAs) :

3.1. Introduction:

Les RNAs ou conventionnels sont des modèles mathématiques et informatiques, des assemblages d'unités de calculs appelés neurones formels, et dont l'inspiration originelle était un modèle de la cellule nerveuse humaine. Cet héritage de la neurobiologie forme une composante importante de la matière, et le souci de maintenir une certaine correspondance avec le système nerveux humain a animé et continue à animer une part importante des recherches dans le domaine. Malgré cet héritage, l'essentiel des travaux d'aujourd'hui ont pour objet les RNA formels et non son corrélat neurobiologique. Vu comme des systèmes de calcul, les RNAs possèdent plusieurs propriétés qui les rendent intéressants d'un point de vue théorique, et fort utiles en pratique [29].

3.2. Les neurones biologiques :

Les cellules nerveuses, appelées neurones, sont les éléments de base du système nerveux central. Celui-ci en posséderait environ cent milliards. Les neurones possèdent de nombreux points communs dans leur organisation générale et leur système biochimique avec les autres cellules, ils présentent cependant des caractéristiques qui leur sont propres et se retrouvent au niveau des cinq fonctions spécialisées qu'ils assurent :

- Recevoir des signaux en provenance de neurones voisins ;
- Intégrer ces signaux ;
- Engendrer un influx nerveux ;
- Le conduire ;
- Le transmettre à un autre neurone capable de le recevoir.

3.3. Structure des neurones :

Un neurone est constitué de trois parties :

- Le corps cellulaire ;
- Les dendrites ;
- L'axone.

3.3.1. Le corps cellulaire :

Il contient le noyau du neurone et effectue les transformations biochimiques nécessaire à la synthèse des enzymes et des autres molécules qui assurent la vie du neurone, il a une forme pyramidale ou sphérique dans la plupart des cas.

3.3.2. Les dendrites :

Chaque neurone possède une « chevelure » de dendrites Celles-ci sont de fines extensions tubulaires, de quelques dixièmes de microns. De diamètre et d'une longueur de quelques dizaines de microns. Elles sont les récepteurs principaux du neurone pour capter les signaux qui lui parviennent

3.3.3.L'axone:

L'axone, qui est à proprement parler la fibre nerveuse, sert de moyen de transport pour les signaux émis par le neurone. Pour former le système nerveux, les neurones sont connectés les uns aux autres suivants des répartitions spatiales complexes, les connexions entre deux neurones se font en des endroits appelés synapses où ils sont séparés par un petit espace synaptique de l'ordre d'un centième de micron. D'une façon simple, on peut dire que le soma du neurone traite les courants électriques qui lui proviennent de ses dendrites, et qu'il transmet le courant électrique (sous forme d'impulsions), chacune de durée d'environ 1ms et une amplitude d'environ 100mv, résultant de ce traitement aux neurones aux quels il est connecté par l'intermédiaire de son axone.

Le schéma classique présenté par les biologistes est celui d'un soma effectuant une sommation des influx nerveux transmis par ses dendrites. Si la sommation dépasse un certain seuil, le neurone répond par un influx nerveux au potentiel d'action qui se propage le long de son axone. Si la sommation est inférieure à ce seuil, le neurone reste inactif [30].

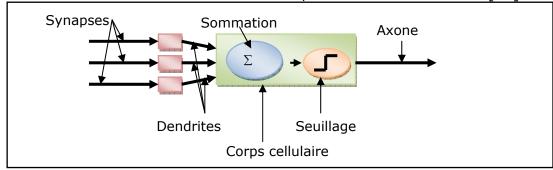


Figure 0:9: Le modèle d'un neurone biologique.

3.4. Neurone formel:

McCulloch et Pitts ont proposé une unité à seuil binaire comme un modèle neuronal. Le diagramme de McCulloch-Pitts est montré en figure (II : 10).

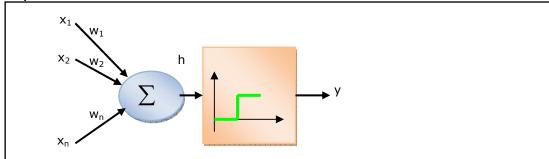


Figure 0:10: Le modèle de McCulloch-Pitts.

Ce modèle mathématique calcule la somme pondérée de ses n entrées, xj, j=1,2,...,n, Le nœud est caractérisé par un seuil [31]. Il génère une sortie de « 1 », si cette somme et supérieure au seuil μ , sinon une sortie de « 0 ». Mathématiquement :

$$v = \begin{cases} 1 & si & \sum_{i=1}^{n} W_{i} \chi_{i} \geq \mu \\ 0 & si & \sum_{i=1}^{n} W_{i} \chi_{i} < \mu \end{cases}$$
(II: 23)

Où $\Phi(.)$ est la fonction d'activation, et w_j est le poids synaptique associé à la j-ème sortie. Parfois, on considère le seuil μ comme un autre poids $w_0 = \mu$ et qui est connecté à un neurone avec une entrée constante, $x_0 = 1$.

Les poids positifs correspondent à des synapses excitateurs, tandis que les poids négatifs correspondent aux synapses inhibitrices. Il y a une analogie évidente (Table II : 3) avec les neurones biologiques :

Neurones biologiques	Neurones artificiels
Synapses	Connections pondérées
Axons	Sorties
Dendrites	Entrées
Sommateur	Fonction d'activation

Table 0:3: L'analogie entre les neurones biologiques et les neurones artificiels

3.5. L'utilisation des RNA :

Les RNA ont apporté une avance considérable et des solutions fructueuses, et de ce succès, les RNA sont devenus maintenant un vrai outil de résolution de problèmes, dont on va citer quelques uns :

3.5.1. Classification:

La tâche de classification est d'assigner un modèle d'entrée (Exemple : un signal vocal ou des caractères manuscrits) représenté par un vecteur caractérisant une des classes pré spécifiées. Des applications bien connues de la classification sont la reconnaissance de caractères, la reconnaissance de la parole, la classification des signaux EEG, la classification des cellules sanguines et l'inspection des circuits imprimés.

3.5.2. Catégorisation:

Dans la catégorisation connue aussi comme la classification non supervisée, il n'y a pas de modèles d'apprentissage bien connus. L'algorithme de catégorisation explore et teste la similarité entre les différents modèles qui lui sont présentés, et place les modèles similaires dans une même catégorie (cluster).

3.5.3. Approximation de fonctions :

Plusieurs problèmes scientifiques et d'ingénieries requièrent l'approximation. Ayant un ensemble de N paires d'apprentissage (entréesortie), générées par une fonction inconnue M(x), la tâche d'approximation de fonction est de trouver une estimation, disons f, de la fonction inconnue M.

3.5.4. Prédiction:

Ayant un ensemble de N échantillons dans une séquence temporelle déterminée, $\{y(t_1), y(t_1), ..., y(t_N)/t_1 < t_2 < ... < t_N\}$, le but est de déterminer la valeur de $y(t_{N+1})$ à l'instant futur t_{N+1} . La prédiction est très utilisée dans la bourse, le contrôle des procédés, les prévisions météorologiques.

3.5.5. Optimisation:

Un problème d'optimisation peut généralement comprendre les éléments suivants :

- Un ensemble de variables indépendantes qui font référence à l'état du processus;
- Une fonction objective (fonction de coût/erreur) qui doit être optimisée;
- Un ensemble de contraintes si elles existent bien sûr.

Le but de l'optimisation est de trouver un état qui satisfait ces contraintes de telle façon que la fonction objectif soit optimisée.

3.5.6. Mémoire associative :

La mémoire associative, peut être accédée par l'intermédiaire de sa contenance, et les données de cette mémoire peuvent être reconnues par cette dernière même si ils sont bruités ou seulement une partie d'un modèle d'entrée est disponible. Exemple : la récupération bibliographique des références d'une revue de l'information partielle, de ce fait la mémoire associative est très appréciée dans la construction des bases de donnés multimédia.

3.5.7. Contrôle:

Considérons un système définie par la paire $\{u(t), y(t)\}$ où u(t) est la commande du système et y(t) est la sortie résultante à l'instant t, (dans le contrôle adaptatif), le but est de générer une entrée de commande u(t) de façon que le système suive la sortie désirée déterminée par le modèle de référence [32].

3.6. Apprentissage:

La capacité à apprendre un trait fondamental de l'intelligence, d'un autre côté, qui veut dire l'apprentissage est parfois difficile à décrire. Le processus d'apprentissage, dans les contextes des RNA peut être vu comme un problème ajustement et d'adaptation de l'architecture et des poids des RNA de façon qu'ils exécutent efficacement une tache spécifique. Au lieu de spécifier un ensemble de règles, les RNA apprennent automatiquement à partir de la collection d'exemples, et c'est ce qui les rend très attractifs.

Il y a trois types de paradigmes d'apprentissage :

- Supervisé ;
- Non supervisé ;
- Hybride.

3.6.1. L'apprentissage supervisé :

C'est actuellement le mode d'apprentissage le plus couramment utilisé. Son principe est élémentaire : on soumet le réseau à un grand nombre d'exemples pour lesquels l'entrée et la sortie associées sont connues et les poids sont modifiés de façon à corriger l'erreur commise par le réseau. Le plus répandu des algorithmes d'apprentissage supervisé est l'algorithme de rétropropagation du gradient d'erreur qui est appliqué aux réseaux multicouches.

3.6.2. L'apprentissage nonsupervisé :

Contrairement aux modes supervisés, seule une base d'entrées est fournie au réseau. Celui-ci doit déterminer lui-même ses sorties en fonction des similarités détectées entre les différentes entrées, c'est-à-dire en fonction d'une règle d'auto-organisation. Il explore la structure des données, ou encore la corrélation entre les modèles d'entrée et les organise à partir de cette corrélation en des catégories.

3.6.3. L'apprentissage hybride:

Il combine l'apprentissage supervisé et nonsupervisé. Dans de nombreux problèmes on ne dispose pas de l'information nécessaire à la construction d'une base d'apprentissage complète. On ne dispose souvent que d'une information qualitative permettant l'évaluation de la réponse calculée, sans pour autant connaître la réponse la plus adaptée.

3.7. Quelques topologies des RNA:

Quelques types de RNA sont montrés en figure (II: 11)

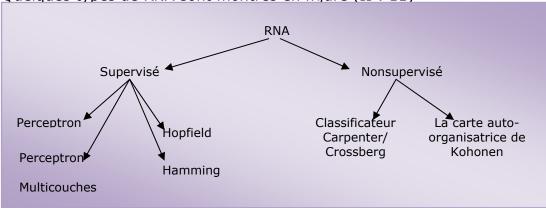


Figure 0:11: Quelques types de RNAs.

3.7.1. Le réseau de Hopfield :

Le réseau de Hopfield est généralement employé avec des entrées binaires. Ces réseaux sont les plus appropriés lorsque des représentations binaires exactes sont possibles (Exemple : des images en noir et blanc où les éléments d'entrée sont des valeurs des pixels, ou avec un texte en code ASCII où chaque caractère peut être représenté par des bits). Les fonctions de transfert sont des fonctions seuil, les entrées sont de type binaire et les sorties peuvent prendre les valeurs +1 et -1. La sortie de chaque neurone est connectée à tous les autres nœuds par l'intermédiaire de facteurs de pondération (poids).

3.7.2. Le réseau de neurones de Hamming :

Le réseau de Hopfield est souvent testé sur des problèmes où les entrées sont produites en choisissant un exemplaire en inversant les bits d'une manière aléatoire et indépendante avec une probabilité donnée. C'est un problème classique dans la théorie de communications qui se produit quand des signaux binaires à longueur constante sont envoyés par un canal binaire symétrique sans mémoire. Le classificateur optimal d'erreur minimale dans ce cas calcule la distance de Hamming de l'exemplaire pour chaque classe et choisit cette classe en fonction de la distance minimum de Hamming.

3.7.3. Le Classificateur de Carpenter et Grossberg :

Carpenter et Grossberg, dans le développement de leur théorie de résonance adaptative ont conçu un RNA de classification avec un apprentissage nonsupervisé. Ce réseau met en application un algorithme de catégorisation qui est très semblable à l'algorithme séquentiel simple de chef. L'algorithme de chef choisit la première entrée comme exemplaire pour le premier faisceau. La prochaine entrée est comparée au premier exemplaire de faisceau. Elle suit "le chef" et groupée avec le premier si la distance au premier est inférieure au seuil. Autrement, c'est l'exemplaire pour un nouveau faisceau. Ce processus est répété pour toutes les entrées suivantes.

3.7.4. Le perceptron multicouche :

Les perceptrons multicouches sont des réseaux non-récurrents avec une ou plusieurs couches de neurones entre les couches d'entrée et de sortie. Ces couches additionnelles contiennent des unités cachées ou des neurones qui ne sont pas directement reliés aux neurones des couches d'entrée et de sortie. Un perceptron avec trois couches dont deux couches cachées est montré en figure (II : 12). Ils sont de plus en plus employés avec le développement de nouveaux algorithmes d'apprentissage.

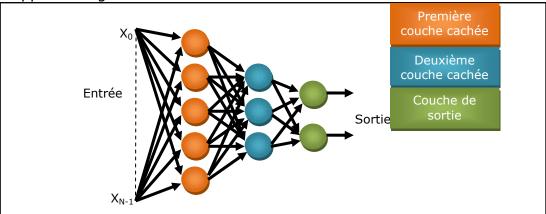


Figure 0:12: Perceptron multicouche.

3.7.5. L'algorithme de rétropropagation :

C'est une généralisation de l'algorithme LMS. Il emploie une technique de recherche de gradient pour minimiser une fonction coût égale à l'erreur quadratique moyenne entre les sorties désirées et réelles. Le réseau est entraîné au début en initialisant aléatoirement les poids avec des petites valeurs, ainsi que les seuils internes et puis en présentant toutes les données d'apprentissage sur plusieurs itérations. Les poids sont ajustés après chaque itération jusqu'à ce que les poids convergent.

L'algorithme est décrit comme suit :

Il requiert une fonction non linéaire différentiable. On utilise une fonction sigmoïde $f(\alpha)$ dont la formule est :

$$f(\alpha) = \frac{1}{1 + e^{-\alpha}}$$
 (II: 24)

• **Etape 1** : Initialisation : Initialiser les poids et les seuils, en les mettant à des petites valeurs aléatoires

- **Etape 2**: Présenter la base de données : présenter un vecteur d'entrée [x₀, x₁,..., x_{N1-1}] et spécifier le vecteur de sortie désirée [d₀, d₁, ..., d_{N2-1}].
- **Etape 3** : Calcul : Calculer les sorties réelles [Y₀,Y₁,...,Y_{N-1}], en propageant l'entrée à travers les différentes couches du réseau de neurones comme suit:

$$x_j^m = f\left(\sum_{i=0}^{L-1} W_{ij} x_i^{m-1} - \theta_j^m\right), \ 0 \le j \le N \ \text{et} \ 0 \le m \le M \ \dots$$
 (II: 25)

M : Le nombre de couches ;

 W_{ij} : Est le poids de connexion entre le j-ème neurone de la couche m et le i-ème neurone de la couche (m-1);

 x_i^m : Est la sortie de la couche m;

 x_i^{m-1} : Est l'entrée de la couche m (ou la sortie de la couche (m-1));

 θ_i^m : Est le seuil de la couche m.

• **Etape 4**: L'ajustement des poids : on utilise un algorithme récursif, et en commençant par les nœuds de sortie, pour les quels on utilisera la formule suivante d'ajustement des poids :

$$W_{ij}(t+1) = W_{ij}(t) + \eta \delta_j x_j^{M-1}$$
 (II: 26)

n: Terme du gain ;

 δ_i : terme d'erreur sur le nœud j.

Si le nœud j est une sortie alors :

$$\delta_{i}^{M} = Y_{i}(1 - Y_{i})(d_{i} - Y_{i})...$$
 (II: 27)

Où d_j est la sortie désirée du nœud j, et où Y_j est la sortie calculée Si j est un nœud interne alors le terme d'erreur sera :

$$\delta_j^{m-1} = x_j^m (1 - x_j^m) \sum_k \delta_k^m W_{jk}$$
 (II: 28)

Où k représente le k-ème nœud de la couche m (couche supérieure) Les seuils sont ajustés de la même façon que les poids, et ça en les supposant comme étant poids de connexion reliés à des entrées à valeurs. Parfois pour accélérer la convergence on utilise un terme α qu'on appelé momentum :

$$W_{ij}(t+1) = W_{ij}(t) + \eta \delta_i x_i + \alpha (W_{ij}(t) - W_{ij}(t-1))$$
 Où : $0 \le \alpha \le 1$ (II : 29)

• Etape 5 : Bouclage : Répéter a partir de "Etape 2" jusqu'a atteindre une erreur quadratique moyenne prédéfinie, ou jusqu'à atteindre un certain nombre d'itérations prédéfinies.

3.7.6. Carte auto-organisatrice de Kohonen :

Un important principe d'organisation des voies sensorielles dans le cerveau est que le placement des neurones soit ordonné et qu'il reflète souvent certaines caractéristiques physiques du stimulus externe ressenti. Cette organisation tonotopique dans la voie auditive s'étend jusqu'au cortex auditif. Bien qu'une grande partie de l'organisation de bas niveau soit génétiquement prédéterminée, il est probable qu'une partie de l'organisation à des niveaux plus élevés soit créée pendant l'apprentissage par des algorithmes qui favorisent l'auto-organisation. Kohonen présente un algorithme semblable, qui produit ce qu'il appelle les cartes à organisation automatique semblables à celles qui se produisent dans le cerveau.

4. Conclusion:

Les AG offrent la possibilité de trouver des solutions à des problèmes très variés dès lors qu'il est possible d'exprimer ces problèmes en termes d'optimisation de fonction. Plus précisément, deux conditions doivent être satisfaites : pouvoir coder les solutions du problème à résoudre par une suite finie de caractères (binaire ou réel) et être capable d'attribuer une valeur sélective à chaque suite de caractère (c.à.d. trouver une bonne fonction sélective).

La technique floue est une démarche très bien adaptée lorsque des données approximatives ou subjectives sont nécessaires pour traiter un problème. Elle permet en effet de traiter des informations provenant de la connaissance à priori du processus. L'analyse de ces informations est faite à l'aide des règles linguistiques.

Les RNA constituent un véritable moyen pour la résolution de plusieurs problèmes où les méthodes classiques ont montré leurs limites. Leur utilisation s'est avérée efficace dans les processus qui nécessitent une interaction avec l'environnement, et ce par leur pouvoir d'adaptation souvent appelé : Plasticité Synaptique. De plus, l'information à l'intérieur des RNAs étant traitée en parallèle et d'une manière distribuée, fait que le temps de calcul s'en trouve réduit. Ces caractéristiques les rendent candidats mêmes là où une solution classique existe déjà. La rapidité et la plasticité permettent, d'autre part, de les utiliser dans les applications qui se font en temps réel.

Chapitre III:

ANALYSE SPECTRALE PAR LES AG, LES RNA ET LES RESEAUX DE NEURONES FLOUES (RNF) :

1. Introduction:

L'analyse spectrale sert à déterminer les harmoniques constituant un signal périodique qui peut être, d'après le théorème de Fourier, décomposée en leur somme. La détermination de ces harmoniques consiste à déterminer leurs caractéristiques : fréquences, amplitudes et phases (A_i, f_i, φ_i) .

L'analyse spectrale par identification de paramètres, proposée ici est basée sur les AG pour déterminer les valeurs (A_i, f_i, φ_i) de chaque harmonique. Le calcul de ces paramètres est basé sur l'optimisation d'une fonction de coût, qui est la somme d'erreurs quadratique entre des échantillons du signal réel à analyser et des échantillons du signal estimé. D'autre part, on propose l'extraction en temps réel des harmoniques d'un signal périodique à base d'un réseau de neurones conventionnel ou d'un réseau de neurones flou. L'apprentissage effectué sur l'un de ces réseaux (soit le réseau de neurones conventionnel ou réseau de neurones flou) est le cœur de cette opération en modifiant chaque fois le rang de l'harmonique à extraire et le représenter comme cible pendant l'apprentissage.

On abordera dans ce chapitre la mise en œuvre de ces trois méthodes d'analyse spectrale.

La première méthode est d'utiliser les AG pour l'évaluation des (A_i, f_i, φ_i) des différentes harmoniques.

La deuxième méthode sert à l'extraction, en temps réel, de ces harmoniques par les RNAs.

La troisième méthode utilise les RNFOAGs.

2. L'identification des paramètres des harmoniques par les AG :

L'algorithme d'identification des amplitudes, fréquences et phases des harmoniques constituant un signal périodique s(t) est schématisé par la figure suivante :

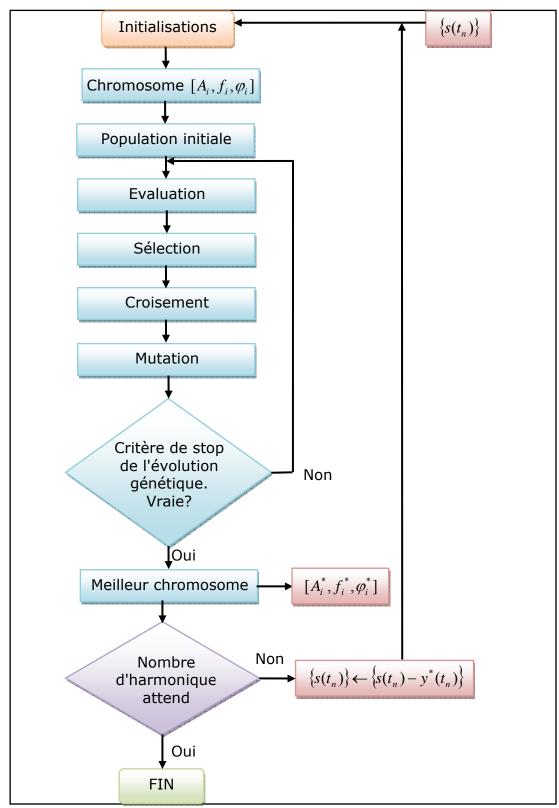


Figure 0:1: L'algorithme d'extraction par les AG.

2.1. Initialisation de l'algorithme :

Dans cette première étape, on déclare les différents paramètres tel que : la taille de la population à manipuler, le nombre maximal de générations,...

2.2. Codage des chromosomes :

Le codage utilisé dans ce cas est de type réel. Chaque chromosome est un vecteur constitué de trois éléments $[A_i,f_i,\varphi_i]$ représentant respectivement l'amplitude, la fréquence et la phase d'un harmonique à extraire.

2.3. Génération de la population initiale :

La population initiale est générée aléatoirement, mais en considérant les valeurs minimales et maximales des amplitudes, des fréquences et des phases. La taille de la population initiale est un nombre fixe M et qui reste constant au cours des générations successives. Notons que pour accélérer la recherche, il est convenable de prendre, pour chaque extraction, les valeurs initiales suivantes :

$$A_i = \max(\left\{s(t_n)\right\}....$$
 (III: 1)

$$f_{i} = \begin{cases} \min(fr\acute{e}quence); & pour \quad i = 0; \\ f_{i-1} + \min(fr\acute{e}quence); & pour \quad i \neq 0; \end{cases}$$
 (III : 2)

2.4. L'évaluation :

Le critère utilisé pendant l'extraction est de minimiser la moyenne des carrées des erreurs décrite par la fonction MSE suivante :

$$MSE = \frac{1}{N} \sum_{j=1}^{N} (s(t_j) - y_i(t_j))^2$$
 (III:3)

Ou

 $s(t_j)$; j = 1,...,N: N Échantillons de signal à traiter.

 $y_i(t_i)$; j = 1,...,N: N Échantillons de l'harmonique de rang i.

Notons que la minimisation de la fonction MSE est, d'un autre côté, la maximisation de son inverse.

2.5. Sélection:

La procédure de sélection utilisée est appelée la sélection géométrique normalisée décrite par les étapes suivantes :

- \bullet Génération de M nombres aléatoires rangés d'une manière croissante (vecteur V) ;
- Rangement des valeurs de la fonction d'évaluation d'une manière croissante ;
- Un individu j ($[A_i^j, f_i^j, \varphi_i^j]$) est sélectionné si : V(j) < G(j) avec G(j) calculé comme suivant :

•
$$rangfit(i) = \frac{p.(1-p)^{M-i}}{1-(1-p)^{M}}$$
 (III:4)

- i: rang de la valeur rangée de la fonction d'évaluation ;
- p : probabilité de sélection ;
- *M* : taille de la population ;
- $G(j) = \sum_{i=1}^{J} rangfit(i)$

2.6. Croisement:

La procédure de croisement utilisée est appeler le croisement arithmétique avec les étapes suivantes :

- Génération de M/2 nombres aléatoires dans l'intervalle [0, 1] ;
- Détermination du nombre de croisements à effectuer par calcul du nombre de nombres aléatoires inférieurs à la probabilité de croisement;
- A chaque croisement sélection aléatoire de deux individus dans la population (les deux parents (x_i^p, x_i^p));
- Génération d'un nombre aléatoire α avec $\alpha \in [-0.5,1.5]$;
- Génération des deux fils (x_k^f, x_l^f) utilisant les deux formules suivantes :

•
$$x_k^f = \alpha x_i^p + (1 - \alpha) x_j^p$$
....(III: 5)

•
$$x_i^f = \alpha . x_j^p + (1 - \alpha) . x_i^p$$
 (III: 6)

2.7. Mutation:

La procédure de mutation sélectionnée est décrite comme suit :

- Génération d'un vecteur G de même taille que la population dont les valeurs sont $(-1)^i$ avec i est la position d'élément.
- Génération d'un nombre aléatoire β;
- Le nouvel individu x_{new} est :

•
$$x_{new} = x_{old} + (\max(x) - x_{old}) \cdot (1 - \beta^{(1 - \frac{t}{M})^5})$$
; Si : $G(i) = 1$ (III : 7)

•
$$x_{new} = x_{old} - (x_{old} - \min(x)).(1 - \beta^{(1 - \frac{t}{M})^5})$$
; Si : $G(i) = -1$(III : 8)

Avec : t est la génération actuelle

M: Nombre total d'éléments de la population.

min(x); max(x): Respectivement les valeurs minimale et maximale d'un individu.

2.8. Evolution génétique :

Pour l'établissement du critère de fin de boucle d'évolution génétique, on peut considérer :

- Un nombre maximum de générations ;
- Pas d'évolution ou d'amélioration dans les résultats ;
- Atteinte d'un niveau d'erreur minimum.

2.9. Meilleurs individus:

Les meilleurs chromosomes vont représenter les paramètres des harmoniques extraits ($[A_i, f_i, \varphi_i]$), dans l'ordre croissant.

2.10. Evolution d'extraction :

Le critère de fin de la procédure d'extraction est généralement fixé comme suit :

- Fixer le nombre d'harmoniques à extraire ;
- Fixer la valeur minimale des amplitudes des harmoniques.

3. <u>Identification des paramètres des harmoniques par les recuits simulés (RS) :</u>

Dans le cas d'une optimisation par un RS, la procédure peut être décrite par le diagramme de la figure (III : 2)

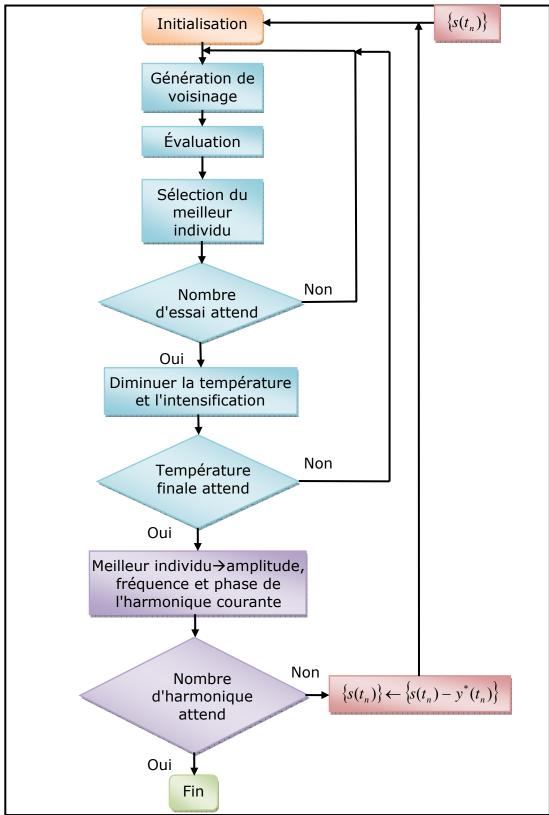


Figure 0:2: L'algorithme d'extraction par les RS.

Les AG utilisent une population d'individus, par contre, le RS procède avec un seul individu ou point x_i , et le nouveau point est choisi dans un voisinage $S(x_i)$ obtenu en modifiant les paramètres de ce point à l'aide d'une transformation élémentaire.

3.1. Génération de voisinage :

Un voisinage $S(x_i)$ d'un élément x_i est un ensemble de points (n points). Parmi ces points, un point est généré aléatoirement, comme pour la valeur initiale x_0 , de x_i qui est déjà générée. Un autre point est obtenu par élitisme, c.à.d. que le meilleur point x^* obtenu jusqu'à l'étape courante est introduit dans le voisinage $S(x_i)$ du point x_i . Les autres points sont obtenus à partir du point x_i en modifiant au moins, un des paramètres d'un pas aléatoire Δx_i et en prenant en considération le domaine de variation de chaque paramètre.

3.2. Diminution de la température :

Dans le cas simple, la diminution de la température se fait par une transformation linéaire, cela veut dire que la valeur prochaine de la température est égale au produit de la valeur actuelle par un facteur inférieur à 1.

3.3. L'intensification:

Pour améliorer la recherche locale, une fois le minimum global approché, une (ou plusieurs) intensification(s) ou concentration(s) de la recherche locale est (sont) faite(s) à partir d'un (ou des paliers) prédéterminé(s) de la température en diminuant le pas Δx_i d'un facteur lui aussi pré déterminé.

3.4. Sélection du meilleur individu :

C'est une sélection simple qui dépend seulement des valeurs d'évaluation. Cela veut dire que le meilleur individu est celui qui correspond à la valeur optimale de la fonction par laquelle le critère d'extraction est exprimé.

Les autres étapes de l'algorithme précédent sont identiques aux étapes de l'AG.

4. Extraction on-line des harmoniques utilisant les RNA [33] :

Il existe de nombreux travaux sur la proposition des techniques alternatives pour l'extraction des harmoniques. La mesure digitale récursive pour détecter des harmoniques du réseau d'alimentation proposée par Girgis et Al. [57] à base d'un filtre de Kalman (KF) pour estimer les composants harmoniques est l'une des méthodes d'extraction en temps réel des harmoniques. Dans ce travail nous proposons une nouvelle méthode d'extraction à base des RNAs.

4.1. Formulation du problème :

D'après le théorème de Fourier, chaque signal périodique est décomposable à une somme unique de sinusoïdes suivant l'expression suivante :

$$s(t) = A_0 + \sum_{i=1}^{N} A_i \sin(2\pi f_0 it + \phi_i) = A_0 + \sum_{i=1}^{N} h_i(t) \dots$$
 (III: 9)

Où:

 A_0 : La valeur moyenne de s(t) (généralement 0);

 f_0 : La fréquence du fondamental du s(t);

 $A_i; \phi_i$: Respectivement l'amplitude et la phase de l'harmonique de rang i. L'échantillonnage du signal s(t) avec une fréquence d'échantillonnage f_e donne un autre signal échantillonné :

$$s(n) = s(t) \Big|_{t=\frac{n}{f_e}}$$
 (III: 10)

Pendant l'apprentissage, un nombre approprié P des échantillons de s(t) ($\{s(n-(P-1)), s(n-(P-2)), ..., s(n-1), s(n)\}$) sera présenté à l'entrée du réseau de neurones ainsi que l'harmonique à extraire (A titre d'exemple le fondamental $h_1(t)$ sera présenté à la sortie comme cible ou objectif).

4.2. Architecture du RNA:

Un réseau multicouche d'une seule couche cachée de type "feed-forward" avec 5 neurones dans la couche cachée et 2 neurones dans la couche de sortie a été utilisé. Il est représenté en figure (III : 3)

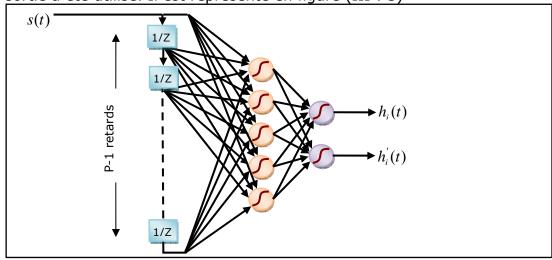


Figure 0:3: L'architecture du réseau de neurones utilisé.

La fonction de transfert de tous les neurones est de type sigmoïde unipolaire (logsig) et l'algorithme utilisé pour l'apprentissage est basé sur la technique du gradient

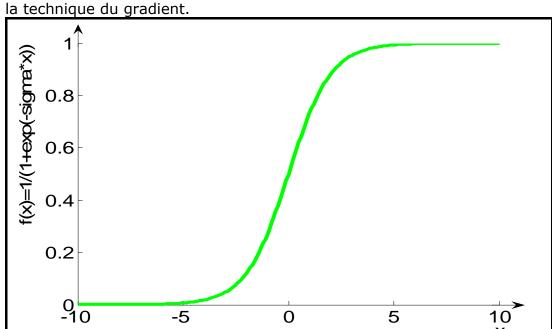


Figure 0:4: La fonction de transfert des neurones (logsig).

4.3. L'apprentissage [34]:

L'apprentissage du réseau sert à modifier, à chaque pas d'apprentissage, les poids et les biais afin de minimiser la somme des carrés des erreurs entre la sortie du réseau et la cible à obtenir. Désignons par Y le vecteur de sortie du réseau et par T le vecteur cible.

L'expression de l'erreur E est :

$$E_{i} = \frac{1}{2} (T_{i}^{T} T_{i} + Y_{i}^{T} Y_{i} - 2Y_{i}^{T} T_{i}) \dots$$
 (III: 11)

i : désigne le pas d'apprentissage.

La sortie du réseau est calculée par :

$$Y_i = f(W_o h_i + b_o)$$
 Avec: $h_i = f(W_c x_i + b_c)$(III: 12)

 $W_{\scriptscriptstyle o}$ / $W_{\scriptscriptstyle c}$: Matrices des poids de la couche de sortie et de la couche cachée respectivement ;

 $b_{\scriptscriptstyle o}$ / $b_{\scriptscriptstyle c}$: Matrices des biais de la couche de sortie et de la couche cachée respectivement ;

 h_i : La sortie de la couche cachée alimentant la couche de sortie au pas i;

 x_i : Le vecteur d'entrée au pas i;

f : La fonction de transfert considéré.

La mise à jour de la matrice W_o au pas i se fait par le calcul de sa variation qui est inversement proportionnelle au gradient d'erreur c.à.d.

$$\Delta W_o^i = -\eta \nabla E_i \Big|_{W_o} \text{Et} : W_o^{i+1} = W_o^i + \Delta W_o^i \dots$$
 (III: 13)

Dérivons l'erreur par rapport à W_a , on obtient :

$$W_o^{i+1} = W_o^i + \eta e_s h_i^T$$
; Avec: $e_s = (T_i - Y_i).*Y_i.*(I - Y_i)$(III: 14)

I : La matrice d'identité ;

 η : Le gain d'apprentissage ;

".*": Désigne la multiplication élément par élément.

La même méthode sera utilisée pour la mise à jour de b_a . On obtient :

$$b_o^{i+1} = b_o^i + \eta e_s$$
 (III: 15)

L'erreur en sortie sera rétropropagée à la sortie de la couche cachée à travers la transposée $\det W_o$, et par analogie on obtient pour la mise à jour de W_c et b_c :

$$W_c^{i+1} = W_c^i + \eta e_c x_i^T$$
; $b_c^{i+1} = b_c^i + \eta e_c$(III: 16)

Ou:
$$e_c = W_o^T e_s * h_i * (I - h_i)$$
 (III: 17)

4.4. Normalisation des entrées/sorties :

À cause de la limitation des valeurs de la fonction sigmoïde unipolaire utilisée ici, il est nécessaire de normaliser les entrées et les sorties avant les utiliser. Pendant l'apprentissage, il est nécessaire de normaliser l'entrée du réseau et la cible à obtenir. Mais pendant l'utilisation, l'entrée sera normalisée comme précédemment, tandis que la sortie du réseau sera dénormalisée (opération inverse). La normalisation d'une valeur $V_{réelle} \in [c,d]$ dans un autre intervalle [a,b] est réalisée suivant l'expression :

$$V_{normalis\acute{e}} = \frac{b-a}{c-d} V_{r\acute{e}elle} + \frac{a.c-b.d}{c-d} \qquad (III: 18)$$

L'opération inverse, c.à.d. la dénormalisation d'une valeur $V_{r\'eelle} \in [a,b]$ est réalisée suivant l'expression :

$$V_{d\acute{e}normalis\acute{e}} = \frac{c-d}{b-a} V_{r\acute{e}elle} + \frac{b.d-a.c}{b-a}$$
 (III: 19)

Le cas le plus fréquent est celui où les valeurs des facteurs (a, b, c, d) sont calculés comme suit :

$$a = -b = V_{net-max};$$
 $c = -d = V_{sig-max}$ (III : 20)

Où : $V_{net-max}$, $V_{sig-max}$ sont respectivement les valeurs maximales (n valeurs absolues) supportables par le réseau et le signal à traiter.

4.5. <u>La sortie :</u>

Elle est constituée de deux sorties : l'harmonique à extraire et sa dérivée. Ceci pour éviter la convergence des poids et des biais vers des valeurs correspondant à un minimum local d'erreur.

4.6. Optimisation par les AG:

Les valeurs des facteurs (a, b, c, d) peuvent être optimisées en utilisant les AG.

5. L'extraction on-line des harmoniques utilisant les RNF :

Dans ce qui suit, nous présentons une autre méthode qui a le même but que la précédente. Elle est à base des RNFs. Plusieurs architectures [35] sont possibles pour la fusion de RNA avec la LF, citons :

- Neuro & fuzzy;
- Neuro / fuzzy;
- Neuro-fuzzy ;
- Fuzzy-Neuro ;
- Neuro=Fuzzy;
- Fuzzy=Neuro ;
- Neuro-like Fuzzy ;
- Fuzzy-like Neuro;
- Neural fuzzy ;
- Neuro with fuzzy input output;
- Fuzzified-Neuro.

La différence entre ces méthodes est liée au degré de fusionnement entre l'architecture des RNAs et celle du contrôle par LF.

5.1. Architecture du RNF utilisé :

Le type de fusion entre l'architecture des RNA et celle du contrôle par logique floue utilisé ici est de type : (Neuro-like Fuzzy) qui est le plus courant et dans lequel le réseau de neurones est utilisé pour l'ajustement des fonctions d'appartenance de l'architecture du contrôleur flou, par exemple, l'utilisation de l'algorithme de rétro propagation [36].

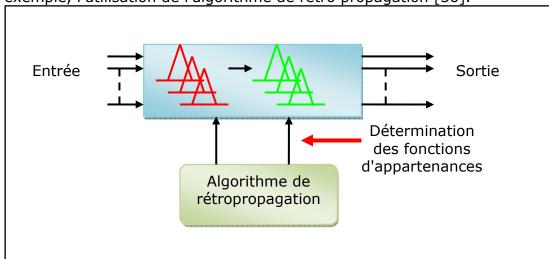


Figure 0 :5 : L'architecture résumée du Neuro-like Fuzzy.

L'architecture détaillée (avec seulement 3 entrées pour simplification) est la suivante :

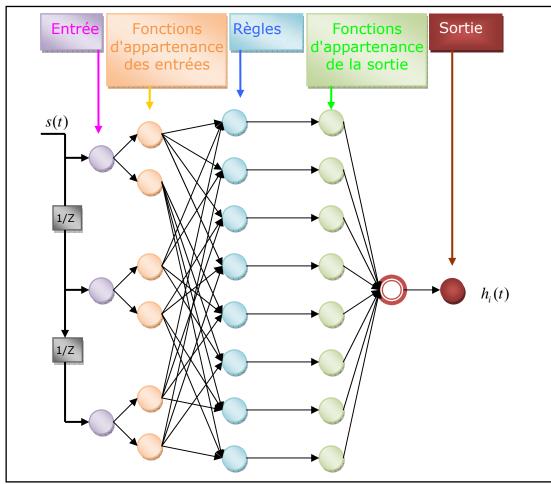


Figure 0 :6 : L'architecture du neurone floue utilisé.

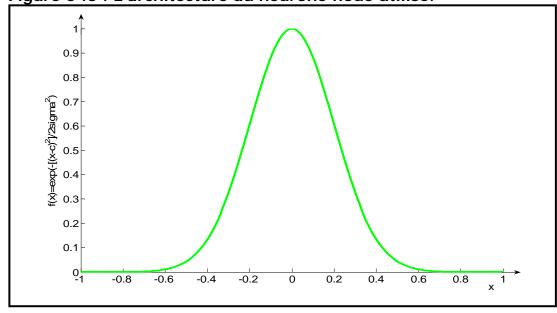


Figure 0:7: La fonction de transfert des neurones.

La fonction de transfert de tous les neurones est de type gaussien (gaussmf) et l'algorithme utilisé pour l'apprentissage est celui de la fonction d'apprentissage « anfis » de la boite à outils « Neural Network TOOLBOX ».

5.2. L'apprentissage :

L'algorithme d'apprentissage est utilisé dans la fonction « anfis » de la boite à outil « Neural Network TOOLBOX ». Il est basé sur l'algorithme de rétropropagation décrit précédemment ou sur un algorithme hybride combinant la méthode de la moyenne des carrés des erreurs et la méthode de rétropropagation pour l'ajustement des fonctions d'appartenance des entrées et de la sortie.

5.3. Normalisation des entrées/sorties :

Ici, la normalisation n'est pas nécessaire mais peut être utilisée pour un but de comparaison.

5.4. La sortie :

IL existe une seule sortie pour ce type de neuro-flou, qui est directement l'harmonique à extraire suivi par la dénormalisation appropriée.

5.5. Optimisation par les AG:

L'entrée et la sortie du réseau pendant la normalisation et la dénormalisation peuvent être optimisées en utilisant les AG pour calculer les valeurs appropriées des facteurs (a, b, c, d) de la fonction de normalisation et de dénormalisation.

6. Conclusion:

Pour mettre en œuvre avec succès un AG simple, il faut choisir avec soin les trois paramètres suivants : Probabilité de croisement Pc, Probabilité de mutation pm ; taille de la population I. Par ailleurs, il faut choisir entre divers mécanismes de sélection, de recombinaison et de mutation. Une difficulté supplémentaire réside dans l'interdépendance de ces divers paramètres et diverses études existent, apportant une aide dans ces choix délicats, des essais se révèlent indispensables pour optimiser les valeurs retenues en fonction du problème à résoudre.

Pour les RNAs et celles des RNFs, l'apprentissage est réalisé par des algorithmes d'optimisation dont le but est de minimiser la fonction de coût qui définit l'écart entre les sorties du réseau et les sorties réelles, en modifiant les poids synaptiques d'une manière itérative. Cet apprentissage est effectué après avoir fixé l'architecture du réseau à savoir le type de connexion, le nombre de neurones, le type de fonction d'activation, le nombre de couches cachées et leurs valeurs sont déterminées par essai. Cette procédure prend beaucoup de temps et ne converge pas nécessairement.

Chapitre IV:

TEST, SIMULATION ET RESULTATS

1. Introduction:

Pour mettre en évidence l'efficacité de la méthode d'estimation des paramètres $[A_i, f_i, \phi_i]$ des harmoniques par les AG, des tests ont été effectués sur différents signaux. Les résultats obtenus sont comparés avec les valeurs théoriques (Analyse de Fourier) et avec ceux issus de l'analyse spectrale par FFT. De plus, une comparaison avec les résultats obtenus par les recuits simulés sera effectuée. D'autre part, la construction de deux réseaux (RNAs et RNFOAGs) a été effectuée avec l'apprentissage approprié. La simulation des deux réseaux en temps réel a été effectuée à l'aide de Matlab, ainsi qu'une comparaison entre les harmoniques réels et extraits.

2. Signal test/simulation et critères de performance :

Dans ce qui suit on utilise, à titre d'exemple, le signal suivant qui est une somme pondérée de sinusoïdes défini par :

$$s(t) = \sum_{i=1}^{13} \frac{220}{i} \sin(2\pi 50it),$$

$$i = 1,3,5,...,13$$
 (IV: 1)

Ce signal contient un fondamental de fréquence $f_1 = 50Hz$ avec une amplitude $A_1 = 220(V)$ et des harmoniques de fréquences multiples de f_1 ,

c.à.d.
$$f_i = i \cdot f_1(Hz), i \ge 2$$
 et d'amplitudes $A_i = \frac{A_0}{i} = \frac{220}{i}(V)$ où i est l'ordre de l'harmonique.

Les critères de performance utilisés pour évaluer les résultats sont de minimiser les quatre expressions suivantes qui représentent l'erreur d'amplitude relative, l'erreur de fréquence relative, le moyen des carrées des erreurs (Mean Squared Error MSE) et le taux du MSE respectivement:

$$EAR(\%) = \frac{\left|A_{estim\acute{e}} - A_{r\acute{e}elle}\right|}{A_{r\acute{e}elle}} \times 100, \text{d'où} \qquad EFR(\%) = \frac{\left|F_{estim\acute{e}} - F_{r\acute{e}elle}\right|}{F_{r\acute{e}elle}} \times 100, \text{d'où} \qquad EFR(\%) = \frac{\left|F_{estim\acute{e}} - F_{r\acute{e}elle}\right|}{F_{e\acute{e}elle}} \times 100, \text{d'où} \qquad F_{r\acute{e}elle} \approx 100, \text{d'où} \qquad F_{r\acute{e}elle$$

$$MSE = \frac{1}{N} \sum_{j=1}^{N} (s(t_j) - y_i(t_j))^2, d'où$$

N: nombre des échantillons; $s(t_j)$: échantillon de s(t) à l'inst. t_j ;

 $y_i(t_j)$: échantillon de l'harmonique (i) $V_{\it efficace}$: la valeur efficace de l'harmonique $du \ s(t) \ \dot{a} \ l'inst. \ t_i$.

$$RMSE(\%) = \frac{\sqrt{MSE}}{V_{efficace}} \times 100, \text{d'où}$$

considérée.

3. Estimation des paramètres des harmoniques par un AG:

En utilisant l'algorithme décrit dans le chapitre précédent, les valeurs $[A_i,f_i]$ des 5 premiers harmoniques de s(t) réelles, estimées et calculées par FFT sont résumées dans le tableau suivant :

• Fenêtre périodique (3 périodes) :

	Valeurs re	éelles	Pa	r FFT	Par l'AG			
i	A_{i}	f_{i}	A_{i}	f_{i}	A_{i}	f_{i}		
1	220	50	220	49.996800	219.751713	49.8289486		
2	73.3333	150	73.3333	149.99040	73.2776342	149.7147999		
3	44	250	44	249.98400	44.0237689	249.7092589		
4	31.4286	350	31.4286	349.97760	30.1176259	349.7335990		
5	24.4444	450	24.4444	449.97120	26.1541192	449.6739020		

Table 0:1: Les différentes valeurs (fenêtre périodique)

	j	1	1 2 3 4						
EAR		0,1128577273	0,0759079436	0,0540202273	4,1712774352	6,9943185351			
LAK	FFT	0	0	0	0	0			
FFD	AG	0,3421028000	0,1901334000	0,1162964400	0,0761145714	0,0724662222			
EFR	FFT	0,0064000000	0,0064000000	0,0064000000	0,0064000000	0,0064000000			

Table 0:2: L'EAR et l'EFR (fenêtre périodique)

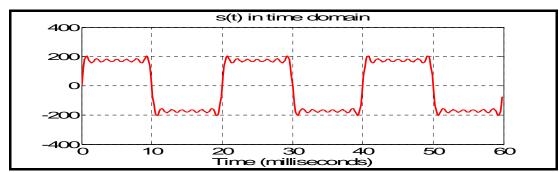


Figure 0:1: Signal s(t) - fenêtre périodique (3 périodes).

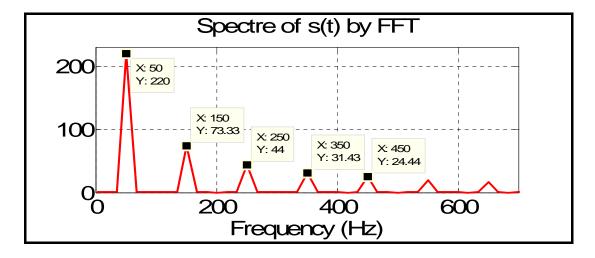


Figure 0:2: Spectre par FFT.

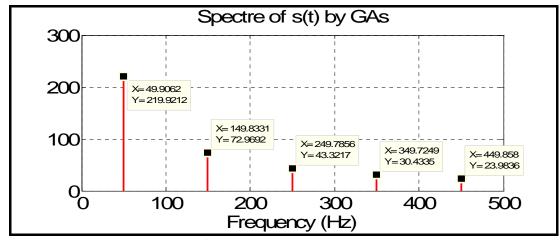


Figure 0:3: Spectre par les AG.

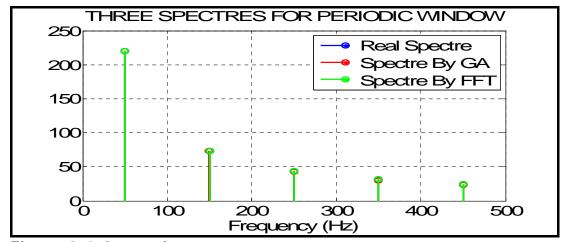


Figure 0:4: Les trois spectres.

• Fenêtre non périodique (3.5 périodes) :

	Valeurs re	éelles	Par l	FFT	Par l'AG				
i	A_{i}	f_{i}	A_{i}	f_{i}	A_{i}	f_{i}			
1	220	50	159.162	42.8544	219.9145522	49.8728441			
2	73.3333	150	46.6908	142.848	73.4350466	149.7982774			
3	44	250	28.3271	257.1264	46.3745752	249.7802375			
4	31.4286	350	20.9071	357.12	28.1153407	349.7920627			
5	24.4444	450	16.8187	457.1136	24.6141094	449.8236152			

Table 0:3: Les différentes valeurs (fenêtre non périodique)

	i	1	2	3	4	5
EAR	AG	0,0388399091	0,1387454267	5,3967618182	10,5421790980	0,6942669896
EAR	FFT	27,653636363	36,3306983321	35,62022727	33,477469565	31,19610217
FFD	AG	0,2543118000	0,1344817333	0,0879050000	0,0594106571	0,0391966222
EFR	FFT	14,2912000000	4,7680000000	2,85056000	2,0342857143	1,580800000

Table 0:4: L'EAR et l'EFR (fenêtre non périodique).

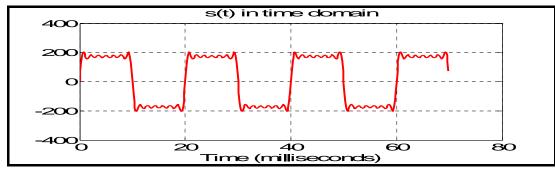


Figure 0:5: Signal s(t) - fenêtre non périodique (3.5 périodes).

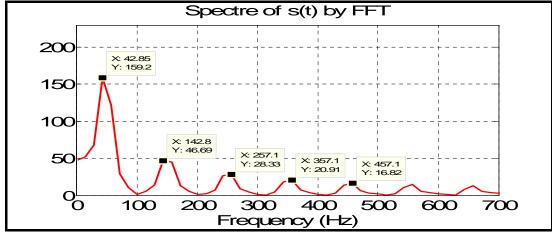


Figure 0:6: Spectre par FFT.

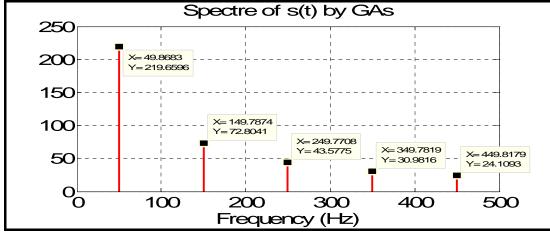


Figure 0:7: Spectre par les AG.

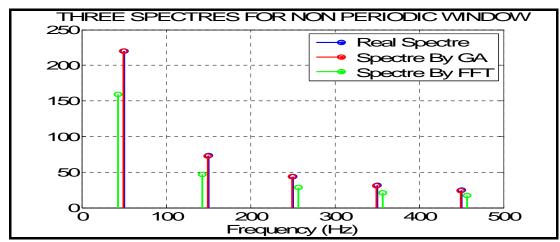


Figure 0:8: Les trois spectres.

Pour l'AG, le EAR maximum est égal à 10.54% et le EFR maximum est égal à 0.34%, alors que pour les FFT, EAR maximum peut arriver jusqu'à 36.33% et EFR maximum peut arriver aussi jusqu'à 14.29% et ça est dans le cas ou on prend une fenêtre non périodique pour faire le calcul.

3.1. Caractéristiques de l'AG utilisé :

- Taille de la population = 200 ;
- Nombre d'itérations = 500 ;
- Probabilité de sélection = 0.005;
- Probabilité de croisement = 0.5 ;
- Probabilité de mutation = 0.0001.

Ces caractéristiques sont obtenues après l'étude de l'effet des paramètres précédents sur le temps d'exécution et l'erreur maximale résultante dans le calcul de l'amplitude et de la fréquence. Cette étude est résumée dans les tableaux suivants (note que toutes les valeurs de ce tableau sont approximatives) :

TP		50	T	100)	150)	200)	25	0	3	00	3.	50	4	00		450)	500	
N/E / A //		100)	50		20		1		1			1		1	. 1			1		1	
ME (A/	F)	100)	300)	0		0		0		()	- (0		0 0		0			0
TEA(s		10		15		25		35		45	5	5	5	6	55	8	35		100)	138	
PS×	10 ⁻²		90)	50		10			5		1	0	.5		0.1		0.	05	(0.01	
ME (۸/F)		25	5	18		50		1	L3		5		L		50		1	8		14	
111 (<u> </u>		10	0	0		100)		0		0	()		100		30	00		1	
TEA			22	2	25		25		3	36	4	17_	4	0		30		2	7		24	
PC ×	10 ⁻²		90		70		50		30		10		9		7		5		3		1	
ME (A	\/F)		6		16		0		1		12		32		10	0	17	7	13		87	
_			100		100		0		0		100		100)	10		0	_	1		100	
TEA	(s)		21		20		44		46		131		82		79)	63	3	40		30	
PM×10	900	5	00	10	00	90	5	50	1	10	9	5		1		0.9	(0.1	C	.09	0.01	
ME	70	1	.5	2	2	5	:	31		1	1	25	5	1		1		1		1	1	
(A/F)	100	1	00	()	0	1	00		0	0	1		0		0		0		0	0	
TEA(s)	15	1	.5	2	5	23	3	33	(1)	38	37	33	3	39		36		28		38	34	
N	IPC			50		100)		150)		200)	3	00		4	00		5	00	
ME (A/E)			1		0			50			1			1		(0			0		
ME (A/F)			0		0			100)		0		(0			0			0		
TE	A(s)			27		36			28			33		2	9		3	7			37	

Table 0:5: L'effet des paramètres : TP, PS, PC, PM et NPC.

- TP: Taille de population;
- PS: Probabilité de sélection;

- PC : Probabilité de croisement ;
- PM : Probabilité de mutation ;
- NPC : Nombre de points à comparer dans le calcul des moindres carrées
- ME : Erreur maximum dans l'amplitude/ la fréquence ;
- TEA: Temps d'exécution approximatif.

4. Valeurs réelles, calculées par l'AG, par FFT et par RS :

Dans le tableau suivant, figurent les erreurs relatives dont les valeurs calculées par l'AG précèdent, par FFT et par un recuit simulé. Aussi, les quatre spectres sont donnés dans une seule figure (Figure (IV : 9), (IV :

10)) et cela pour une fenêtre périodique et non périodique.

FFT/EAR	AG/EAR	RS/EAR	FFT/EFR	AG/EFR	RS/EFR
0,00000	0,21188	1,20709	0,00640	0,19451	0,21020
0,00000	0,33337	1,21669	0,00640	0,09608	0,11639
0,00000	1,91070	2,54250	0,00640	0,06652	0,07446
0,00000	0,84687	1,31485	0,00640	0,03809	0,04390
0,00000	0,74279	0,30415	0,00640	0,01827	0,01156

Table 0:6: Comparaison entre les erreurs relatives (fenêtre périodique)

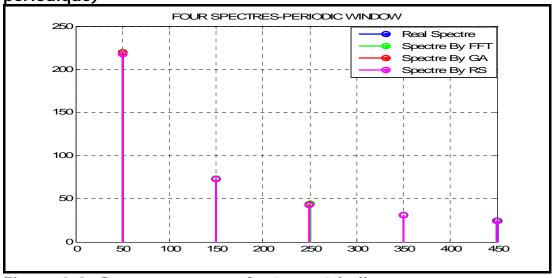


Figure 0:9: Quatre spectres : fenêtre périodique.

FFT/EAR	AG/EAR	RS/EAR	FFT/EFR	AG/EFR	RS/EFR
27,65350	0,21187	0,29476	14,29120	0,19445	0,25900
36,33070	0,33343	0,61333	4,76800	0,09608	0,13572
35,62023	1,91002	0,89141	2,85056	0,06653	0,08612
33,47747	0,84655	1,22109	2,03429	0,03809	0,05625
31,19606	0,74311	1,26144	1,58080	0,01830	0,03530

Table 0:7: Comparaison entre l'erreurs relatives (fenêtre non périodique)

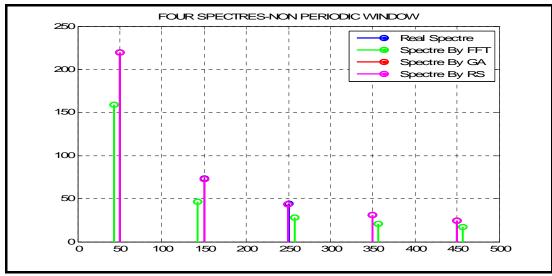


Figure 0:10: Quatre spectres : fenêtre non périodique.

Les valeurs calculées par l'AG et celles calculées par le RS considéré sont très proches mais les valeurs issues de l'AG sont plus précises.

4.1. Propriétés du RS utilisé :

- Température initiale = 1000 ;
- Température finale = 2 ;
- Diminution de la température par transformation linéaire α=0.65 ;
- Nombre d'essais par palier de température = 500 ;
- Le voisinage $S(x_i)$ de x_i est constitué de 10 points;
- 3 Intensifications utilisés.

5. Extraction des harmoniques on-line par les RNA :

Utilisant le même signal test, l'extraction de ses harmoniques on-line se fait par la construction de plusieurs réseaux. Chacun va extraire seulement un harmonique. Généralement, il suffit d'extraire seulement les cinq premiers harmoniques ayant la plus grande contribution dans le signal original ou total.

Le signal de test a été contaminé par un bruit blanc de puissance 0.00001. Les figures suivantes sont obtenues utilisant un RNA de 31 entrées.

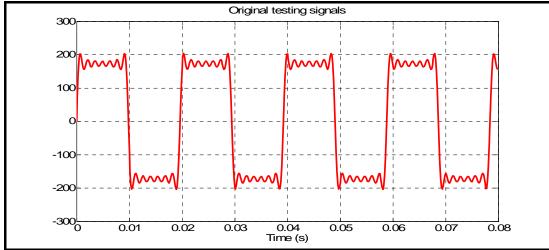
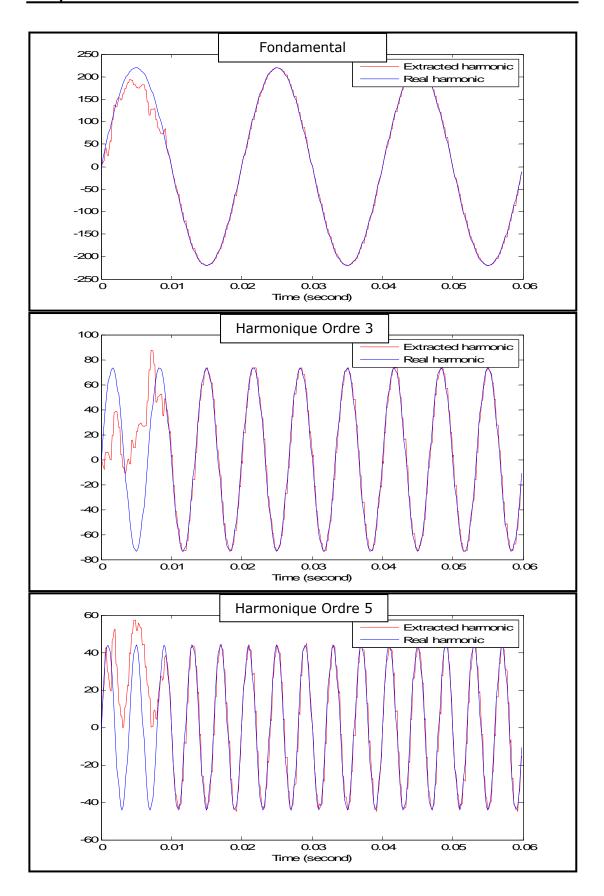


Figure 0:11: Signal s(t).



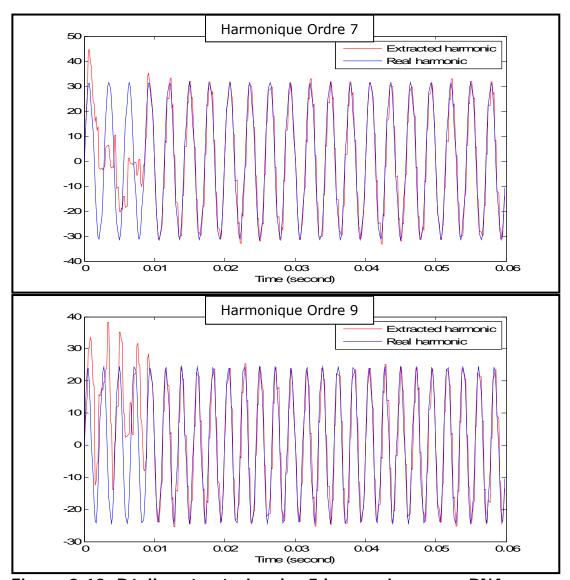


Figure 0:12: Réelles et extraies des 5 harmoniques par RNA.

5.1. Propriétés du réseau utilisé :

- Nombre d'entrée = 31 ;
- Nombre de neurones dans la couche cachée = 5;
- Nombre de neurones dans la couche de sortie= 2;
- Nombre d'époques d'apprentissage = 500 ;
- Type de fonctions de transfert : sigmoïde unipolaire ;
- Fréquence du fondamental=50 Hz.

5.2. Modèle de simulation SIMULINK:

Pour l'extraction d'un seul harmonique, il suffit d'utiliser un seul réseau entraîné et le modèle de simulation avec SIMULINK est donné en figure (IV: 14). De plus, si on veut extraire plus d'un harmonique (ex: 5 harmoniques), il faut utiliser des réseaux séparés. Un par harmonique à extraire comme illustré en figure (IV: 15) représentant le modèle de simulation SIMULINK.

On peut visualiser les signaux soit par oscilloscope soit par tracé après avoir exporté et sauvegardé les résultats dans un fichier.

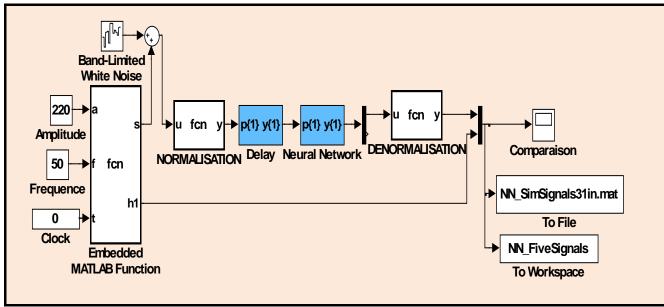


Figure 0:13: Model de simulation pour un seul harmonique.

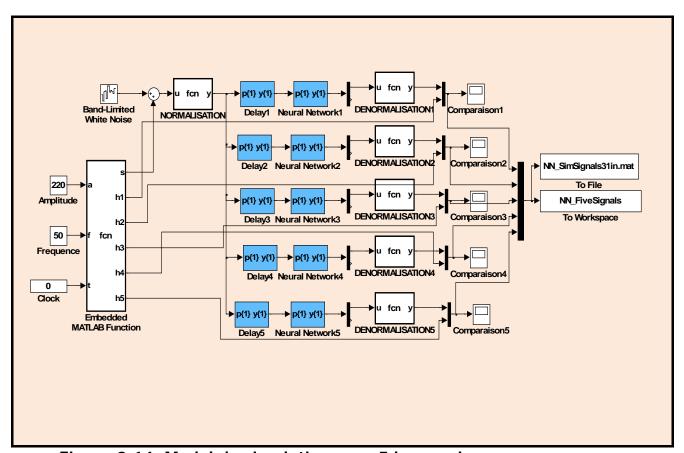
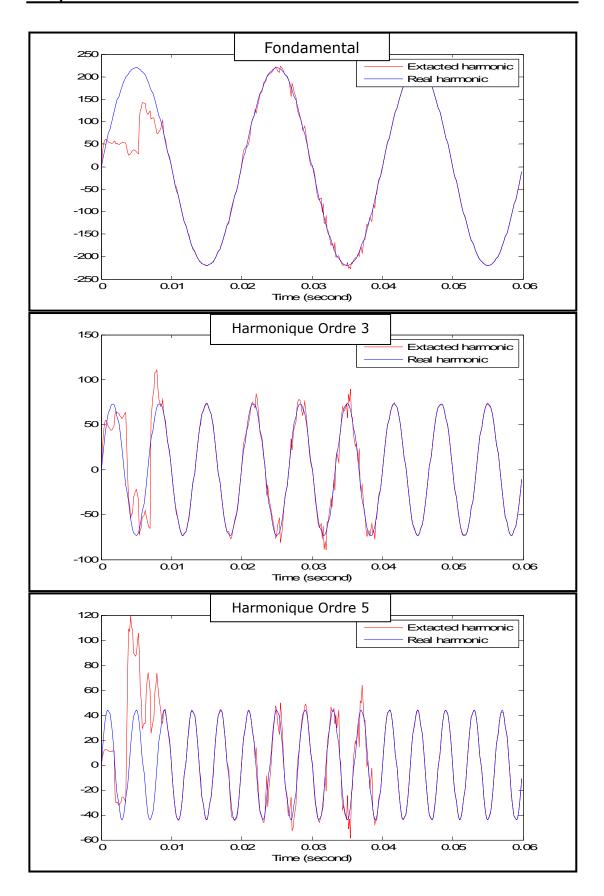


Figure 0:14: Model de simulation pour 5 harmoniques.

6. Extraction des harmoniques on-line par les RNFOAG :

Comme dans le cas précédent, c.à.d. l'utilisation du même signal, l'extraction de ses harmoniques on-line se fait par la construction de plusieurs RNFOAGs. Chacun va extraire seulement un harmonique. Les figures suivantes sont obtenues utilisant un RNFOAG de 06 entrées.



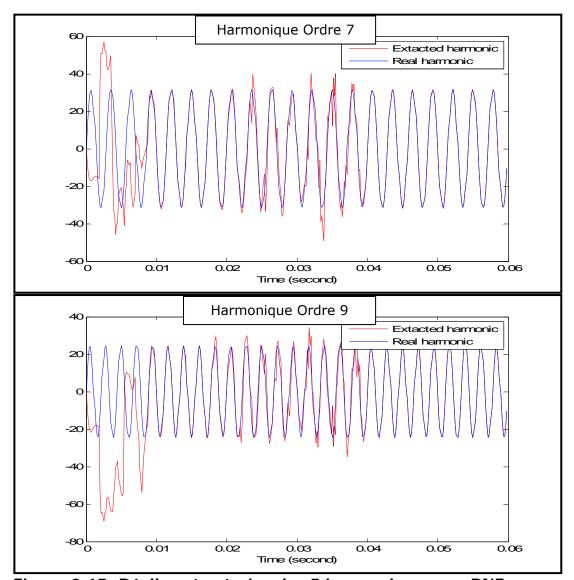


Figure 0:15: Réelles et extraies des 5 harmoniques par RNF.

6.1. Propriétés du RNFOAG utilisé :

- Nombre d'entrées = 6;
- Nombre de fonctions d'appartenance pour chaque entrée = 2 ;
- Nombre d'époques d'apprentissage = 10 ;
- Type de fonctions d'appartenance utilisées : gaussiennes ;
- Fréquence du fondamental=50 Hz.

6.2. Modèle de simulation SIMULINK :

Pour l'extraction d'un seul harmonique, le modèle de simulation est illustré en figure (IV : 16). Si on veut extraire, comme précédemment, 5 harmoniques, on utilise le modèle de la figure (IV : 17).

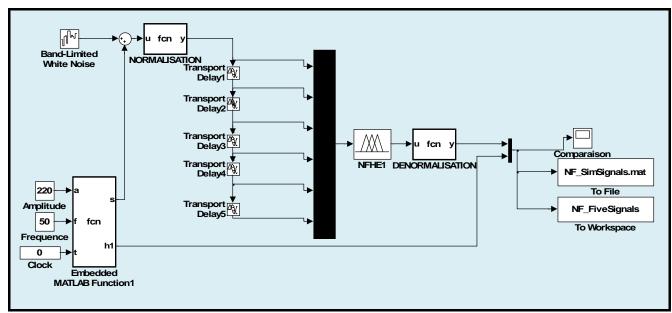


Figure 0:16: Model de simulation pour un seul harmonique.

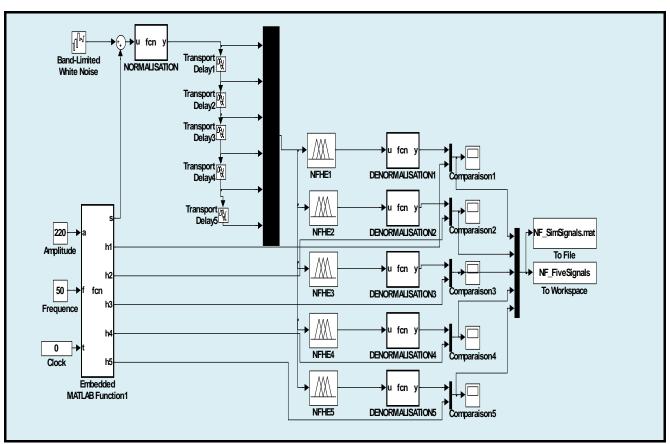


Figure 0:17: Model de simulation pour 5 harmoniques.

7. Comparaison entre RNA et RNFOAG:

Cette comparaison est effectuée entre quatre réseaux : deux RNA de 31 et 64 entrées (in) et deux autres RNFOAG de 6 et 8 entrées (in) prenant les 100 derniers points et calculant le MSE (Erreur quadratique moyenne). Les résultats obtenus sont dans le tableau (IV.8)

	R	NA	RNFOAG				
	31in	64in	6in	8in			
Fondamental	32.6489	0.11464	0.0014787	1.47E-06			
Har. ord 3	26.1275	0.38362	0.00066169	1.03E-05			
Har. ord 5	31.0541	0.50809	0.0014977	7.77E-06			
Har. ord 7	26.5705	0.45681	0.001101	6.97E-06			
Har. ord 9	24.5685	0.45711	0.00053314	6.45E-06			

Table 0:8: Comparaison entre des RNFOAGs et RNAs.

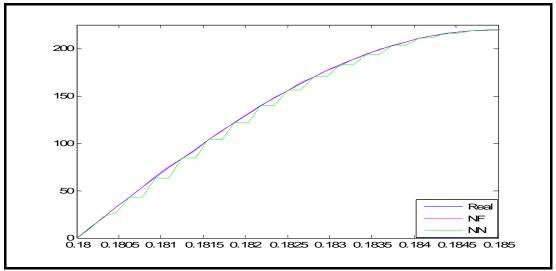


Figure 0:18: Un quart de période du fondamental : réel, extrait par RNA (31 in) et par RNFOAG (6 in).

La figure ci-dessus représente un quart de période du fondamentale de s(t), le trait rose c'est l'extrait par le RNFOAG et qui est presque totalement superposé avec le réel en bleu, le trait en vert c'est l'extrait par le RNA.

8. L'influence du nombre d'entrées :

Une étude a été conduite sur l'influence du nombre d'entrées d'un réseau de neurones flou sur la précision d'extraction. Toujours en calculant le MSE, les résultats obtenus pour les cinq harmoniques sont résumés dans le tableau suivant :

	5in	6in	7in	8in			
Fondamental	0.39023	0.0014787	0.00012944	1.47E-06			
Har. ord 3	0.48703	0.00066169	0.00015333	1.03E-05			
Har. ord 5	ar. ord 5 0.013937		0.0001221	7.77E-06			
Har. ord 7	lar. ord 7 0.33196		0.00015929	6.97E-06			
Har. ord 9	0.10423	0.00053314	0.0001582	6.45E-06			

Table 0:9: L'effet de nombre d'entrée sur la précision d'extraction

On peut conclure que pour plus de précision d'extraction, il faut utiliser plus d'entrées et cela est également valable pour les RNAs.

9. L'influence de la fréquence :

Prenons une variation de fréquence égale à $\pm 1\%$ de la valeur du fondamental, les différents MSE : à f=50 et à $f=50\pm 1\%$ pour différents nombre d'entrées sont résumés dans le tableau suivant :

	RNFOAG RNA											
f=50	5in		6in		7	7in		8in		31	in	64in
Fondamental	0.39	9023	0.0014787		0.00012944		44	1.47E-06		32.6	489	0.11464
Har. ord 3	0.48	3703	0.00066169		0.00015333		33	1.03E-05		26.1	275	0.38362
Har. ord 5	0.01	3937	0.0014977		0.0001221		1	7.77E-06		31.0	541	0.50809
Har. ord 7	0.33	3196	0.001101		0.00015929		29	6.97E-06		26.5	705	0.45681
Har. ord 9	0.10	0423	0.00053314		0.00	0.0001582 6			5.45E-06 2		685	0.45711
			F	NFC	AG					F	RNA	
f=50.5		5in	6in		7in		8in		31in			64in
Fondamenta	al 23	7.1111	1 15.845 14		.6503	6503 11.07		47	67.6645		5	9.4889
Har. ord 3	26	8.3937	7 14.1237 7		3868 6.7 5		758	50.89		.897	2	10.1084
Har. ord 5	2	1.759	11.6808 7		.7807 5.7		708	083 61		.8683		8.4302
Har. ord 7	18	0.6651	1 6.7651 7		.6215 6 .		5.5328		49.7086		9.4836	
Har. ord 9	75	75.7804		10.6008 7		5297 5.9 8		44 48.		.2605		10.7335
			RNF	OAG					RNA			
f=49.5		5i	n	6in		in	8	in	31	in	(64in
Fondamental		302.8	801 1	2.601	15 12.9	12.9978		1479 24.1		L48	9.2837	
Har. ord 3		322.6	591 1	2.857	74 7.0	7.0988		5351 16.4		257 10 .).2181
Har. ord 5		24.6	716	9.877	1 7.2	7.2269		6045 20.		595	8	.2573
Har. ord 7		206.6	751 6	5.437	6 6.5	6.5458		9859 17.		943	8	.6773
Har. ord 9		93.7	912 9.902		5 7.0	7.0564 5.		28 14.820 ⁴		204	11.0772	

Table 0:10: L'effet de variation de fréquence sur la précision d'extraction

Comparons entre un RNA de 64 entrées et un RNFOAG de 8 entrées, on peut voir que pour une range de variation de fréquence qui est de 48 à 52 Hertz, les plus petits valeurs de MSE sont celles du RNFOAG.

D'autre part, pour observer l'influence de la variation de fréquence, l'erreur RMSE relative a été utilisée, comme définie par (IV.2). Les erreurs pour différentes fréquences entre 48-50 Hz pour un RNA de 64 entrées et un RNFOAG de 8 entrées sont données dans les deux figures (IV : 19) et (IV : 20) respectivement. Cela est pour les trois premiers harmoniques constituant le signal de test s(t).

RMSE (%) =
$$\frac{\sqrt{mse}}{V_{efficace}} \times 100$$
 (IV: 2)

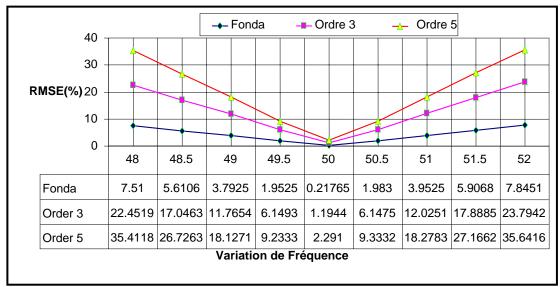


Figure 0:19: Le RMSE en pourcentage pour un RNA de 64 entrées.

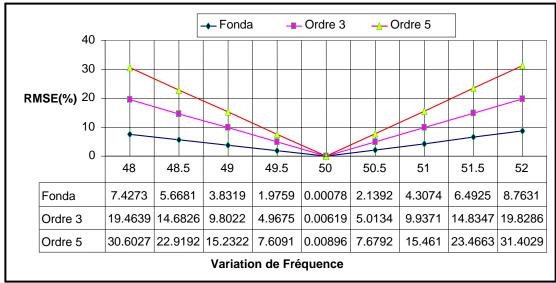


Figure 0:20: Le RMSE en pourcentage pour un RNFOAG de 8 entrées.

Pour les deux réseaux, les résultats prouvent que pour des variations de fréquence entre (50Hz±1%), les composantes harmoniques extraites changent seulement légèrement des composantes harmoniques réelles. Toutefois, pour la range de variation de fréquence considérée, on remarque que les plus petits valeurs de RMSE (en pourcentage) sont celles du RNFOAG.

10. L'influence de l'amplitude :

Prenons une variation d'amplitude égale à $\pm 5\%$ de la valeur du fondamental, les différents MSE : à A=220 et à $A=220\pm 5\%$ pour différent nombre d'entrée sont résumés dans le tableau suivant :

		RNFOAG								RNA		
A=220	5in	6in			7in		8in	3	31in	64in		
Fondamental	0.3902	23 0.00147		787	0.00	0.00012944		1.47E-0	6 32	.6489	0.11464	
Har. ord 3	0.4870	0.0	0066	5169	0.00	0.00015333		1.03E-0	<mark>15</mark> 26	.1275	0.38362	
Har. ord 5	0.0139	37 0.0	0014	977	0.0	0.0001221		7.77E-0	6 31	.0541	0.50809	
Har. ord 7	0.3319	96 0.	0011	.01	0.00	0.00015929		6.97E-0	<mark>16</mark> 26	.5705	0.45681	
Har. ord 9	0.1042	23 0.0	0.000533		0.0	001	582 6.45E- 0		<mark>16</mark> 24	.5685	0.45711	
			F	RNFC	DAG	AG			RNA			
A=231	5in		6	in	7iı	n		8in	31	1in	64in	
Fondamenta	10624	41.9416	49.1662		123.4	494	29.493		67.8291		44.2922	
Har. ord 3	5139	99.401	65.972		39.6	6751 3.E		.8252	28.	.374	3.4939	
Har. ord 5	7026	7026.6729		121.3818		14	4.1358		34.7312		1.5554	
Har. ord 7	3259	32599.0229 1		02.9784 29		167	4.0579		29.0756		2.8408	
Har. ord 9	2500	25000.7516 5		9804 41.6379		1	1.849		9542	1.527		
		RNF				FOAG				RNA		
A=209		5in	5in (n	7in		8in	31i	n	64in	
Fondamental		96705.4	05.4563 39		024 109.6693		5693	27.8407	62.18	75 4	1.7433	
Har. ord 3		44101.1	.01.1311 55.8		316	16 32.4597		3.8082	25.99	84 3	3.1443	
Har. ord 5		6356.65	56.6509 97.6		845	15 20.437		3.8362	32.24	42 1	3177	
Har. ord 7		28041.0	421	84.9	545 2	24.1	.579	3.5086	25.54	96 2	2.7057	
Har. ord 9		22708.3	398	42.6	742	33.4	487	1.595	24.23	43 1	.2823	

Table 0:11: L'effet de variation d'amplitude sur la précision d'extraction

Comparons entre un RNA de 64 entrées et un RNFOAG de 8 entrées, on peut voir que pour une range de variation d'amplitude qui est de 198 à 242 Volts, les plus petits valeurs de MSE sont celles du RNFOAG.

Pour observer l'influence des variations d'amplitude, l'erreur RMSE, comme précédemment, est utilisée. Les RMSE à différentes amplitudes entre 198-242 pour un RNA de 64 entrées et un RNFOAG de 8 entrées sont données dans les deux figures (IV:22) et (IV:23) respectivement. Cela pour les trois premiers harmoniques constituant le signal de test s(t).

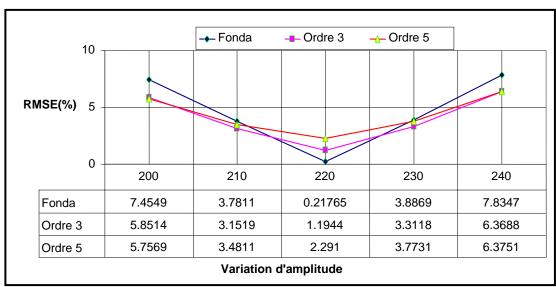


Figure 0:21: Le RMSE en pourcentage pour un RNA de 64 entrées.

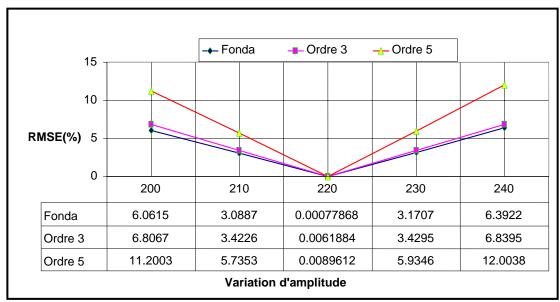


Figure 0:22: Le RMSE en pourcentage pour un RNFOAG de 8 entrées.

Pour les deux réseaux, les résultats prouvent que pour des variations d'amplitude entre (220±5%), les composantes harmoniques extraites changent seulement légèrement des composantes harmoniques réelles. Toutefois, pour la range de variation d'amplitude considérée, on remarque que les plus petits valeurs de RMSE (en pourcentage) sont celles du RNFOAG.

11. Interprétation et discussion :

Les résultats reportés précédemment démontrent que l'extracteur d'harmoniques basé sur un RNFOAG est capable d'extraire fidèlement les composantes harmoniques à partir d'un signal bruité. De plus, il est montré que cette méthode est aussi rapide pendant l'extraction, de même pour l'extracteur basé sur un RNA. Bien que le temps de réponse des filtres d'IIR puisse être réduit en abaissant l'ordre, l'ondulation inhérente à ce type de filtres est habituellement un souci important [37]. Le filtrage de Kalman (KF) semble fournir une extraction des harmoniques beaucoup plus rapide que les filtres conventionnels d'IIR. Cependant, les complexités informatiques telles que l'inversion de matrice impliquée dans l'opération de KF le rendent moins attrayant pour l'implémentation matérielle et l'utilisation en temps réel [33].

Cette méthode s'avère être robuste pour des variations courantes de fréquence et d'amplitude du fondamental et plus précise que l'extracteur à base de RNA (Figure (IV : 18), Table (IV : 8), Figure (IV : 20), Figure (IV : 24)).

12. Conclusion:

Les résultats obtenus à partir de l'extracteur des harmoniques basé sur un RNFOAG décrit précédemment montrent plus de rapidité et de précision d'extraction comparativement aux autres techniques conventionnelles, telles que le filtre de Kalman et l'extracteur des harmoniques basé sur un RNA. Les résultats montrent aussi que la technique étudiée est robuste vis-à-vis niveaux des variations de fréquence du fondamental et du bruit associé au signal examiné (normalement produit dans la pratique). Cette technique peut être employée pour l'extraction des harmoniques individuelles dans de nombreuses applications en temps réel.

CONCLUSION GENERALE

L'analyse spectrale par TFD est une technique dans le domaine fréquentiel beaucoup utilisée dans des applications à travers l'implémentation des algorithmes de calcul rapides (TFR ou FFT). Ces algorithmes sont suffisamment rapides pour être utilisés dans des applications en temps réel. D'autres techniques dans le domaine temporel tel que les filtres IIR peuvent être aussi utilisés dans de telles applications. Cependant, les problèmes liés à ces techniques limitent leurs utilisations. En utilisant une méthode d'identification en boucle fermée et un modèle basé sur les RNF optimisé par les AGs, ces problèmes peuvent être éliminés ou du moins minimisés

Les AG peuvent offrir la possibilité de trouver des solutions à des problèmes très variés dès lors qu'il est possible d'exprimer ces problèmes en termes d'optimisation de fonction. Les réseaux de neurones constituent un véritable moyen pour la résolution de plusieurs problèmes où les méthodes classiques ont montré leurs limites. De l'autre côté, la technique floue est une démarche très bien adaptée lorsque des données approximatives ou subjectives sont nécessaires pour traiter un problème. Combinant ces deux dernières techniques et utilisant les algorithmes génétiques pour l'optimisation, on peut obtenir un modèle d'identification qui peut surmonter les problèmes liés aux techniques classiques.

Cette technique floue prouve par les résultats obtenus qu'elle est plus rapide et précise pour l'extraction d'harmoniques des signaux bruités comparativement aux autres techniques conventionnelles, telles que le filtre de Kalman et l'extracteur des harmoniques basé sur un réseau de neurones conventionnel. Ses résultats montrent aussi que cette technique est robuste vis-à-vis des variations courantes de la fréquence du fondamental et du bruit associé au signal étudié. Pour cela, cette technique peut être employée pour l'extraction des harmoniques individuelles dans beaucoup d'applications.

PERSPECTIVE :

Les perspectives à court terme sont d'implémenter la technique proposée, c. a. d. le RNFOAG, sur un système de simulation en temps réel dSPACE de type DS1104 et d'extraire les harmoniques en temps réel d'un système de puissance hardware utilisant des oscilloscopes pour la visualisation des signaux réels et extraits.

BIBLIOGRAPHIE

- [1] : Analyse harmonique : les séries de Fourier. Mathématiques pour les sciences physiques, Alain ROBICHON.
- [2] http://www2.univ_reunion.fr/~lcks/Enseignement/CM_MPPA/Cours% 20TS_CM.doc
- [3]: A. ROBICHON, Analyse spectrale avec l'oscilloscope METRIX OX 8042 (ou 8062), Editions Dunod, 1989.
- [4]: R. Lorion, Eléments sur l'Echantillonnage, Editions Dunod, 1990.
- [5]: J. L. Schiff, The Laplace Transform: Theory and Applications, Wiley and sons, 1995.
- [6] : A. Cyril, Traitement du signal : transformée en Z, Edition Princeton, 1999.
- [7]: http://www.creea.u-bordeaux.fr/html/downloads.htm
- [8] : H. BOURLES, Introduction aux systèmes et mathématiques pour l'automatique, Cours B2 d'automatisme industriel, Octobre 2001.
- [9] : M. VILLAIN, Automatique 2, Systèmes asservis linéaires , Ellipses 1996.
- [10]: I. D. LANDAU, Identification in closed loop; a powerful design tool (Better design models, simpler controllers), Control Engineering Practice, 2000.
- [11] : S. Colonges, Identification des systèmes en boucle fermée, Janvier 2002.
- [12]: E. E. Altshuler, D. S. Linden, "Wire Antenna design using genetic algorithms,", IEEE Antennas and Propagation Magazines, vol.39, no. 2, April 1997.
- [13]: M. D. Vose, G. E. Liepins, "Punctuated equilibrium in genetic search," Complex Systems, vol. 5, pp. 31-44, 1991.
- [14]: D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Inc., 1989.
- [15] : G. Berthiau, "La méthode du recuit simulé pour la conception des circuits électroniques : adaptation et comparaison avec d'autres méthodes d'optimisation," Thèse de Doctorat de l'Ecole Centrale de Paris, Décembre 1994.

- [16]: S. W. MAHFOUD, D. E. GOLDBERG, "Parallel recombinative simulated anealing: A genetic algorithm", Parallel computing, 21, 1993.
- [17] : Z. Tidjani, "Commande décentralisée floue adaptative appliquée aux robots manipulateurs ", Thèse de magister, ENP, 1999.
- [18]: L. Zadeh, "Fuzzy sets", Information and Control, Vol. 8, 1965.
- [19]: J.S. R. Jang, "Neuro fuzzy modelling and control", Proceedings of the IEEE, Vol. 83, no.3, pp. 378-404, Mar. 1995.
- [20]: D. Driankov, H. Hellendoor and M. Reinfrank, "An introduction to fuzzy control", Springer-Verlag, Berlin Heidelberg, 1993.
- [21]: D. Dubois, H. Prade, "On the combination of uncertain or imprecise pieces of information in rule-based systems A discussion in the framework of possibility theory", Int. J. of Approximate Reasoning, 2(1), 65-87, (1988).
- [22]: T. M. Mendel, "Fuzzy logic systems for engineering: A tutorial", proceedings of the IEEE, Vol. 83, no. 3, pp. 345-377, Mar. 1989.
- [23]: E.H. MAMDANI, "Application of fuzzy algorithms for control of simple dynamic plant", Proceedings of IEEE, vol 121, n°12, pp 1585-1588, 1974.
- [24]: P.M. LARSEN, "Industrial Applications of Fuzzy Logic Control", International Journal of Man-Machine Studies, vol 12, n°1, pp 3-10, 1980.
- [25]: L.A. ZADEH, "A theory of approximate reasoning", Memorandum UCB/ERL –M77/58, 1977.
- [26]: H. Bühler, "Réglage par logique floue", Presse polytechnique et universitaire Romande, Lausane (Suisse), 1994.
- [27]: H. Bühler, "Réglage par logique floue", Presse polytechnique et universitaire Romande, Lausane (Suisse), 1994.
- [28]: H. Bandemer, S. Gottwald, Fuzzy sets, fuzzy logic, fuzzy methods with applications, Edition Wiley, England, 1995.
- [29] : J.F. Jodouin, Les réseaux neuromimétiques, modèles et applications, Editions Hermès, Paris, 1994.
- [30] : E. Davalo, P. Naim, Les réseaux de neurones, 2^{ème} édition, 1993.

- [31]: R.P. Lipmann, An introduction to computing with neural network, IEEE ASSP Magazine, April 1987.
- [32]: A. K. Jain. J. Mao., K. Mohiuddin, Artificial Neural Network, IEEE computer Special Issue On Neural Computing, 1996.
- [33] : J.V. Wijayakulasooriya, G.A. Putrus and C.H. Ng, Fast non-recursive extraction of individual harmonics using artificial neural networks, IEEE Proc.-Gener. Trans. Distrib., Vol. 152, No. 4, July 2005.
- [34]: M. Marie, M. Mokhtari, Application of Matlab and Simulink, Edition Ellipse, 1998.
- [35]: H. Takagi, Fusion Technology of Neural Networks and Fuzzy Systems: A Chronicled Progression from the Laboratory to Our Daily Lives, International Journal of Applied Mathematics and Computer Science, vol.10, no.4, pp.647–673, 2000.
- [36]: E. Caunic, Acquisition of fuzzy Knowledge by NN and GA, Second International Conference on Knowledge-Based Intelligent Systems, 21-23 April 1998, Texas, USA.
- [37]: A.A. Girgis, W.B. Chang, and E.B. Makram "A digital recursive measurement scheme for on line tracking of power system harmonics", IEEE Trans. Power Deliv., vol. 6, n°3, pp. 1153–1160, 1991.

ABSTRACT

Real-time harmonics monitoring has become a major concern in many applications, as in electrical power systems which are experienced to harmonic distortion due to non-linear loads. The time limitations and computational complexity associated with conventional techniques make it appealing to investigate alternative techniques for harmonic extraction. From another hand, almost applications, such as filter design, need techniques for the fast extraction of individual harmonic components. Therefore, a technique based on fuzzy neural networks optimized by genetic algorithms for the fast extraction of individual harmonic components is presented here. It uses a type of fusion between artificial neural network and fuzzy logic control whose membership functions are adjusted suitably, due to the artificial neural network learning ability, to accurately estimate, on-line, the individual harmonic components of a distorted signal. The proposed technique is realized and simulated. Results show that the proposed technique is significantly faster and accurate than conventional techniques.

Key words: neural networks, fuzzy logic, genetic algorithms, simulated annealing, fuzzy neural networks, real time.

ملخص

في العديد من التطبيقات, أصبحت المراقبة في وقت حقيقي للمركبات الجيبية للإشارة أمر مهم. كمثال, في شبكات الطاقة الكهربائية التي تتعرض إلى تشويش في المركبة الجيبية بسبب الحمولات الغير خطية. قيد الوقت والتعقيد الحسابي في التقنيات المألوفة يدفع للبحث في طرق بديلة لاستخراج المركبات الجيبية. من جهة أخرى, اغلب التطبيقات مثل صناعة المرشحات تحتاج إلى تقنيات أسرع لاستخراج المركبات الجيبية للإشارات. من اجل ذالك ,نقترح هنا تقنية تعتمد على الشبكات العصبية الغامضة محسنة باستعمال الخوار زميات الجينية لاستخراج أسرع للمركبات الجيبية للإشارات. تستعمل هذه التقنية نوع من الالتحام بين الشبكات العصبية والمراقبة بواسطة المنطق الغامض الذي تعدل دوال الانتماء فيه بشكل مناسب بفضل قابلية التعلم للشبكات العصبية وهذا من اجل الاستخراج الدقيق في وقت حقيقي للمركبات الجيبية لإشارة مشوشة. التقنية المقترحة تحققت وأجريت لها محاكاة بواسطة برنامج محاكاة. النتائج المحصل عليها أثبتت أن هذه التقنية أسرع وأدق من التقنيات المألوفة.

الكلمات المفتاحية: الشبكات العصبية, المنطق الغامض, الخوارزميات الجينية, الشبكات العصبية الغامضة, الوقت الحقيقي.