



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département d'Electronique

Laboratoire LDCCP

Mémoire de Magister

Option : Electricité Solaire

Présenté par : **OBEIDI THAMEUR**

Ingénieur d'Etat en Electronique

Thème

Application des Algorithmes Génétiques dans la Commande des Hacheurs MPPT

Membres du Jury

M HADDADI Mourad, Pr	Président
M LARBES Chérif, MC	Rapporteur
M AIT CHEIKH Med Salah, CC	Rapporteur
M TRABELSI Mohamed, MC	Examineur
M MALEK Ali, MR	Examineur
M ZERGUI Rachid, CC	Examineur

Juillet 2006

ملخص

تستعمل طريقة متابعة نقطة الاستطاعة العظمى في الأنظمة الكهروضوئية بغرض الحصول على استطاعة عظمى في مخرج اللوح الكهروضوئي مهما كانت المؤثرات الخارجية من حرارة وشدة ضوئية.

في عملنا هذا نستعرض طريقة جديدة في تتبع نقطة الاستطاعة العظمى تعتمد على المتحكم بالمنطق الغامض. هذه الأخيرة يتم تطويرها بواسطة نوع جديد من أساليب الذكاء الاصطناعي، ألا وهي الخوارزمية الوراثية وذلك من أجل تحسين خصائص الطريقة السالفة الذكر. وبواسطة المحاكاة بلغة (MATLAB, SIMULINK) على المتحكمين بالمنطق الغامض والمتحكم بالمنطق الغامض المطور بالخوارزمية الوراثية. ومع إدراج تغييرات مختلفة على الحرارة والشدة الضوئية، بينت نتائج المحاكاة الخصائص الجيدة للمتحكم بالمنطق الغامض وكذا التحسينات الهامة والمعتبرة التي طرأت عليه باستعمال الخوارزمية الوراثية.

كلمات مفتاحية : متابعة نقطة الاستطاعة العظمى، الأنظمة الكهروضوئية ، الخوارزمية الوراثية، المتحكم بالمنطق الغامض...

Résumé

La poursuite du point de puissance maximale par un contrôleur MPPT est utilisée dans un système photovoltaïque (PV) pour obtenir une puissance de sortie maximale quelles que soient les conditions de température et d'irradiation.

Ce travail présente une nouvelle méthode de contrôle MPPT basée sur la commande floue ainsi qu'une méthode d'optimisation de ce contrôleur basée sur les Algorithmes Génétiques (AGs) peut améliorer ses performances.

Des simulations effectuées sous MATLAB et SIMULINK sur ces deux contrôleurs, MPPT flou et MPPT flou optimisée par les AGs, sous différentes conditions de température et d'irradiation, ont confirmé les très bonnes performances d'un contrôleur MPPT flou et les améliorations apportées par l'optimisation par les algorithmes génétiques.

Mots clés : convertisseur DC/DC, poursuite du point de puissance maximale, systèmes photovoltaïque, les algorithmes génétiques, contrôleur flou...

Abstract

Maximum Power Point Tracking (MPPT) is used in photovoltaic (PV) system in order to maximize the output power of the PV array irrespective of the temperature and irradiation conditions.

This work presents a new MPPT controller which is based on fuzzy control and an optimization method of this controller based on genetic algorithms to improve its performances.

Simulations of these controllers, fuzzy MPPT and optimised fuzzy MPPT, were carried out using MATLAB and SIMULINK under different temperature and irradiation conditions. The obtained results have reinforced the very good performances of the fuzzy MPPT and the improvements introduced by the optimised fuzzy MPPT.

Key Words : DC/DC Converters, Maximum Power Point Tracking, Photovoltaic Systems, Genetic Algorithms, Fuzzy Logic Controller...

TABLES DES MATIERES

Introduction générale.....	1
----------------------------	---

CHAPITRE I

CONCEPTS DE BASE DE SYSTEME PHOTOVOLTAÏQUE

I.1.	Introduction.....	3
I.2.	L'effet photovoltaïque.....	3
I.3.	La photopile.....	3
I.4.	Caractéristiques de la cellule photovoltaïque.....	4
I.5.	Influence de l'ensoleillement et de la température.....	5
I.6.	Le module photovoltaïque.....	8
I.7.	Modèle mathématique d'un panneau photovoltaïque.....	8
I.8.	Les batteries.....	8
I.9.	Les Convertisseurs DC-DC (Les Hacheurs).....	10
I.9.1.	Convertisseur Buck.....	10
I.9.1.1.	Modèle mathématique équivalent.....	11
I.9.1.2.	Modèle approximé du convertisseur Buck.....	11
I.9.1.3.	Les ondulations des courants et des tensions.....	12
I.9.1.4.	Etude en régime continu.....	13
I.9.2.	Convertisseur Boost.....	15
I.9.2.1.	Modèle mathématique équivalent.....	15
I.9.2.2.	Modèle approximé du convertisseur Boost.....	16
I.9.2.3.	Les ondulations des courants et des tensions.....	16
I.9.2.4.	Etude en régime continu.....	17
I.9.3.	Convertisseur Buck-Boost.....	18
I.9.3.1.	Modèle mathématique équivalent.....	19
I.9.3.2.	Les ondulations des courants et des tensions.....	20
I.10.	Méthodes de poursuite MPPT.....	20
I.10.1.	Algorithme de contrôle MPPT adaptatif	21
I.10.2.	Algorithme Perturbation et Observation simple (P and O simple)	22
I.10.3.	Algorithme de l'incrémentement de conductance.....	23
I.10.4.	Méthodes MPPT par contre réaction de tension	25
I.11.	Conclusion.....	26

CHAPITRE II

LA COMMANDE PAR LOGIQUE FLOUE

II.1.	Introduction.....	27
II.2.	Terminologie de la logique floue	27
II.2.1.	Variable linguistique.....	27
II.2.2.	Fonctions d'appartenance.....	27
II.2.3.	Opérateurs de la logique floue.....	28
II.3.	Structure de base d'un contrôleur flou.....	28

II.3.1.	Fuzzification.....	29
II.3.2.	Les règles floues	30
II.3.2.1.	Les règles de Mamdani	30
II.3.2.2.	Les règles de Takagi Sugeno.....	30
II.3.3.	Inférence	30
II.3.4.	Defuzzification.....	32
II.3.4.1.	Méthode du maximum.....	32
II.3.4.2.	Méthode de la moyenne des maxima.....	33
II.3.4.3.	Méthode du centroïde.....	33
II.3.4.4.	Méthode de la somme pondérée.....	33
II.4.	Avantages et désavantages du réglage par logique floue.....	34
II.5.	Conclusion.....	34

CHAPITRE III

LES ALGORITHMES GENETIQUES

III.1.	Introduction.....	35
III.2.	Description détaillée.....	37
III.2.1.	Codage des données.....	37
III.2.2.	Génération aléatoire de la population initiale.....	38
III.2.3.	Gestion des contraintes.....	38
III.2.4.	Opérateur de croisement.....	38
III.2.4.1	Croisement réel.....	40
III.2.5.	Opérateur de mutation.....	42
III.2.6.	Principes de sélection.....	43
III.2.7.	La réinsertion.....	45
III.2.8.	Les paramètres d'un AG.....	46
III.2.8.1.	La taille de la population.....	46
III.2.8.2.	Le taux de croisement.....	46
III.2.8.3.	Le taux de mutation.....	46
III.3.	Améliorations classiques.....	47
III.3.1.	Introduction.....	47
III.3.2.	Scaling.....	47
III.3.3.	Sharing.....	50
III.4.	Avantages et inconvénients des algorithmes génétiques.....	51
III.5.	Conclusion	52

CHAPITRE IV

CONTROLEUR MPPT FLOU OPTIMISE PAR LES ALGORITHMES GENETIQUES

V.1.	Introduction.....	53
V.2.	Contrôleur MPPT Flou optimisé par les Algorithmes Génétiques.....	53
V.3.	Conception du contrôleur MPPT flou original.....	54
V.3.1.	Fuzzification.....	54
V.3.1.1.	Variables linguistiques.....	54
V.3.1.2.	Fonctions d'appartenances.....	54
V.3.2.	Méthode d'inférence.....	55

V.3.3.	Déffuzification.....	56
V.4.	Optimisation par les Algorithmes Génétiques.....	56
V.4.1.	Choix du critère d'optimisation.....	56
V.4.2.	Création de la population initiale	56
V.4.3.	Sélection.....	58
V.4.4.	Croisement.....	58
V.4.5.	Mutation.....	58
V.4.6.	Réinsertion.....	58
V.4.7.	Critère d'arrêt.....	58
V.4.8.	Fonctions d'appartenance du CLF optimale.....	59
V.5.	Simulation et résultats.....	60
V.5.1.	Simulation du système de poursuite MPPT.....	60
V.5.1.1.	Module PV.....	61
V.5.1.2.	Batterie Pb-acide.....	62
V.5.1.3.	Hacheurs.....	62
V.5.1.4.	Contrôleurs MPPT.....	63
V.5.2.	Résultats de simulation du système MPPT en utilisant un hacheur boost.....	64
V.5.2.1.	Test de performances.....	64
V.5.2.2.	Tests de robustesse.....	66
V.5.3.	Résultats de simulation du système MPPT en utilisant un hacheur buck.....	92
V.6.	Conclusion.....	100
	Conclusion générale.....	101
	Bibliographie.....	102

INTRODUCTION GÉNÉRALE

Avec l'augmentation des prix du pétrole et l'intérêt porté au développement durable, l'énergie solaire est devenue très intéressante et très attractive.

L'énergie solaire est disponible partout sur la planète en des degrés divers et elle est entièrement renouvelable. Son apport est variable, au gré des jours et des saisons, mais elle est relativement prévisible. Sa puissance maximale est de 1kW/m^2 et, même si elle est relativement diluée, son apport énergétique annuel pourrait répondre des milliers de fois à la consommation énergétique de la plupart des pays.

L'énergie solaire photovoltaïque (PV) est de plus en plus utilisée pour opérer diverses applications terrestres comme l'éclairage, les télécommunications, la réfrigération et le pompage. Les systèmes PV ne nécessitent aucun apport extérieur de combustible; de plus, le générateur lui-même ne contient aucune pièce mobile et ne requiert donc pratiquement pas d'entretien. Par conséquent, les coûts récurrents d'opération et de maintenance sont relativement faibles. Pour ces raisons, cette source d'énergie convient particulièrement bien pour les utilisations en milieu rural où les populations sont réparties dans de petites communautés et où la demande énergétique est relativement faible.

Pour une gestion optimale de l'énergie disponible à la sortie du générateur PV, les techniques d'optimisations, de régulation et de contrôle permettent d'augmenter le rendement énergétique de l'ensemble de l'installation PV, en transférant à la charge la puissance électrique maximale issue du générateur PV, et ce quelles que soient les variations de l'éclairement et de la température.

Le générateur PV, composé de cellules photovoltaïques interconnectées, fonctionne dans une large gamme de tension et de courant, ne fournit sa puissance électrique maximale qu'en un seul point de fonctionnement bien déterminé sur la caractéristique courant-tension. Ce point est appelé le Point de Puissance Maximale (MPP).

Pour une optimisation de la gestion de la production électrique, il est donc indispensable de se positionner en ce point, lequel malheureusement peut fluctuer en fonction des changements climatologiques, comme la température et l'éclairement [1]. Pour la poursuite efficace de ce point, il est recommandé d'incorporer un dispositif électronique, entre le générateur PV et la charge, qui est un convertisseur DC-DC. Ce dernier est commandé par son rapport cyclique par un autre dispositif de poursuite du MPP qui est appelé le contrôleur MPPT.

Dans ce mémoire on s'intéresse à la conception d'un contrôleur MPPT flou optimisé par les algorithmes génétiques (AGs). On va étudier l'utilisation des algorithmes génétiques pour la conception et l'optimisation d'un contrôleur par logique floue. La stratégie de contrôle ainsi synthétisée sera appliquée à la commande MPPT du système photovoltaïque.

Beaucoup de commandes conventionnelles appliquées aux systèmes non linéaires, complexes, instables, sont difficiles à concevoir et donnent lieu à de piètres performances. Pour de tels systèmes, les contrôleurs par logique floue offrent des solutions souvent meilleures, la conception de ces contrôleurs non linéaires, utilise la connaissance de l'expert (connaissance humaine) et l'imprécision implicite. En l'absence d'une telle connaissance, la conception des contrôleurs sera lente, non guidée et basée sur l'essai et l'erreur. Les AGs

procurent une manière de surmonter cette imperfection. Ces algorithmes emploient quelques concepts de la théorie de l'évolution pour rechercher des solutions optimales, dans un espace de solutions grand et complexe en un temps beaucoup plus rapide.

Les AGs sont utilisés pour l'optimisation des solutions d'un problème; particulièrement ceux qui sont analytiquement infranchissables. Ils sont inspirés, comme leurs noms l'indiquent, des concepts biologiques de la génétique et de l'évolution.

Les différentes étapes de cette recherche sont mentionnées ci après :

Dans le premier chapitre, nous présentons les principaux éléments composant le système photovoltaïque par une étude détaillée sur la modélisation du générateur PV et des hacheurs ainsi que de la batterie en terminant par une revue des différentes techniques MPPT, utilisées pour la poursuite du PPM.

Dans le second chapitre et le troisième chapitre les principes de base de la logique floue et des algorithmes génétiques sont expliqués respectivement.

Ensuite, dans le quatrième chapitre la stratégie de commande adoptée (l'optimisation d'un contrôleur par logique floue en employant des algorithmes génétiques) sera présentée avec les résultats obtenus.

Enfin, une conclusion et des perspectives sont proposées.

CHAPITRE I

CONCEPTS DE BASE DE SYSTEMES PHOTOVOLTAIQUES

I.1. Introduction

Ce chapitre décrit les concepts de base de systèmes photovoltaïques et de production d'électricité grâce à l'effet photovoltaïque. On y explique comment l'énergie solaire varie selon l'endroit et la saison.

Les principaux éléments du système photovoltaïque sont étudiés, en allant du panneau photovoltaïque, passant par le contrôleur et nous terminant par les batteries.

A la fin, nous présentons quelques méthodes de contrôle du point de puissance maximale (PPM).

I.2. L'effet photovoltaïque

L'énergie photovoltaïque (PV) est la transformation directe de la lumière en électricité. On utilise pour cela une photopile. L'effet photovoltaïque, c'est-à-dire la production d'électricité directement de la lumière, fut observé la première fois en 1839 par le physicien français Edmond Becquerel. Toutefois, ce n'est qu'au cours des années 1950 que les chercheurs de la compagnie Bell Telephone, aux États-Unis, parvinrent à fabriquer la première photopile, l'élément primaire d'un système photovoltaïque.

I.3. La photopile

Cette photopile, qu'on appelle aussi cellule solaire ou cellule photovoltaïque, est fabriquée à l'aide de matériaux semi-conducteurs. On peut la représenter comme une diode plate qui est sensible à la lumière.

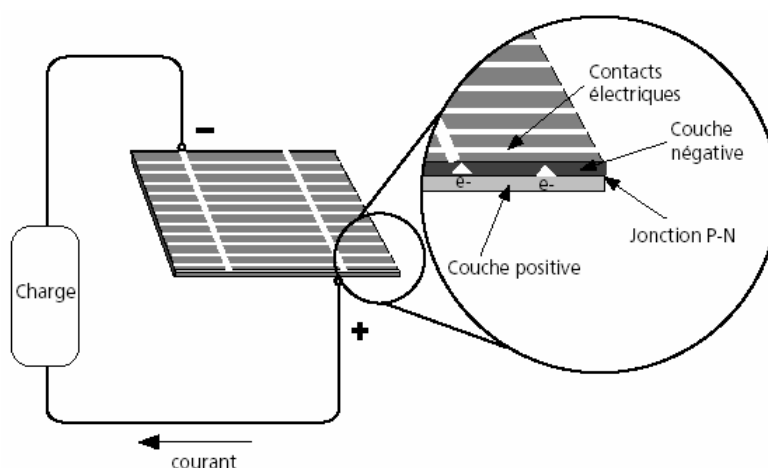


Figure I.1. Description d'une cellule photovoltaïque.

Quand un photon, d'énergie suffisante, heurte un atome sur la partie négative de cette diode, il excite un électron et l'arrache de sa structure moléculaire, créant ainsi un électron libre sur cette partie. Une photopile est fabriquée de manière à ce que cet électron libre ne puisse se recombiner facilement avec un atome à charge positive, avant qu'il n'ait accompli

un travail utile en passant dans un circuit extérieur. La cellule photovoltaïque produira de l'électricité à courant continu (CC), et son énergie produite sera fonction principalement de la lumière reçue par la photopile.

I.4. Caractéristiques de la cellule photovoltaïque

Le schéma du circuit équivalent d'une cellule photovoltaïque qui est largement utilisé dans la littérature [3], [5] est représenté sur la figure I.2 :

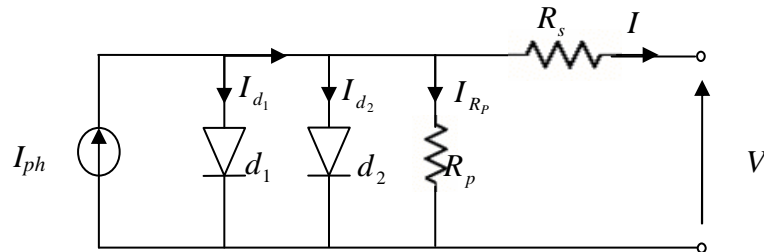


Figure I.2 Schéma du modèle équivalent à deux diodes d'une cellule photovoltaïque.

Comme le montre le schéma de la figure I.2 une photopile comporte en réalité une résistance série R_s et une résistance en dérivation ou shunt R_p . Ces résistances auront une certaine influence sur la caractéristique I-V de la photopile :

- La résistance série est la résistance interne de la cellule ; elle dépend principalement de la résistance du semi-conducteur utilisé, de la résistance de contact des grilles collectrices et de la résistivité de ces grilles ;
- La résistance shunt est due à un courant de fuite au niveau de la jonction ; elle dépend de la façon dont celle-ci a été réalisée.

D'après la figure I.2 le modèle mathématique pour la caractéristique courant-tension est donnée par [6]:

$$I = I_{ph} - I_{s1} \left[e^{\frac{q(V+I.R_s)}{n_1 k T}} - 1 \right] - I_{s2} \left[e^{\frac{q(V+I.R_s)}{n_2 k T}} - 1 \right] - \frac{V + I R_s}{R_p}. \quad (I.1)$$

Où :

I et V sont le courant et la tension de sortie de la cellule photovoltaïque,

I_{ph} est le photo-courant produit,

I_{s1} et I_{s2} sont les courants de saturation des diodes,

n_1 et n_2 les facteurs de pureté de la diode,

R_s et R_p sont respectivement la résistance série et la résistance parallèle,

T est la température absolue en Kelvin.

q ($1,602 \cdot 10^{-19} \text{ C}$) est la charge élémentaire constante

La constante de Boltzmann k ($1,380 \cdot 10^{-23} \text{ J/K}$).

Le photo-courant $I_{ph.max}$ est atteint à une insolation maximum, souvent on a ($I_{ph} = S \cdot I_{ph.max}$) avec S : pourcentage d'insolation.

Il est évident de l'équation (I.1), que la caractéristique courant-tension dépend fortement de l'insolation et de la température. La dépendance de la température est encore amplifiée par les propriétés du photo-courant I_{ph} et les courants de saturation inverse des diodes qui sont donnés par [5]:

$$I_{ph}(T) = I_{ph}|_{(T=298.K)} \left[1 + (T - 298 \cdot K) \cdot (5 \cdot 10^{-4}) \right] \quad (I.2)$$

$$I_{s1} = K_1 T^3 e^{-\frac{E_g}{kT}}, \quad (I.3)$$

$$I_{s2} = K_2 T^{\frac{5}{2}} e^{-\frac{E_g}{kT}}, \quad (I.4)$$

Où : E_g est la bande d'énergie du semi-conducteur et

$$K_1 = 1,2 \text{ A/cm}^2 \cdot K^3 \quad (I.5)$$

$$K_2 = 2,9 \cdot 10^5 \text{ A/cm}^2 \cdot K^{5/2}. \quad (I.6)$$

I.5. Influence de l'ensoleillement et de la température

Le courant produit par la photopile I_{ph} est pratiquement proportionnel à l'éclairement solaire S . Par contre, la tension V aux bornes de la jonction varie peu car elle est fonction de la différence de potentiel à la jonction N-P du matériau lui-même [5]. La tension de circuit ouvert ne diminuera que légèrement avec l'éclairement. Ceci implique donc que :

- La puissance optimale de la cellule (P_m) est pratiquement proportionnelle à l'éclairement;
- Les points de puissance maximale se situent à peu près à la même tension (voir figure (I.3)).

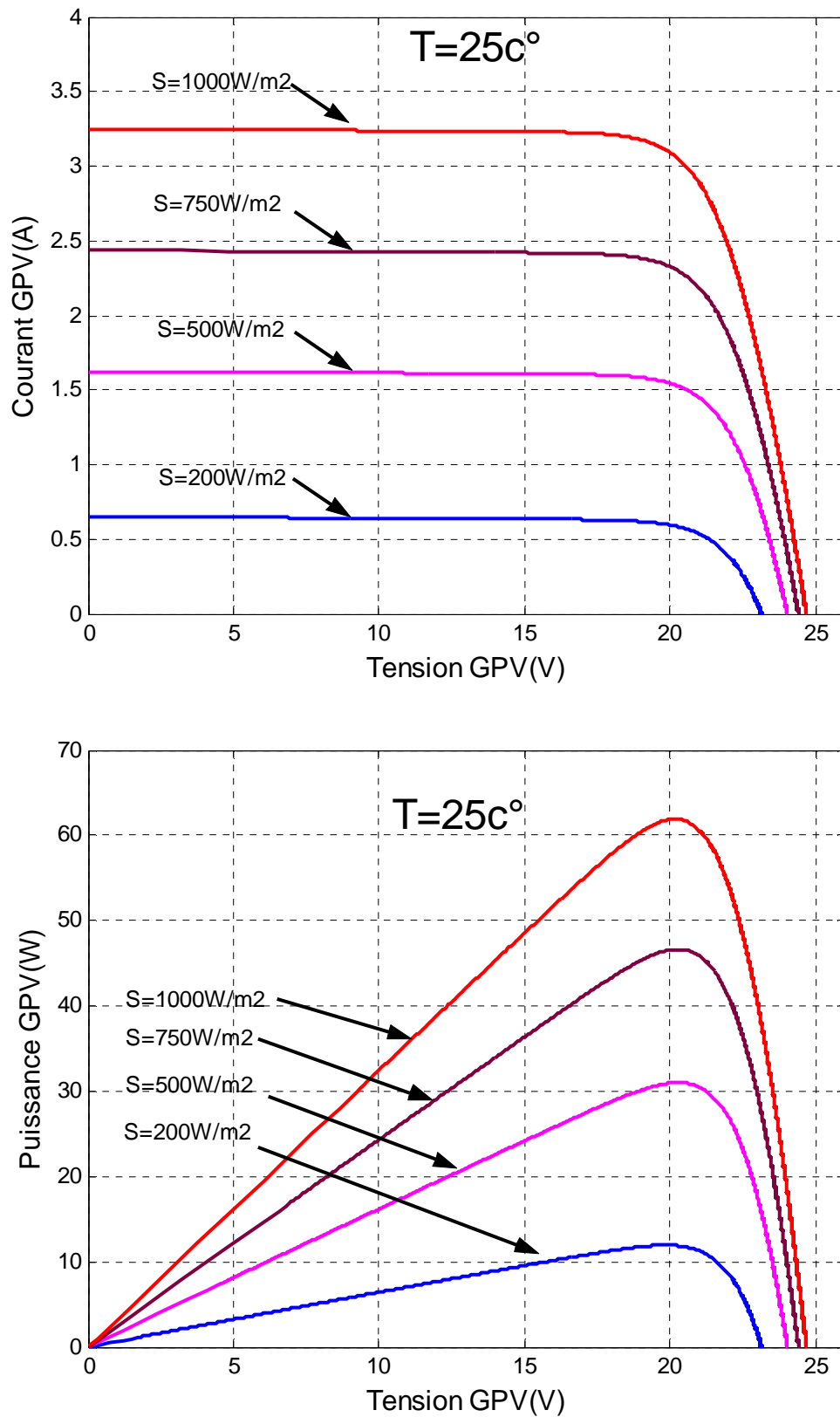


Figure I.3. Influence de l'ensoleillement sur les courbes I-V et P-V.

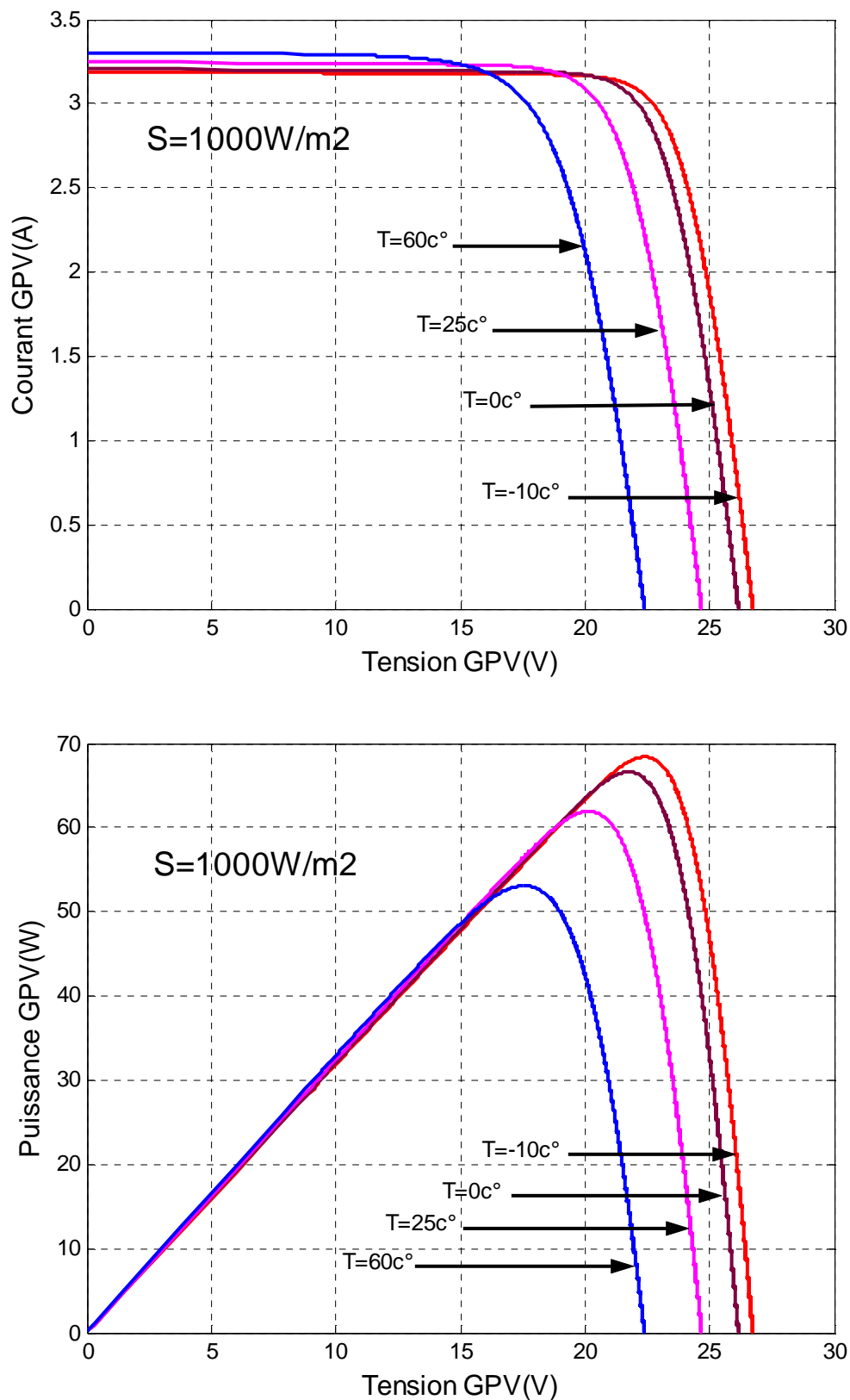


Figure I.4. Influence de la température sur les courbes I-V et P-V.

L'influence de la température est non négligeable sur la caractéristique courant/tension d'un semi-conducteur (voir figure (I.4)). pour une température qui change, on peut voir que la tension change considérablement tandis que le courant demeure constant.

I.6. Le module photovoltaïque

Afin d'augmenter la tension d'utilisation, les cellules PV sont connectées en série. La tension nominale du module est habituellement adaptée à la charge de 12 volts et les modules auront donc généralement 36 cellules. De plus, la fragilité des cellules au bris et à la corrosion exige une protection envers leur environnement et celles-ci sont généralement encapsulées sous verre ou sous composé plastique. Le tout est appelé un module photovoltaïque.

Les modules peuvent également être connectés en série et en parallèle afin d'augmenter la tension et l'intensité d'utilisation.

I.7. Modèle mathématique d'un panneau photovoltaïque

La considération du modèle de circuit équivalent (figure (I.2)) mène à l'équation (I.7) pour une rangée photovoltaïque de cellules (généralement considérée comme un panneau solaire) avec z cellules photovoltaïques raccordées en série [5].

$$I = I_{ph} - I_{s1} \left[e^{\frac{q(V+IzR_s)}{z n_1 k T}} - 1 \right] - I_{s2} \left[e^{\frac{q(V+IzR_s)}{z n_2 k T}} - 1 \right] - \frac{V + IzR_s}{zR_p}. \quad (I.7)$$

Les modules PV sont les éléments de base de tout système photovoltaïque. Ils peuvent être branchés en série pour augmenter leur tension d'utilisation et en parallèle pour augmenter leur courant. Cet ensemble est appelé le champ de modules PV.

I.8. Les batteries

Le système tampon utilisé le plus couramment pour les systèmes photovoltaïques, est la batterie d'accumulateurs électrochimiques. Les deux types de batteries, utilisés le plus couramment dans les systèmes photovoltaïques, sont les batteries avec accumulateurs au plomb-acide (Pb acide) et les batteries avec accumulateurs au nickel-cadmium (Ni-Cd). La batterie au plomb-acide est la plus connue, étant utilisée depuis plus de 150 ans pour fournir le courant de démarrage des voitures.

Une modélisation du circuit électrique de la batterie plomb-acide qui a été proposée dans la littérature [8], est la suivante :

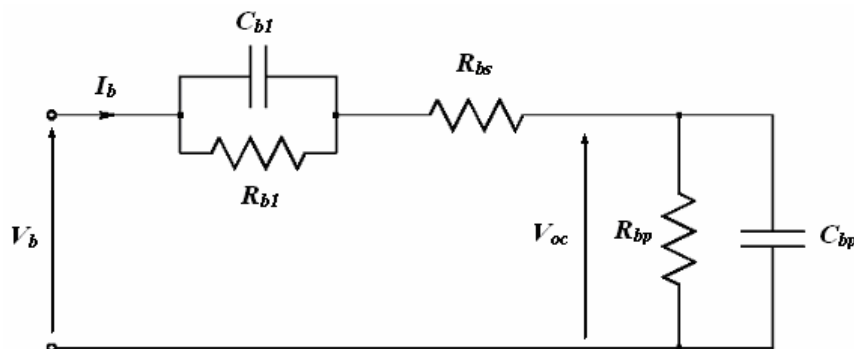


Figure I.5 Modèle équivalent de batterie.

C_{bp} est la capacité électrochimique de la batterie ; elle est donnée par l'expression générale de l'énergie :

$$E_c = \frac{1}{2} C V_c^2, \quad (I.8)$$

Avec C étant la capacité et V_c la tension du condensateur. À la différence d'un condensateur, la tension d'une batterie ne sera pas égale à zéro à son plus bas état de charge. C'est équivalent à un condensateur ayant un niveau minimum de charge égal à l'énergie $E_{c\min}$. La capacité entièrement chargée de la batterie est représentée par un niveau maximum de charge $E_{c\max}$. Ceci est indiqué par l'équation suivante [8]:

$$E_b = E_{c,\max} - E_{c,\min} = \frac{1}{2}CV_{\max}^2 - \frac{1}{2}CV_{\min}^2 = \frac{1}{2}C_{bp}(V_{\max}^2 - V_{\min}^2) \quad (I.9)$$

L'énergie E_b est donnée par le constructeur de la batterie directement en kilowatt par heure (kWh).

Les tensions V_{\max} et V_{\min} sont la tension maximale et la tension minimale de la batterie en circuit ouvert respectivement. La conversion de l'équation (I.9) mène finalement à une expression pour le condensateur représentant la capacité de la charge de la batterie

$$C_{bp} = \frac{2E_b}{V_{\max}^2 - V_{\min}^2} \quad (I.10)$$

La résistance interne de la batterie est représentée par les deux résistances R_{bs} et R_{b1} en série. La résistance en bloc d'électrolyte et de plaque est représentée par la résistance R_{bs} tandis que la résistance R_{b1} représente la diffusion d'électrolyte. Ceci représente la tension du circuit ouvert de la batterie dès qu'une charge sera reliée. De même on peut observer un saut soudain de tension avec l'application d'un courant de remplissage. En utilisant la notation indiquée sur la figure (I.5) on peut exprimer la tension de batterie V_b en fonction de la tension de batterie en circuit ouvert et des autres composants R_{b1} , R_{bs} et C_{b1} avec la constante de temps $\tau = R_{b1}C_{b1}$.

$$V_b = V_{oc} + R_{b1}\left(1 - e^{-t/\tau}\right)i_b + R_{bs}i_b \quad (I.11)$$

Une autre caractéristique très importante d'une batterie est la décharge spontanée, représentée par la résistance R_{bp} parallèle avec condensateur principal C_{bp} . Elle est provoquée par électrolyse de l'eau aux tensions élevées et par la fuite lente à travers les bornes de batterie aux basses tensions [9].

Le circuit sur la figure I.5 exprime l'impédance équivalente d'entrée d'une batterie plomb-acide par :

$$\begin{aligned} Z(s) &= R_{bs} + (R_{b1} \parallel C_{b1}) + (R_{bp} \parallel C_{bp}) \\ &= R_{bs} + \frac{R_{b1}}{R_{b1} \cdot C_{b1}s + 1} + \frac{R_{bp}}{R_{bp} \cdot C_{bp} \cdot s + 1} \end{aligned} \quad (I.12)$$

D'où :

$$Z(s) = \frac{a_2s^2 + a_1s + a_0}{b_2s^2 + b_1s + b_0} \quad (I.13)$$

Les coefficients a_i et b_j sont employés pour représenter les différents composants:

$$\begin{aligned}
a_2 &= R_{bs} R_{b1} R_{bp} C_{b1} C_{bp}, \\
a_1 &= R_{bs} R_{b1} C_{b1} + R_{bs} R_{bp} C_{bp} + R_{b1} R_{bp} C_{bp} + R_{bp} R_{b1} C_{b1}, \\
a_0 &= R_{bs} + R_{b1} + R_{bp}, \\
b_2 &= R_{b1} R_{bp} C_{b1} C_{bp}, \\
b_1 &= R_{b1} C_{b1} + R_{bp} C_{bp}, \\
b_0 &= 1.
\end{aligned}
\tag{I.14}$$

C'est la forme finale du modèle mathématique de la batterie plomb-acide qui sera employée dans les simulations de système en chapitre V.

I.9. Les Convertisseurs DC-DC (Les Hacheurs)

Dans cette partie nous présentons le principe des trois types de convertisseurs DC/DC (Buck, Boost et Buck-Boost), utilisés fréquemment dans les systèmes photovoltaïques pour générer les tensions et les courants souhaités. Ce type de convertisseurs n'est constitué que par des éléments réactifs (Selfs, Capacités) qui, dans le cas idéal, ne consomment aucune énergie. C'est pour cette raison qu'ils sont caractérisés par un grand rendement [5] [10] [2].

Dans ces études, l'interrupteur du convertisseur est attaqué par un signal MLI (*Modulation Large Impulsion*), avec une fréquence F_s fixe et un rapport cyclique D variable.

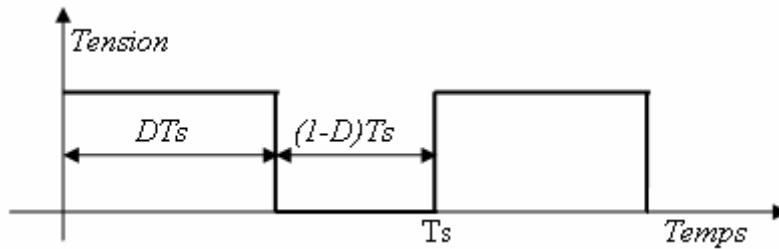


Figure I.6. Tension PWM pour la commande de l'interrupteur (transistor).

I.9.1. Convertisseur Buck

La figure I.7 donne le circuit électrique d'un convertisseur Buck. Le transistor MOSFET travaille en régime de commutation avec une période T_s . Dans le premier laps de temps DT_s le transistor est dans un état de saturation, alors l'inductance L se charge d'énergie avec augmentation du courant I_L . Dans le deuxième laps de temps $(1-D) T_s$ l'inductance L libère cette énergie à la charge Z avec une diminution de courant I_L . Alors le circuit est décomposé en deux circuits linéaires qui correspondent à chaque état du transistor S. La figure I.8 donne les schémas équivalents du convertisseur Buck dans les deux laps de temps.

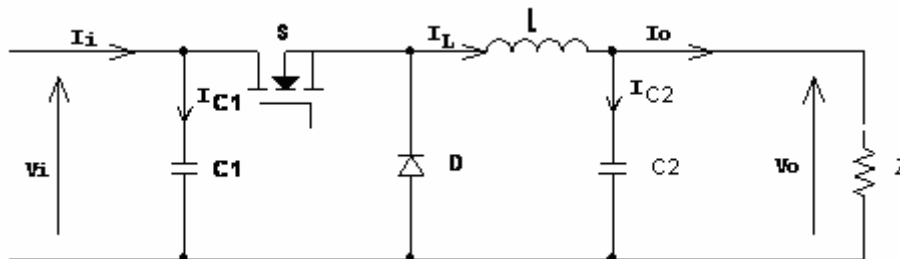


Figure I.7. Circuit électrique d'un convertisseur DC-DC de type Buck.

I.9.1.1. Modèle mathématique équivalent

Pour extraire le modèle mathématique du convertisseur, il faut l'étudier dans les deux phases de fonctionnement (S fermé, et S ouvert), ensuite donner son modèle approximé, qui englobe les différentes grandeurs moyennes d'entrée et de sortie du convertisseur.

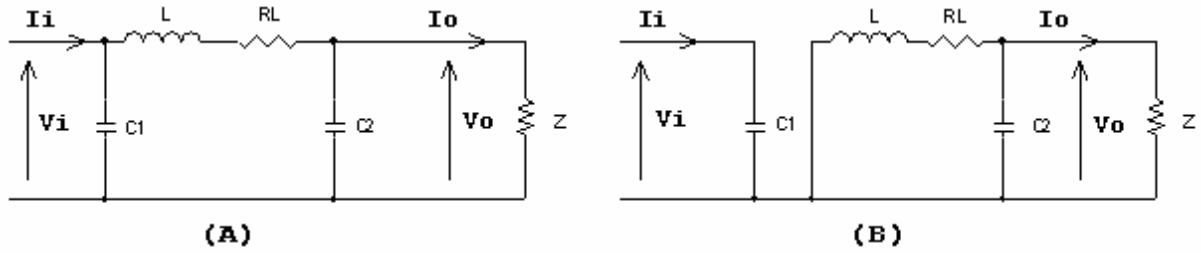


Figure I.8. Circuits équivalents du convertisseur Buck, (A) : S fermé, (B) : S ouvert.

Les variables dynamiques du circuit sont i_L , v_{C1} , v_{C2} , associée avec les composants L , C_1 , C_2 . Les équations qui lient les dérivées $\frac{di_L}{dt}$ et $\frac{dv_c}{dt}$, aux variables d'entrée et de sortie ainsi que les composants du convertisseur et les variables dynamiques i_L , v_c sont de la forme :

$$\frac{dv_c}{dt} = f(i_L, i_c, v_c, L, R_L, C) \quad (I.15)$$

$$\frac{di_L}{dt} = g(i_L, i_c, v_c, L, R_L, C) \quad (I.16)$$

Les grandeurs temporelles sont représentées par des lettres minuscules alors que les grandeurs moyennes sont représentées par des majuscules.

En appliquant les lois de Kirchhoff sur les deux circuits précédents, on obtient les systèmes d'équations suivantes :

$$\begin{cases} i_{C1} = C_1 \frac{dvi}{dt} = i_i - i_L \\ i_{C2} = C_2 \frac{dv_o}{dt} = i_L - i_o \\ v_L = L \frac{di_L}{dt} + R_L i_L = v_i - v_o \end{cases} \quad 0 < t < DTs \quad (I.17)$$

$$\begin{cases} i_{C1} = C_1 \frac{dvi}{dt} = i_i \\ i_{C2} = C_2 \frac{dv_o}{dt} = i_L - i_o \\ v_L = L \frac{di_L}{dt} + R_L i_L = -v_o \end{cases} \quad DTs < t < Ts \quad (I.18)$$

I.9.1.2. Modèle approximé du convertisseur Buck

Les systèmes d'équations de base (I.17) (I.18) représentent le convertisseur Buck pour une période DTs et $(1-D)Ts$ respectivement. Le convertisseur oscille entre ces deux états avec une fréquence élevée, nous devons trouver une représentation dynamique approximée valable pour les deux intervalles de temps. Pour cela nous considérons que la variation des variables dynamiques i_c , v_L est de forme linéaire, en d'autres termes nous pouvons faire une approche d'exponentielle par un segment ($e^\varepsilon \approx 1 + \varepsilon$ Si $\varepsilon \ll 1$) et ainsi la dérivée de ces grandeurs sera constante.

Cette approche nous permet de décomposer l'expression de la valeur moyenne de la dérivée de la variable dynamique x sur les deux laps de temps DTs et $(1-D)Ts$:

$$\left\langle \frac{dx}{dt} \right\rangle_{Ts} = \frac{dx}{dt}_{(DTs)} \cdot DTs + \frac{dx}{dt}_{((1-D)Ts)} \cdot (1-D)Ts \quad (I.19)$$

Où $\left\langle \frac{dx}{dt} \right\rangle$ est la valeur moyenne de la dérivée de x sur une période Ts . Cette relation est valide si $\frac{dx}{dt}_{(DTs)}$ et $\frac{dx}{dt}_{((1-D)Ts)}$ sont constants sur les périodes DTs et $(1-D)Ts$ respectivement, en d'autres termes cette approximation est valable si les périodes DTs et $(1-D)Ts$ sont très faibles devant la constante de temps du circuit C_1Rg , C_2Z , L/R_L [10].

Dans ce cas la forme exponentielle du courant qui parcourt la self et la tension aux bornes de la capacité est de forme linéaire comme le montre la figure I.9.

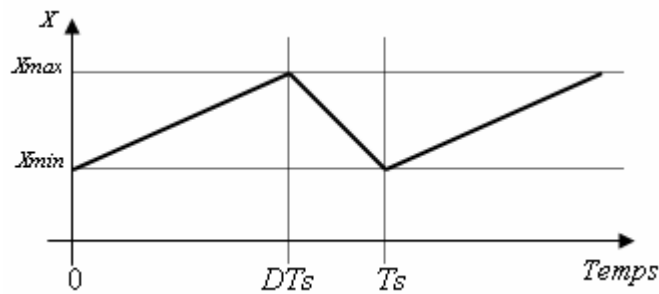


Figure I.9. Allure des variables dynamiques I_L .

En appliquant la relation (I.19) sur les systèmes d'équations (I.17) et (I.18), on obtient les équations qui régissent le système sur une période entière :

$$\begin{cases} C_1 \frac{dv_i}{dt} Ts = DTs(i_i - i_L) + (1-D)Ts i_i \\ C_2 \frac{dv_o}{dt} Ts = DTs(i_L - i_o) + (1-D)Ts(i_L - i_o) \\ L \frac{di_L}{dt} Ts = DTs(v_i - v_o - R_L i_L) + (1-D)Ts(-v_o - R_L i_L) \end{cases} \quad (I.20)$$

En arrangeant les termes des équations précédentes pour qu'on puisse interconnecter le convertisseur avec les autres blocs de simulation, on obtient la modélisation dynamique du convertisseur Buck :

$$\begin{cases} i_o = i_L - C_2 \frac{dv_o}{dt} \\ i_L = \frac{1}{D} (i_i - C_1 \frac{dv_i}{dt}) \\ v_i = \frac{1}{D} (v_o + R_L i_L + L \frac{di_L}{dt}) \end{cases} \quad (I.21)$$

I.9.1.3. Les ondulations des courants et des tensions

Pour le dimensionnement des différents composants du circuit afin de diminuer les ondulations des courants et des tensions sans faire un surdimensionnement ce qui accroîtrait le poids et le prix des circuits, un calcul de ces composants en fonction des ondulations voulues est nécessaire. Cette remarque est très importante pour le dimensionnement de l'inductance L

afin de respecter le courant admissible par le transistor MOSFET S , où dans le cas pratique les ondulations du courant I_L sont plus importantes par rapport aux autres ondulations.

En appliquant la relation $v_L = L \frac{di_L}{dt}$, et par l'approximation des segments d'exponentielles par des droites, la pente du courant I_L pendant la première période de fonctionnement est donnée par :

$$\frac{di_L}{dt} = \frac{v_L}{L} \approx \frac{V_i - V_o - R_L I_L}{L} \quad (\text{I.22})$$

A partir de la relation (I.22), la valeur crête à crête du courant I_L est :

$$I_{Lcc} = 2\Delta I_L = \frac{V_i - V_o - R_L I_L}{L} DT_s \quad (\text{I.23})$$

La valeur de l'inductance L à choisir pour certaine ondulation ΔI_L est :

$$L = \frac{V_i - V_o - R_L I_L}{2\Delta I_L} DT_s \quad (\text{I.24})$$

Pour le calcul des capacités C_1 et C_2 , on a :

$$\frac{dV_{c1}}{dt} = \frac{i_{c1}}{C_1} \approx \frac{I_i - I_L}{C_1} \quad (\text{I.25})$$

$$\frac{dV_{c2}}{dt} = \frac{i_{c2}}{C_2} \approx \frac{I_L - I_o}{C_2} \quad (\text{I.26})$$

Les valeurs des ondulations crête à crête des tensions d'entrées et de sorties sont :

$$V_{icc} = \frac{I_i - I_L}{C_1} DT_s \quad (\text{I.27})$$

$$V_{occ} = \frac{I_L - I_o}{C_2} DT_s \quad (\text{I.28})$$

Les valeurs des capacités C_1 et C_2 sont respectivement données par :

$$C_1 = \frac{I_i - I_L}{2\Delta V_i} DT_s \quad (\text{I.29})$$

$$C_2 = \frac{I_L - I_o}{2\Delta V_o} DT_s \quad (\text{I.30})$$

I.9.1.4. Etude en régime continu

Le régime continu est obtenu en éliminant les dérivées des variables dynamiques, et en remplaçant ces signaux par leurs valeurs moyennes.

Le système d'équations I.21 donne :

$$\begin{cases} I_i - DI_L = 0 \\ I_o - I_L = 0 \\ DV_i - V_o - R_L I_L = 0 \end{cases} \quad (\text{I.31})$$

Rapport de conversion et rendement :

Le rapport de conversion M est défini comme étant le rapport entre la tension de sortie et la tension d'entrée comme suit :

$$M(D) = \frac{V_o}{V_i} = \eta \cdot D \quad (\text{I.32})$$

Où η est le rendement du convertisseur défini comme étant le rapport entre la puissance de sortie sur la puissance d'entrée :

$$\eta = \frac{P_o}{P_i} = \frac{V_o I_o}{V_i I_i} \quad (\text{I.33})$$

La relation (I.27) donne :

$$M(D) = \frac{V_o}{V_i} = \frac{1}{1 + \frac{R_L I_o}{V_o}} D = \frac{1}{1 + \frac{R_L}{Z}} D = \eta D \quad (\text{I.34})$$

Avec :

$$\eta = \frac{1}{1 + \frac{R_L}{Z}} \quad (\text{I.35})$$

Avec Z l'impédance complexe de la charge.

A partir des relations (I.34) et (I.35) on conclut que le rapport de conversion M reste linéaire en fonction de D et confiné entre zéro et la valeur du rendement. Et des charges Z faible causent une grande perte dans le transfert de puissance à travers le convertisseur ainsi qu'une tension de sortie faible.

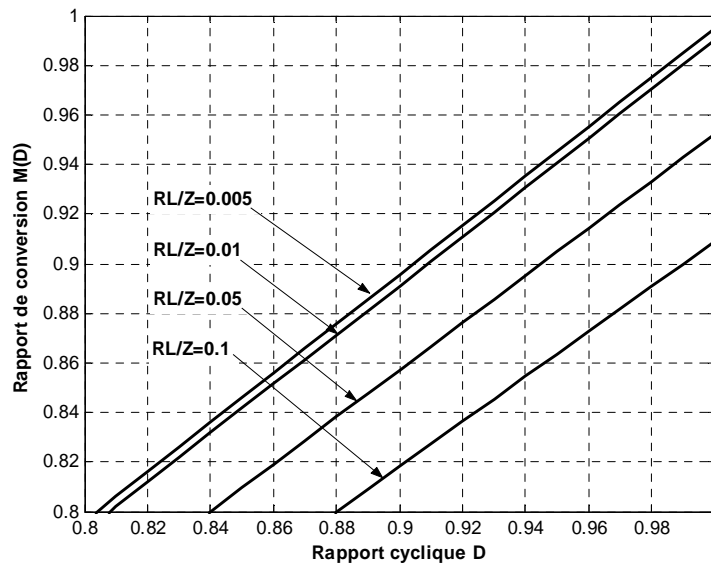


Figure 1.10. Rapport de conversion $V_o/V_i(D)$ d'un convertisseur Buck en fonction de D avec paramètre R_L/Z .

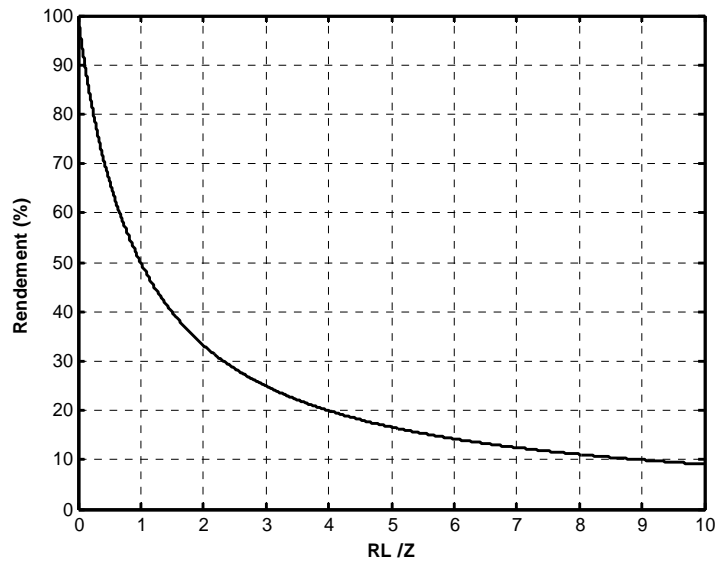


Figure I.11. Le rendement η d'un convertisseur Buck en fonction de R_L/Z .

I.9.2. Convertisseur Boost

Le convertisseur Boost est connu par le nom d'élévateur de tension. Le schéma de la figure 1.12 représente le circuit électrique du Boost. Au premier temps DT_s , le transistor S est fermé, le courant dans l'inductance croît progressivement, et au fur et à mesure elle emmagasine de l'énergie, jusqu'à la fin de la première période. Ensuite, le transistor S s'ouvre et l'inductance L s'opposant à la diminution de courant I_L , génère une tension qui s'ajoute à la tension de source, qui s'applique sur la charge Z à travers la diode D.

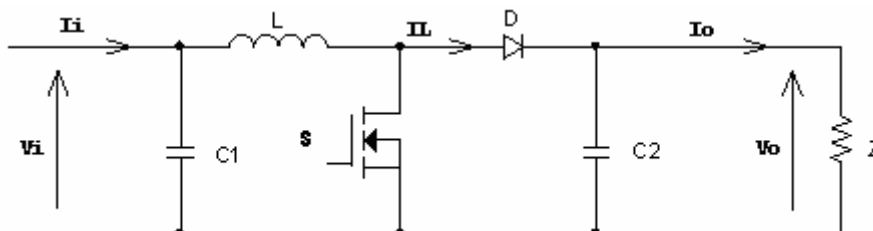


Figure I.12. Circuit électrique du convertisseur Boost.

I.9.2.1. Modèle mathématique équivalent

Comme pour le circuit Buck, l'application des lois de Kirchhoff sur les deux circuits équivalents des deux phases de fonctionnement donne :

Pour la première période DT_s :

$$\begin{cases} i_{C1} = C_1 \frac{dvi}{dt} = i_i - i_L \\ i_{C2} = C_2 \frac{dv_o}{dt} = -i_o \\ v_L = L \frac{di_L}{dt} = v_i - R_L i_L \end{cases} \quad (I.36)$$

Pour la deuxième période $(1-D)T_s$:

$$\begin{cases} i_{C1} = C_1 \frac{dv_i}{dt} = i_i - i_L \\ i_{C2} = C_2 \frac{dv_o}{dt} = i_L - i_o \\ v_L = L \frac{di_L}{dt} = v_i - v_o - R_L i_L \end{cases} \quad (I.37)$$

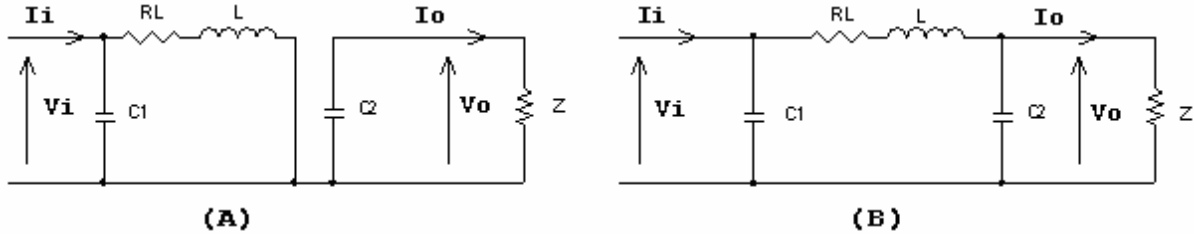


Figure 1.13 Circuits équivalents du convertisseur Boost, (A) : S fermé, (B) : S ouvert.

I.9.2.2. Modèle approximé du convertisseur Boost

Comme pour le Buck, en appliquant la relation (I.19) sur les systèmes d'équations (I.36) et (I.37), on trouve le modèle approximé du convertisseur Boost :

$$\begin{cases} C_1 \frac{dv_i}{dt} T_s = DT_s(i_i - i_L) + (1-D)T_s(i_i - i_L) \\ C_2 \frac{dv_o}{dt} T_s = -DT_s i_o + (1-D)T_s(i_L - i_o) \\ L \frac{di_L}{dt} T_s = DT_s(v_i - R_L i_L) + (1-D)T_s(v_i - v_o - R_L i_L) \end{cases} \quad (I.38)$$

En arrangeant les termes des équations précédentes, (pour qu'on puisse interconnecter le Boost avec les autres blocs de simulation), on obtient la modélisation dynamique du convertisseur Boost :

$$\begin{cases} i_L = i_i - C_1 \frac{dv_i}{dt} \\ i_o = (1-D)i_L - C_2 \frac{dv_o}{dt} \\ v_i = (1-D)v_o + R_L i_L + L \frac{di_L}{dt} \end{cases} \quad (I.39)$$

I.9.2.3. Les ondulations des courants et des tensions

Comme pour le circuit Buck la pente de courant I_L et des tensions V_{c1} et V_{c2} pendant la première période de fonctionnement est donnée par :

$$\begin{cases} \frac{di_L}{dt} \approx \frac{v_L}{L} = \frac{V_i - R_L I_L}{L} \\ \frac{dv_{C1}}{dt} \approx \frac{i_{C1}}{C_1} = \frac{I_i - I_L}{C_1} \\ \frac{dv_{C2}}{dt} \approx \frac{i_{C2}}{C_2} = \frac{-I_o}{C_2} \end{cases} \quad (I.40)$$

Les valeurs crête à crête des courants et des tensions sont :

$$\begin{cases} I_{Lcc} = 2\Delta I_L = \frac{V_i - R_L I_L}{L} DTs \\ V_{icc} = 2\Delta V_i = \frac{I_i - I_L}{C_1} DTs \\ V_{occ} = 2\Delta V_o = \frac{-I_o}{C_2} DTs \end{cases} \quad (I.41)$$

Les valeurs des composants à choisir pour des ondulations données sont :

$$\begin{cases} L = \frac{V_i - R_L I_L}{2\Delta I_L} DTs \\ C_1 = \frac{I_i - I_L}{2\Delta V_i} DTs \\ C_2 = \frac{-I_o}{2\Delta V_o} DTs \end{cases} \quad (I.42)$$

I.9.2.4. Etude en régime continu

Comme pour le circuit Buck, on peut remplacer les signaux du convertisseur par leurs grandeurs moyennes par l'égalisation des dérivées des signaux par la valeur zéro, cela simplifiera ainsi le système d'équations I.39 :

$$\begin{cases} I_i = I_L \\ I_o = (1-D)I_L \\ V_i = (1-D)V_o + R_L I_L \end{cases} \quad (I.43)$$

Rapport de conversion et rendement

En utilisant les relations (I.37), on peut calculer le rapport de conversion V_o/V_i :

$$M(D) = \frac{V_o}{V_i} = \frac{1}{(1-D) + \frac{R_L I_L}{V_o}} = \frac{1}{1 + \frac{R_L I_o}{(1-D)V_o}} \frac{1}{1-D} = \frac{1}{1 + \frac{R_L}{(1-D)^2 Z}} \frac{1}{1-D} = \eta \frac{1}{1-D} \quad (I.44)$$

On remarque que le rendement η ne dépend pas seulement de la charge Z du convertisseur et des résistances parasites des composants, mais dépend aussi du rapport cyclique D . Ainsi il est recommandé pour que le Boost travaille avec un bon rendement de ne pas dépasser certaines valeurs de rapports cycliques D , fixés par la qualité de l'inductance et la charge utilisée (Figure I.15).

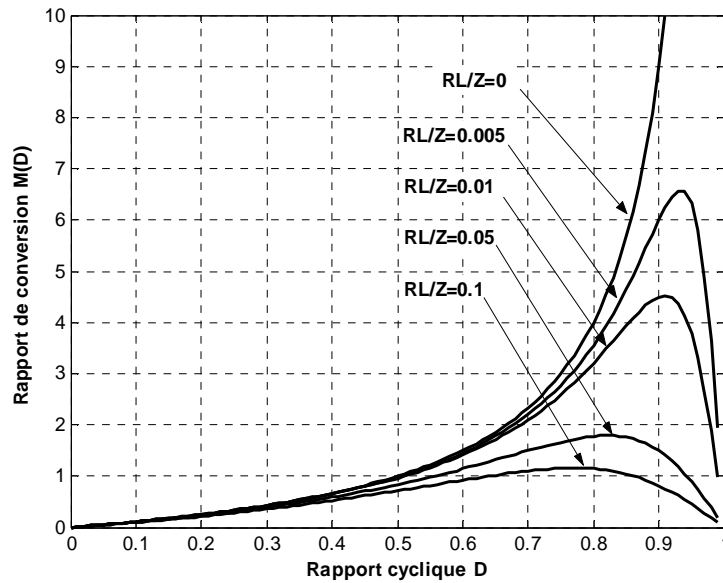


Figure I.14. Rapport de conversion $M(D) = V_o/V_i$, en fonction de R_L/Z .

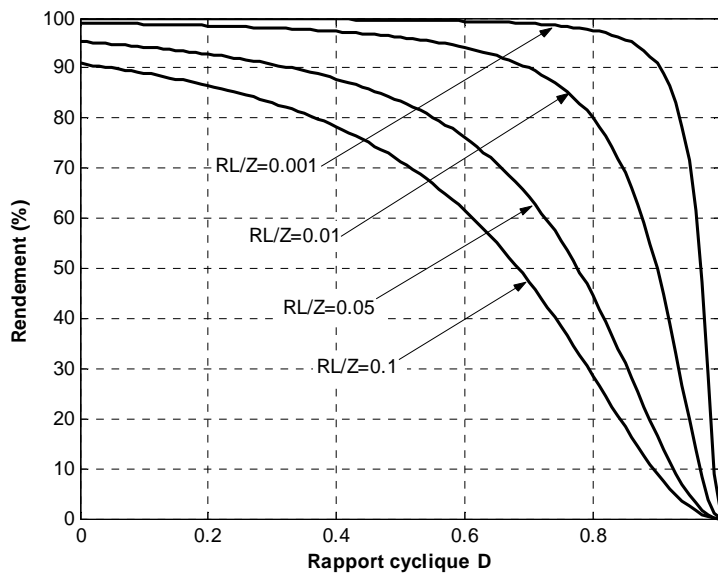


Figure 1.15. Le rendement $\eta(D)$ d'un circuit Boost en fonction de R_L/Z .

I.9.3. Convertisseur Buck-Boost

Le convertisseur Buck-Boost combine les propriétés des deux convertisseurs précédents. Il est utilisé comme un transformateur idéal de n'importe quelle tension d'entrée pour n'importe quelle tension de sortie désirée [1].

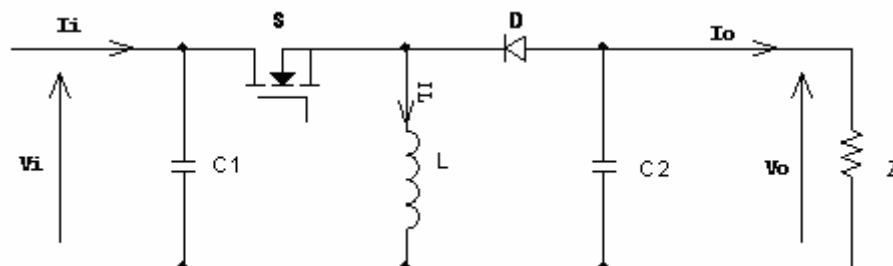


Figure I.16. Circuit électrique d'un convertisseur Buck-Boost.

En premier temps, S est fermé la tension de la source est appliquée aux bornes de l'inductance L, où elle se charge d'énergie jusqu'au début de la deuxième phase de fonctionnement, puis S s'ouvre et la tension de l'inductance se trouve appliquée à la charge, où son courant circule dans le sens inverse d'une aiguilles de montre à travers la diode D et ainsi la tension de sortie sera négative, (Figure I.16).

I.9.3.1. Modèle Mathématique équivalent

La figure I.17 montre les deux schémas équivalents du convertisseur Buck-Boost pour les deux périodes de fonctionnement.

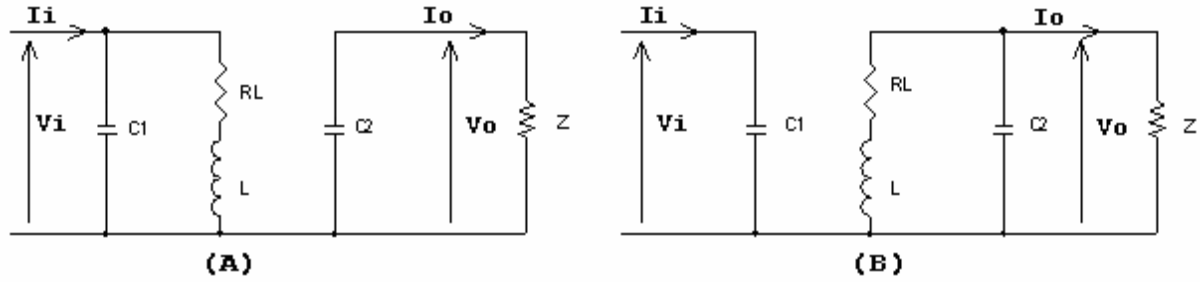


Figure I.17 Circuits équivalents de Buck-Boost, (A) :S fermé, (B) :S ouvert.

En appliquant les lois de Kirchhoff sur les circuits équivalents précédents, on obtient :

Pour la première période DT_s :

$$\begin{cases} i_{C1} = C_1 \frac{dvi}{dt} = i_i - i_L \\ i_{C2} = C_2 \frac{dv_o}{dt} = -i_o \\ v_L = L \frac{di_L}{dt} = v_i - R_L i_L \end{cases} \quad (I.45)$$

Et pour la deuxième période $(1-D)T_s$:

$$\begin{cases} i_{C1} = C_1 \frac{dvi}{dt} = i_i \\ i_{C2} = C_2 \frac{dv_o}{dt} = -i_o - i_L \\ v_L = L \frac{di_L}{dt} = v_o - R_L i_L \end{cases} \quad (I.46)$$

En appliquant la relation (I.19) sur les systèmes d'équations (I.45) et (I.46), on trouve le modèle approximé du convertisseur Buck-Boost :

$$\begin{cases} C_1 \frac{dvi}{dt} T_s = DT_s(i_i - i_L) + (1-D)T_s i_i \\ C_2 \frac{dv_o}{dt} T_s = -DT_s i_o + (1-D)T_s (-i_o - i_L) \\ L \frac{di_L}{dt} T_s = DT_s(v_i - R_L i_L) + (1-D)T_s (v_o - R_L i_L) \end{cases} \quad (I.47)$$

En arrangeant les termes des équations précédentes, on obtient la modélisation dynamique du Buck Boost :

$$\begin{cases} i_L = \frac{1}{D} \left[ii - C1 \frac{dvi}{dt} \right] \\ i_o = -(1-D)i_L - C2 \frac{dv_o}{dt} \\ v_i = \frac{1}{D} \left[-(1-D)v_o + R_L i_L + L \frac{di_L}{dt} \right] \end{cases} \quad (I.48)$$

I.9.3.2. Les ondulations des courants et des tensions

En suivant les mêmes procédures précédentes, on trouve les mêmes résultats que pour le circuit Boost, les valeurs crête à crête des courants et des tensions sont :

$$\begin{cases} I_{LCC} = 2\Delta I_L = \frac{Vi - R_L I_L}{L} DTs \\ V_{iCC} = 2\Delta V_{c1} = \frac{Ii - I_L}{C_1} DTs \\ V_{oCC} = 2\Delta V_{c2} = -\frac{I_o}{C_2} DTs \end{cases} \quad (I.49)$$

Les valeurs des composants à choisir pour des ondulations données sont :

$$\begin{cases} L = \frac{Vi - R_L I_L}{2\Delta I_L} DTs \\ C_1 = \frac{Ii - I_L}{2\Delta Vi} \\ C_2 = -\frac{I_o}{2\Delta V_o} DTs \end{cases} \quad (I.50)$$

I.10. Méthodes de poursuite MPPT

Afin de faire la poursuite du P.P.M du système photovoltaïque, et comme on l'a déjà vu dans la section I.9 la tension de sortie du système photovoltaïque est régulé, par le convertisseur DC/DC qui lui-même est contrôlé continuellement par son rapport cyclique. Il existe plusieurs méthodes de commande pour la poursuite MPPT, on les cite brièvement dans la section suivante :

Philosophie du MPPT

Presque la plupart des méthodes de poursuite du PPM reposent sur la caractéristique P-V de la cellule PV (voir figure I.18).

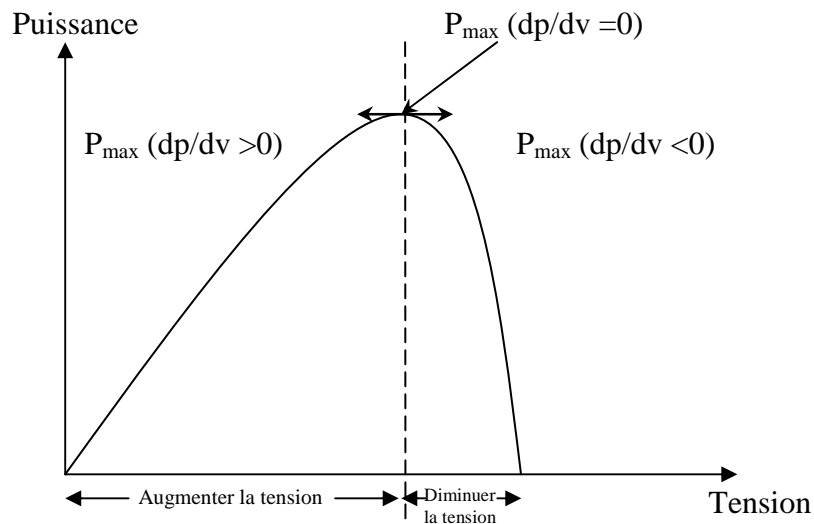


Figure I.18. Caractéristique P-V d'une cellule PV.

La caractéristique P-V est découpée en deux demi plans :

- ✓ Le demi plan gauche ou la dérivée $dp/dv > 0$: Si on est dans cette zone donc on essaye d'augmenter la tension pour aller vers le PPM et cela se fait par l'incrémentement du rapport cyclique D ce qui veut dire que : $D(k) = D(k-1) + C$ (avec C : le pas d'incrémentement),
- ✓ Le demi plan droit ou la dérivée $dp/dv < 0$: Si on est dans cette zone donc on essaye de diminuer la tension pour aller vers le PPM et cela se fait par la désincrémentement du rapport cyclique D ce qui veut dire que : $D(k) = D(k-1) - C$.

Dans la section suivante, on cite plusieurs méthodes qui fonctionnent suivant cette philosophie.

I.10.1. Algorithme de contrôle MPPT adaptatif

Le principe de cette méthode a été proposé par A.F. Boeringer [4]. L'algorithme est décrit comme illustré sur la figure (I.19) :

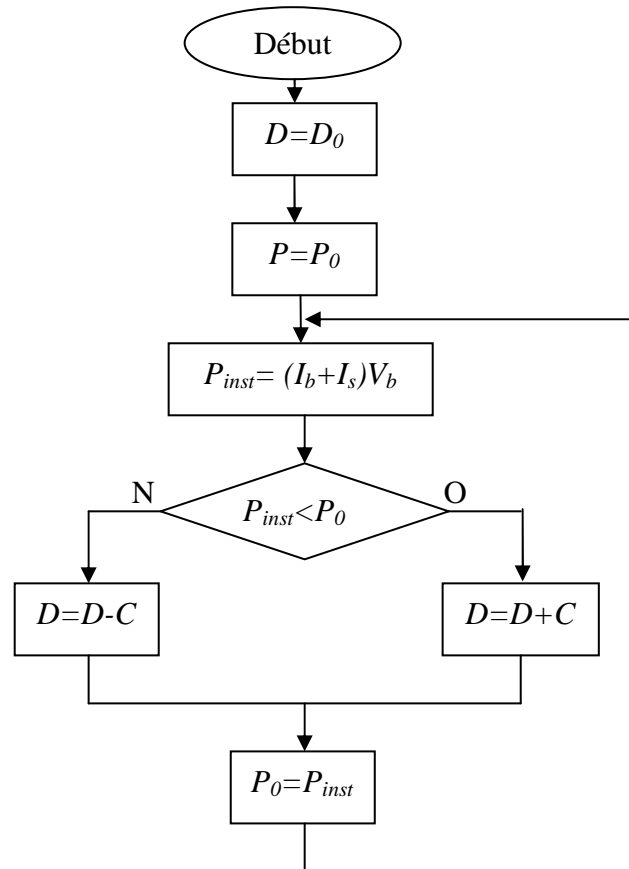


Figure I.19. Principe classique d'une méthode MPPT.

Comme illustré sur le schéma de la figure (I.19), le système démarre avec un rapport cyclique initiale D_0 et une puissance initiale P_0 , ensuite on mesure la puissance instantanée du générateur PV, et puis on la compare avec P_0 . Si P_{inst} est inférieure à P_0 alors D est incrémenté, sinon D est désincrémenté. Et enfin P_0 prend la valeur P_{inst} .

I.10.2. Algorithme Perturbation et Observation simple (P and O simple)

La figure I.20 donne l'organigramme de l'algorithme P&O.

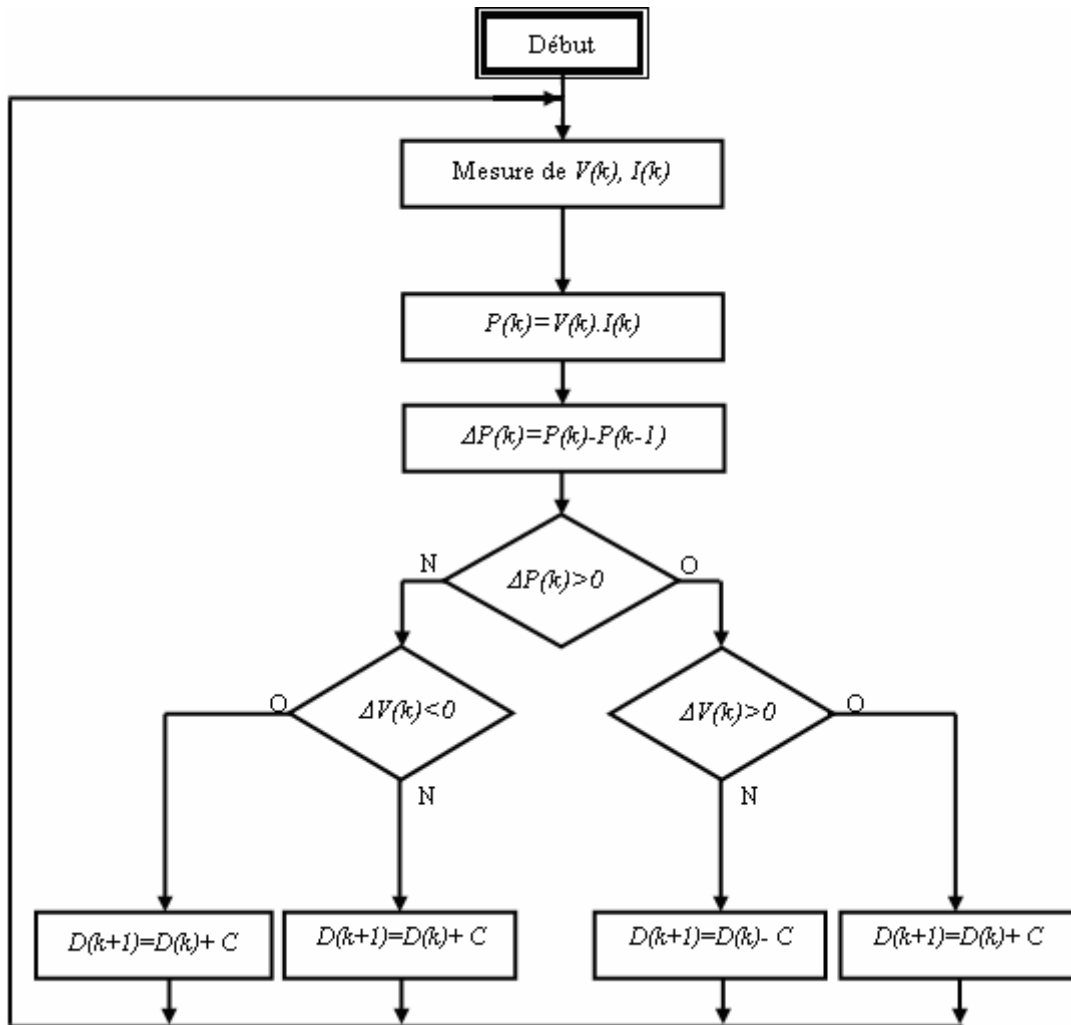


Figure I.20. Organigramme de l'algorithme Perturbation et Observation (P and O).

L'algorithme P&O fonctionne comme suit :

D'abord on calcule la puissance $P(k)$ et on la compare avec la valeur $P(k-1)$. Ensuite on effectue un test sur le $\Delta V(k)$; s'il est positif par exemple donc on est dans le plan gauche de la caractéristique P-V ce qui veut dire qu'on doit augmenter la tension donc on incrémente le rapport cyclique, et ainsi de suite...

I.10.3. Algorithme de l'incrément de conductance

La dérivé partielle $\frac{dP}{dV}$ est donnée par :

$$\frac{dP}{dV} = I + V \frac{dI}{dV} \quad (I.51)$$

$$\frac{1}{V} \frac{dP}{dV} = \frac{I}{V} + \frac{dI}{dV} \quad (I.52)$$

On définit la conductance de la source $G = \frac{I}{V}$ et l'incrémentale conductance $\Delta G = \frac{dI}{dV}$.

Puisque la tension V du panneau est toujours positive, la relation (I.52) explique que le point de puissance maximale MPP est atteint si la conductance de la source G égale l'incrémentale conductance ΔG de la source avec un signe moins, et qu'elle est à gauche de ce point lorsque

la conductance G est supérieure à l'incrémentale conductance ΔG et vice-versa, [2] comme suit :

$$\begin{cases} \frac{dP}{dV} > 0 & \text{si } \frac{I}{V} > -\frac{dI}{dV} \\ \frac{dP}{dV} = 0 & \text{si } \frac{I}{V} = -\frac{dI}{dV} \\ \frac{dP}{dV} < 0 & \text{si } \frac{I}{V} < -\frac{dI}{dV} \end{cases} \quad (\text{I.53})$$

Pour éviter que la différentielle dV de la tension du panneau ne devienne nulle, lorsque le MPP est atteint dans les précédents cycles ou par une stabilisation du point de fonctionnement, ce qui conduirait à une division par zéro, l'algorithme fait un test de dV . S'il est nul il teste si le $dI=0$ si c'est le cas alors le PPM est atteint et l'algorithme ne fait aucun changement, dans le cas contraire l'algorithme teste le signe de dI pour déterminer la position du PPM et ajuste la tension de référence correspondante. La figure I.21 donne l'organigramme de cet algorithme.

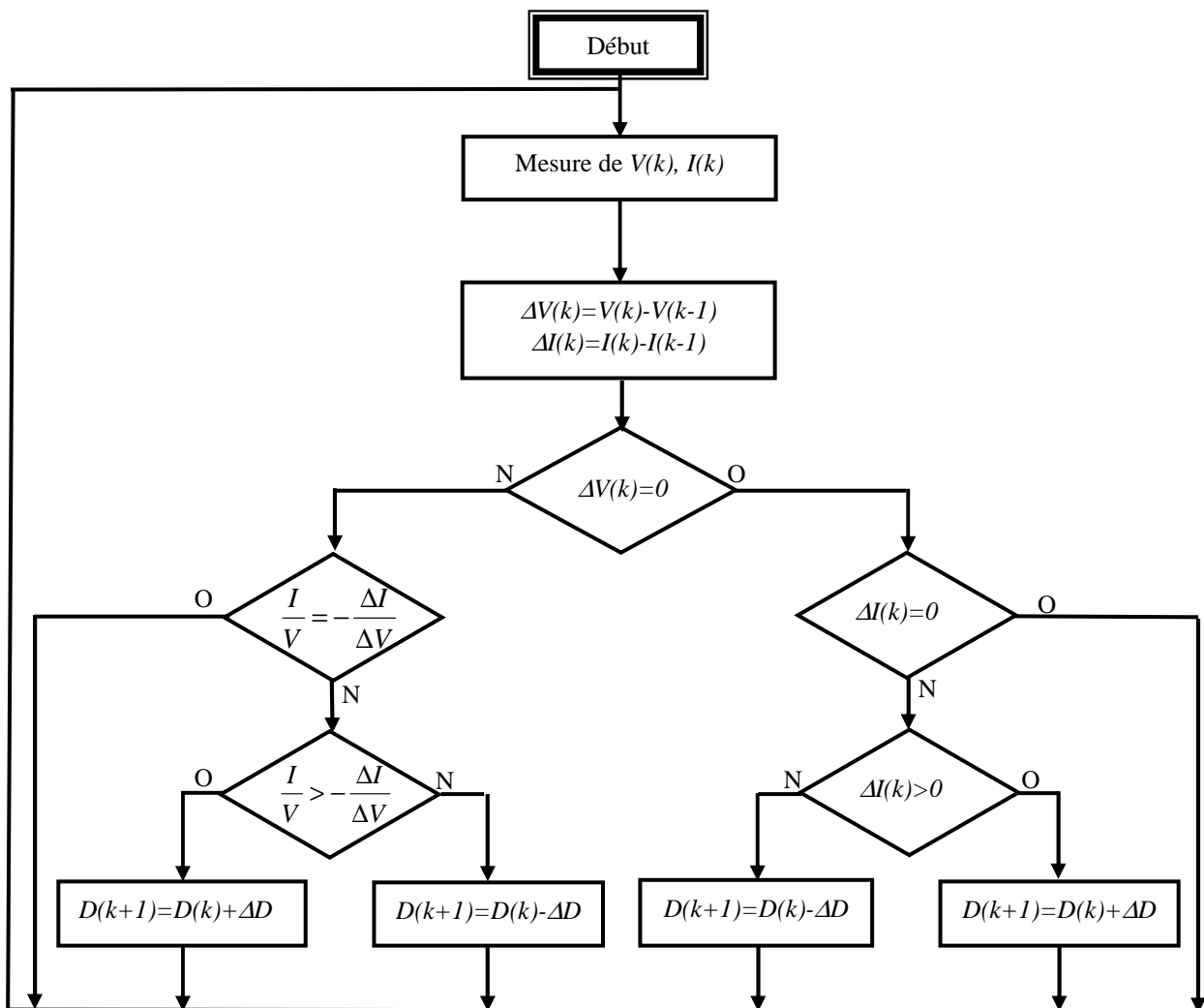


Figure I.21. Organigramme de l'algorithme incrémentation de conductance.

L'avantage de cet algorithme est qu'il n'oscille pas autour du MPP, à cause du test de $dI=0$. L'algorithme se stabilise une fois le PPM atteint et le signe de dI donne la vraie direction à emprunter pour la poursuite du PPM une fois le système stable, ce qui conduit à une réponse rapide du système pour des changements brusques des conditions atmosphériques.

Mais à cause de l'approximation des dérivées dV et dI et l'utilisation d'un pas d'action constant, la condition $\frac{I}{V} = -\frac{dI}{dV}$ est rarement vraie ce qui conduit à des oscillations autour du PPM. Pour remédier à ce problème une erreur marginale ε est ajoutée aux conditions du MPP, c'est à dire que le MPP est atteint si $\left|\frac{I}{V} + \frac{dI}{dV}\right| \leq \varepsilon$, la valeur de ε est limitée entre le problème de non fonctionnement dans le MPP exact et le problème de l'amplitude des oscillations [2].

I.10.4. Méthodes MPPT par contre réaction de tension

Comme il est montré sur la figure I.22, ces méthodes de poursuite sont basées sur la comparaison de la tension du panneau avec une tension de référence, pour ajuster le rapport cyclique de la MLI de commande afin d'annuler l'erreur.

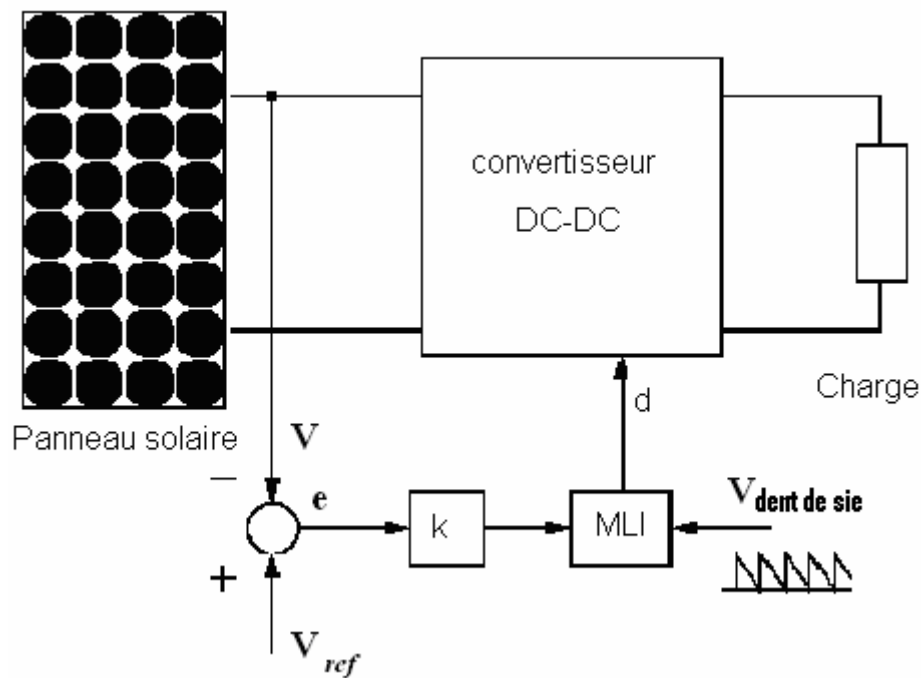


Figure I.22. Méthode avec Contre-réaction de tension avec MLI.

La tension de référence V_{ref} est calculée par plusieurs méthodes :

- ✓ Méthode de calcul de la tension de référence fixe $V_{ref}=V_{moy}$

V_{ref} est égale à la tension moyenne des tensions maximales de la caractéristique P-V :

$$V_{ref}=V_{moy} \quad (I.54)$$

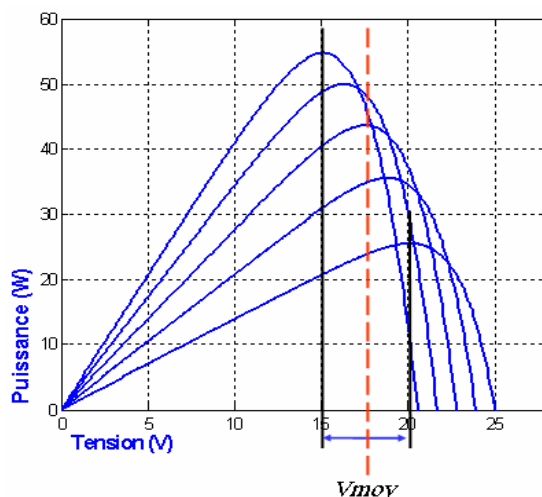


Figure I.23. Intervalle de variation de la tension de fonctionnement optimale.

✓ **Méthode de calcul de la tension de référence en fonction de V_{oc}**

La tension maximale du panneau est exprimée comme une fonction linéaire de la tension en circuit ouvert du panneau. $V_{mp} = kV_{oc}$, où le ($k=0.77$) [7], et le V_{oc} est prélevée soit :

- régulièrement par le débranchement du panneau pour une courte durée de temps,
- ou par la tension de circuit ouvert d'une cellule pilote (c'est une cellule ajoutée au panneau, elle qui est indépendante électriquement).

I.11. Conclusion

Cet état de l'art préliminaire a permis de présenter les points qu'il faut améliorer sur les systèmes de conversion photovoltaïque (PV). En effet, étant donnée la recrudescence d'intérêts sur les énergies renouvelables, il est nécessaire, si l'on souhaite que le PV connaisse l'essor espéré dans l'avenir, d'améliorer et de sécuriser ce type de conversion d'énergie.

Tout le long de ce travail, il est à noter qu'un effort important dans le domaine commence à se faire sentir, notamment du point de vue des publications scientifiques. Aujourd'hui, étant donnée la maturité des travaux dans ce domaine, un effort vers l'industrialisation dans un futur proche pourra être fait sur la base des derniers résultats de recherche.

CHAPITRE II

LA COMMANDE PAR LOGIQUE FLOUE

II.1. Introduction

La logique floue est un axe de recherche important sur lequel sont focalisés de nombreux scientifiques, car elle utilise la méthodologie du raisonnement humain, à la résolution des problèmes sans avoir besoin d'une modélisation mathématique.

En 1965 le professeur L.Zadeh a fondé les bases théoriques de la logique floue. L'application de cette théorie sur la régulation n'a eu lieu qu'à partir des années soixante dix [13]. A partir de 1985 la logique floue est devenue une actualité scientifique pour un large public chercheur.

Dans la théorie des ensembles classique, un élément ne peut avoir que deux états différents $\{0,1\}$ alors qu'en logique floue, il peut admettre les deux propositions avec un degré de vérité pour chacune.

L'idée est de remplacer l'ensemble binaire $\{0,1\}$ par un intervalle $[0, 1]$ ceci permet des graduations dans l'appartenance d'un élément à une situation, ce qui permet la modélisation de l'observation humaine exprimée par des expressions linguistiques [13].

II.2. Terminologie de la logique floue

II.2.1. Variable linguistique

La description d'une certaine situation imprécise ou incertaine peut contenir des expressions floues comme : très grand, grand, moyen, petit, ...etc. Ces expressions forment les valeurs d'une variable x , appelée "linguistique", soumise à des fonctions appelées les fonctions d'appartenance.

II.2.2. Fonctions d'appartenance

La variable x varie dans un domaine appelé univers de discours, ce dernier est partagé en sous-ensembles flous à l'aide de la théorie des ensembles flous, et l'expertise humaine, de façon que dans chaque zone il y a une situation dominante. Ces zones sont décrites par des fonctions convexes sous forme triangulaire, trapézoïdale, ...etc., admettant comme argument la position de la variable x dans l'univers de discours, et comme sortie le degré d'appartenance de x à la situation décrite par la fonction ; notée :

$\mu_E(x)$: degré d'appartenance de x à l'ensemble E .

Pour un choix judicieux, le chevauchement entre deux fonctions d'appartenance, doit être entre zéro et la mi-hauteur [14].

Quelques formes des fonctions d'appartenance sont illustrées sur la Figure II.1.

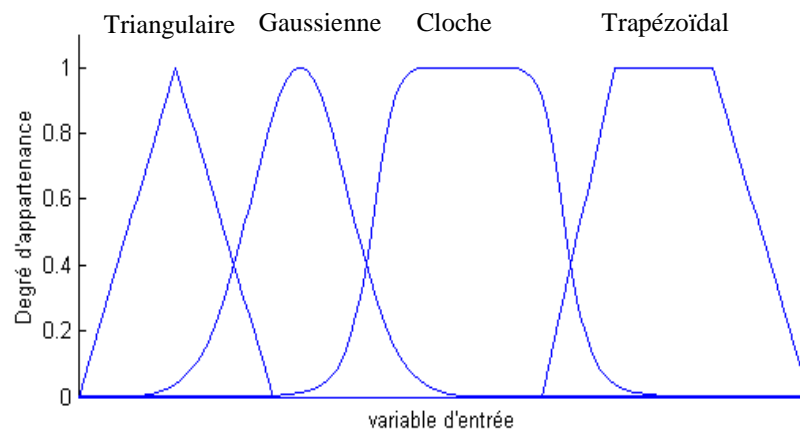


Figure II.1. Exemple de fonctions d'appartenance

II.2.3. Opérateurs de la logique floue

La description d'une situation où il y a plus qu'une variable qui interviennent, nécessite l'utilisation des opérateurs logiques tels que : "ET", "OU", et "NON".

Dans la théorie de la logique floue l'opérateur "ET" correspond à l'opération "Minimum", "OU" à l'opération "Maximum", et "NON" au complément à un.

L'analogie d'utilisation de ces opérateurs dans les deux logiques classique et floue est sur le Tableau II.1.

	Logique classique	Logique floue
$C = A \text{ ET } B$	$C = A \cap B$	$\mu_C(x) = \text{Min}(\mu_A(x), \mu_B(x))$
$C = A \text{ OU } B$	$C = A \cup B$	$\mu_C(x) = \text{Max}(\mu_A(x), \mu_B(x))$
$C = \text{NON } A$	$C = \bar{A}$	$\mu_C(x) = 1 - \mu_A(x)$

Tableau II.1. Application des opérateurs dans les deux ensembles.

Avec : A, B, C : ensembles

II.3. Structure de base d'un contrôleur flou

Chaque contrôleur flou comporte les trois blocs suivants : Fuzzification, inférence et defuzzification, la structure de base est donnée par la Figure II.2 [13].

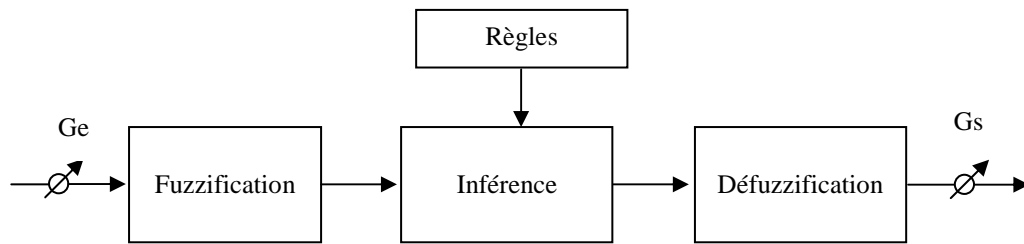


Figure II.2. Structure de base d'un contrôleur flou.

G_e : Gain d'entrée, G_s : Gain de sortie.

Nous exposons en détails, les différentes parties du régulateur flou dans la section suivante :

II.3.1. Fuzzification

La fuzzification proprement dite consiste à définir des fonctions d'appartenances pour les différentes variables, réalisant le passage d'une grandeur physique à une grandeur linguistique [13].

Exemple II.1 : soit une grandeur x (appartenant à $[-1, 1]$) définie par 5 sous-ensemble flous NG (négatif grand), NM (négatif moyen), EZ (environ zéro), PM (positif moyen) et PG (positif grand).

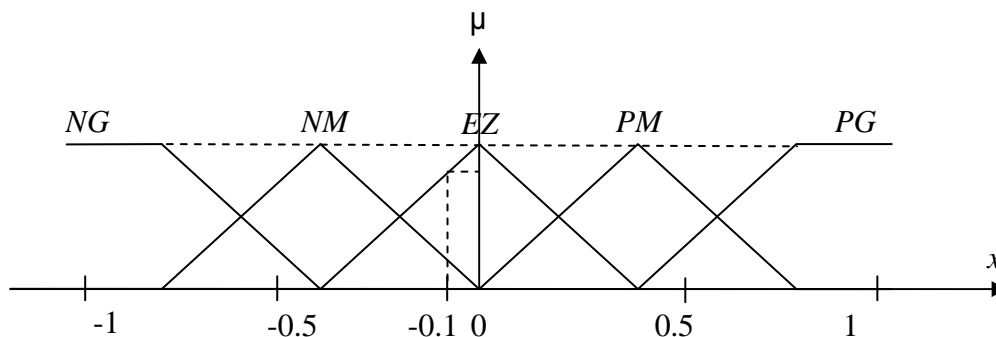


Figure II.3. Fuzzification avec cinq fonctions d'appartenance.

Donc à chaque variable linguistique d'entrée x , on fait correspondre une valeur linguistique (Négatif Grand, Négatif Moyen,...) avec un degré d'appartenance.

- Pour $x=0.5$, on associe $\mu_{PM}(0.5) = 0.75$ et $\mu_{PG}(0.5) = 0.25$
- Pour $x=-0.1$, on associe $\mu_{EZ}(-0.1) = 0.9$ et $\mu_{NM}(-0.1) = 0.1$.

II.3.2. Les règles floues

Une règle floue peut être définie comme étant l'ensemble d'implications. Elle exprime le lien entre les variables linguistiques d'entrée et celles de la sortie.

La forme typique d'une règle floue est :

Si <Conditions liées entre eux par des opérateurs flous> **Alors** <Action ou Description> (II.1)

La première partie (entrée de la règle) teste la validité de certaines conditions, et la deuxième (sortie de la règle) pour exprimer la réaction souhaitée ou la description adaptée en fonction de la validité de la première.

Il existe deux types de règles : règles de Mamdani (M) et règles de Takagi Sugeno (TS) [12].

II.3.2.1. Les règles de Mamdani

La forme typique d'une règle Mamdani s'écrit :

Si x_1 est E_1 (et) x_2 est E_2 (et)...(et) x_m est E_m **Alors:** u_1 est S_1 , u_2 est S_2 , ..., u_n est S_n (II.2)

Où : x_1, \dots, x_m : les variables d'entrées

u_1, \dots, u_n : les variables de sorties

$E_1, \dots, E_m, S_1, \dots, S_n$: ensembles flous

(*est*) : signifier l'appartenance

II.3.2.2. Les règles de Takagi Sugeno

La première partie d'une règle de type Takagi Sugeno est similaire à celle de Mamdani tandis que la deuxième est une fonctionnelle.

La forme typique de cette règle s'écrit donc :

Si x_1 est E_1 (et) x_2 est E_2 (et)... (et) x_m est E_m : **Alors** : $u_1 = f_1(x_1, \dots, x_m)$, $u_2 = f_2(x_1, \dots, x_m)$, ..., $u_n = f_n(x_1, \dots, x_m)$ (II.3)

Où f_1, \dots, f_n : fonctions réelles, théoriquement peuvent être linéaires ou non linéaires mais l'implémentation de la méthode exige qu'elles doivent être des fonctions linéaires.

II.3.3. Inférence

L'inférence est parfois appelée raisonnement flou, elle est utilisée dans la règle floue pour déterminer le résultat de cette dernière pour des valeurs données aux variables d'entrée.

Cette partie décisionnelle où le contrôleur décide l'action similaire à celle d'un opérateur humain, elle présente et englobe toutes les connaissances de l'expert humain sur la dynamique, et les caractéristiques du système.

La description des inférences peut se faire par des expressions linguistiques, symboliques, ou par une matrice d'inférence.

Exemple II.2 : Soit deux entrées x_1 et x_2 et une sortie x_r , toutes trois définies par les 5 sous-ensembles de l'exemple précédent.

On peut avoir une description d'une base de règles possible comme suit :

		x_1				
		NG	NM	EZ	PM	PG
x_2	NG			PG	PM	
	NM			PM	EZ	NM
	EZ	PG	PM	EZ	NM	NG
	PM	PM	EZ	NM		
	PG		NM	NG		

Tableau II.2. Matrice d'inférence pour deux variables linguistiques x_1 et x_2 et une variable de sortie x_r .

On peut aussi exprimé les règles du tableau sous forme de :

Si (x_1 **NG** et x_2 et **EZ**), alors x_r **PG** **ou**

Si (x_1 **NG** et x_2 et **PM**), alors x_r **PM** **ou**

...

Si (x_1 **EZ** et x_2 et **PG**), alors x_r **NG**.

Il faut noter qu'on n'est pas obligé de compléter toute la table. Les règles sont élaborées par un expert et sa connaissance du problème. Si ce dernier estime qu'il n'est pas nécessaire de remplir la table, c'est qu'il sait que les cas non considérés n'interviendront pas lors de la mise en application [12].

Selon la fonction qu'on donne au "**Alors**" on trouve plusieurs méthodes de calculs de l'inférence, parmi ces méthodes [13] :

- La méthode de Mamdani (Rm)
- La méthode de produit de Larsen (Rl)
- La méthode produit énergétique (Rpe)
- La méthode de produit limité (Rpl)

Les définitions de ces méthodes sont données au tableau II.3.

Où :

$\mu_S(u)$: la fonction d'appartenance correspond à l'ensemble flou S de la sortie de la règle (III.2),

μ : le degré d'appartenance combiné dans la première partie de la règle (III.2), u : la variable de sortie de la règle (III.2).

Méthode d'inférence	Définition
Inférence de Mamdani (Rm)	$\min(\mu, \mu_S(u))$ pour tout μ
Inférence du produit de Larsen (RI)	$\mu \cdot \mu_S(u)$, pour tout μ
Inférence du produit énergétique (Rpe)	$\begin{cases} \mu, & \text{pour } \mu_S(u) = 1 \\ \mu_S(u), & \text{pour } \mu = 1 \\ 0, & \text{pour } \mu < 1 \text{ et } \mu_S(u) < 1 \end{cases}$
Inférence du produit limité (Rpl)	$\max(\mu + \mu_S(u) - 1, 0)$

Tableau II.3. Définitions des méthodes d'inférence.

II.3.4. Defuzzification

Cette étape sert à convertir la conclusion de l'étage d'inférence d'une variable linguistique à une variable physique [11]. Plusieurs méthodes ont été proposées, parmi ces méthodes on distingue les quatre méthodes de défuzzification suivantes:

II.3.4.1. Méthode du maximum

La sortie correspond à l'abscisse du maximum de la fonction d'appartenance résultante.

Trois cas peuvent produire :

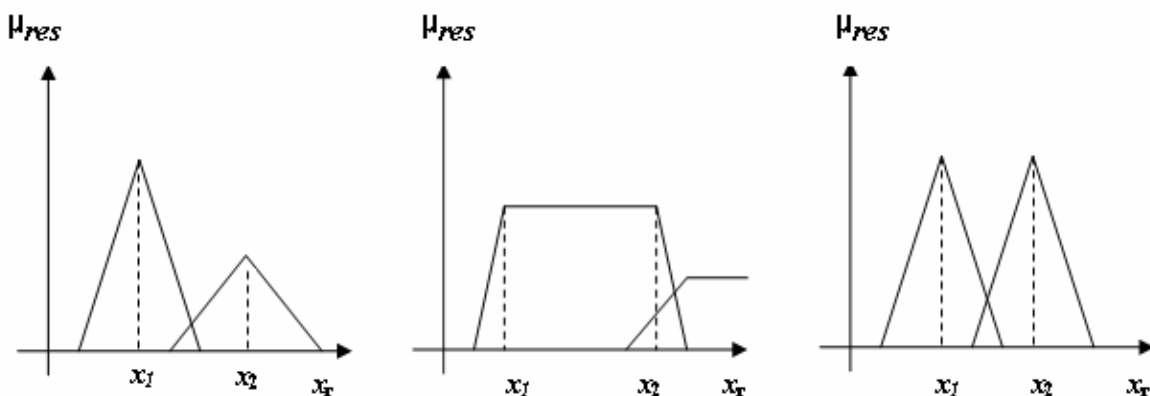


Figure II.4. Défuzzification par valeur maximum.

- Dans le premier cas il n'y a pas de problèmes (car le maximum est x_1).
- Dans les deux autres cas, une ambiguïté apparaît. Il n'y a pas de règle générale sur la décision à prendre. Certains opérateurs préféreront prendre la plus petite sortie, d'autres

la plus grande et d'autres une valeur entre x_1 et x_2 (uniquement pour le deuxième cas de la figure II.8).

En somme, la méthode est simple, rapide et facile mais elle introduit des ambiguïtés et une discontinuité de la sortie.

II.3.4.2. Méthode de la moyenne des maxima

Dans le cas où plusieurs sous-ensembles auraient la même hauteur maximale, on réalise la moyenne.

II.3.4.3. Méthode du centroïde

La sortie correspond à l'abscisse du centre de gravité de la surface de la fonction d'appartenance résultante.

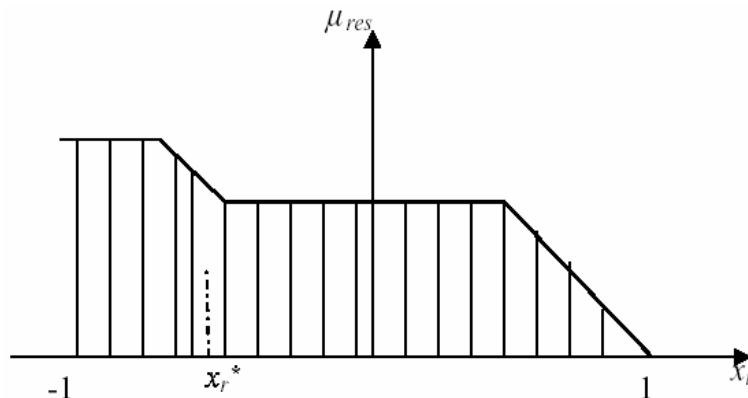


Figure II.5. Défuzzification par la méthode du centroïde.

Il existe deux méthodes :

1. On prend l'union des sous-ensembles flous de sortie et on tire le centroïde global (calculs très lourds).
2. On prend chaque sous-ensemble séparément et on calcul son centroïde, puis on réalise la moyenne des de tous les centroïde.

La défuzzification peut se faire à l'aide de la relation suivante :

$$x_R^* = \frac{\int_{-1}^1 x_R \mu_{Res} dx_R}{\int_{-1}^1 \mu_{Res} dx_R} \quad (\text{II.4})$$

En conclusion, on n'a plus de discontinuités et d'ambiguïtés, mais cette méthode est plus complexe et demande des calculs plus importants.

II.3.4.4. Méthode de la somme pondérée

Il s'agit d'un compromis entre les deux méthodes précédentes. On calcule individuellement les sorties relatives à chaque règle selon le principe de la moyenne des

maxima, puis on réalise leur moyenne pondérée [11].

II.4. Avantages et désavantages du réglage par logique floue

II.4.1. Avantages

Le réglage par logique floue présente les avantages suivants :

- La non nécessité de la modélisation du système.
- La possibilité d'implémenter des connaissances linguistiques de l'opérateur du processus.
- La maîtrise du système à régler avec un comportement complexe (systèmes non linéaire et difficile à modéliser).
- La disponibilité de système de développement efficace, soit par microprocesseur ou PC, soit par circuits intégrés.

II.4.2. Désavantages

- Le manque de directives précises pour la conception d'un réglage (choix de grandeur mesurée, détermination de la fuzzification, des inférences et de la défuzzification).
- L'approche artisanale et asymptotique (implantation des connaissances de l'opérateur souvent difficile).
- L'impossibilité de la démonstration de la stabilité du circuit de réglage en toute généralité (en l'absence d'un modèle valable).
- La cohérence des inférences non garanties (possibilité d'apparition de règles d'inférences contradictoires).

II.5. Conclusion

Dans ce chapitre, nous avons présenté la structure interne, la structure externe et le principe du régulateur flou. Plusieurs directives ont été formulées pour un choix judicieux des fonctions d'appartenances, règles d'inférences, méthodes d'inférences et la défuzzification. mais le choix de paramètres reste toujours difficile à effectuer pour garantir de bonnes performances, cela constitue toujours un axe de recherche actif.

CHAPITRE III

LES ALGORITHMES GÉNÉTIQUES

III.1. Introduction

Les algorithmes génétiques sont des algorithmes d'optimisation s'appuyant sur des techniques dérivées de la génétique et de l'évolution naturelle : croisements, mutations, sélection, etc. Les algorithmes génétiques ont déjà une histoire relativement ancienne, puisque les premiers travaux de John Holland sur les systèmes adaptatifs remontent à 1962 [16].

Un algorithme génétique recherche le ou les extrema d'une fonction définie sur un espace de données. Pour l'utiliser, on doit disposer des cinq éléments suivants :

1. Un principe de codage de l'élément de population. Cette étape associe à chacun des points de l'espace d'état une structure de données. Elle se place généralement après une phase de modélisation mathématique du problème traité. Le choix du codage des données conditionne le succès des algorithmes génétiques. Les codages binaires ont été très employés à l'origine [16], [17]. Les codages réels sont désormais largement utilisés, notamment dans les domaines applicatifs, pour l'optimisation de problèmes à variables continues [18], [20].

2. Un mécanisme de génération de la population initiale. Ce mécanisme doit être capable de produire une population d'individus non homogène qui servira de base pour les générations futures. Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global. Dans le cas où l'on ne connaît rien du problème à résoudre, il est essentiel que la population initiale soit répartie sur tout le domaine de recherche.

3. Une fonction à optimiser. Celle-ci prend ses valeurs dans \mathbb{R}^+ et est appelée *fitness* ou fonction d'évaluation de l'individu. Elle est utilisée pour sélectionner et reproduire les meilleurs individus de la population.

4. Des opérateurs permettant de diversifier la population au cours des générations et d'explorer l'espace d'état. L'opérateur de croisement recompose les gènes d'individus existant dans la population, l'opérateur de mutation a pour but de garantir l'exploration de l'espace d'état.

5. Des paramètres de dimensionnement : taille de la population, nombre total de générations ou critère d'arrêt, probabilités d'application des opérateurs de croisement et de mutation.

Le principe général du fonctionnement d'un algorithme génétique est représenté sur la figure III.1.

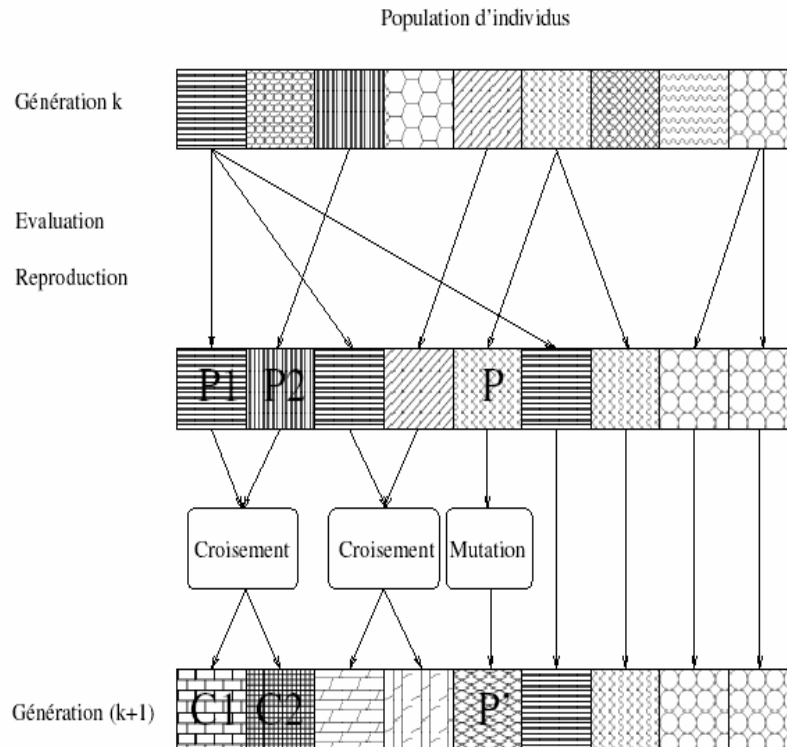


Figure. III.1 Principe général des algorithmes génétiques.

On commence par engendrer une population d'individus de façon aléatoire.

Pour passer d'une génération k à la génération $k+1$, les trois opérations suivantes sont répétées pour tous les éléments de la population k . Des couples de parents $P1$ et $P2$ sont sélectionnés en fonction de leurs adaptations. L'opérateur de croisement leur est appliqué avec une probabilité Pc (généralement autour de 0.6) et engendre des couples d'enfants $C1$ et $C2$. D'autres éléments P sont sélectionnés en fonction de leur adaptation. L'opérateur de mutation leur est appliqué avec la probabilité Pm (Pm est généralement très inférieur à Pc) et engendre des individus mutés P' . Les enfants ($C1, C2$) et les individus mutés P' sont ensuite évalués avant insertion dans la nouvelle population (la figure 2.1 présente le cas où les enfants et les individus mutés remplacent les parents). Différents critères d'arrêt de l'algorithme peuvent être choisis :

- Le nombre de générations que l'on souhaite exécuter peut être fixé a priori. C'est ce que l'on est tenté de faire lorsque l'on doit trouver une solution dans un temps limité.
- L'algorithme peut être arrêté lorsque la population n'évolue plus ou plus suffisamment rapidement.

On peut détailler le fonctionnement des algorithmes génétiques par la figure suivante (figure: III.2)

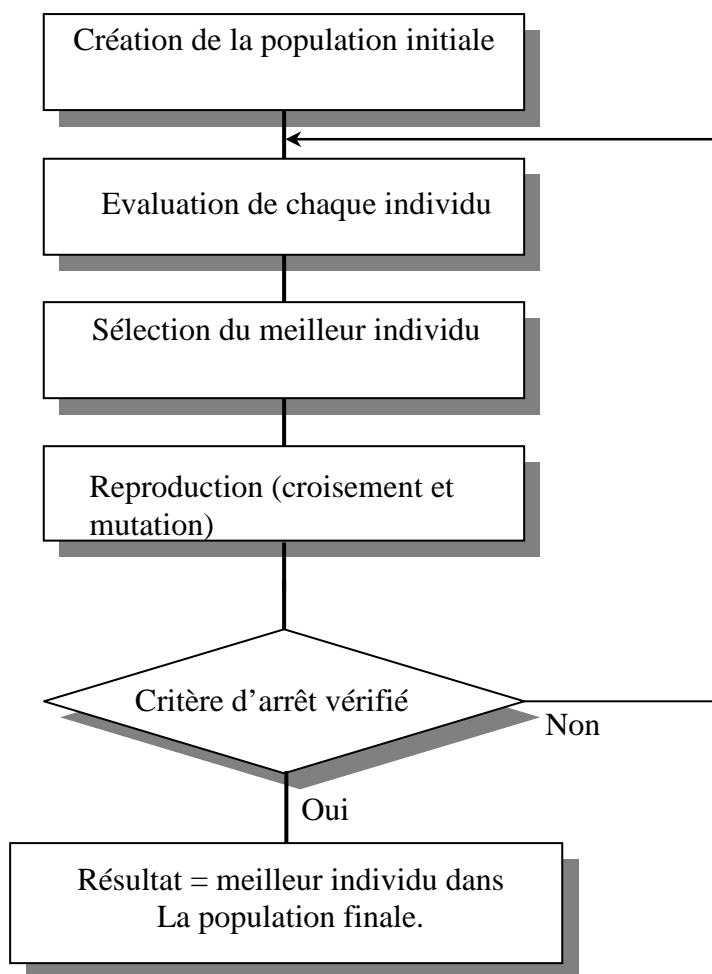


Figure III.2. L'Algorithme Génétique de base.

III.2. Description détaillée

III.2.1. Codage des données

Historiquement, le codage utilisé par les algorithmes génétiques était représenté sous forme de chaînes de bits contenant toute l'information nécessaire à la description d'un point dans l'espace d'état. Ce type de codage a pour intérêt de permettre de créer des opérateurs de croisement et de mutation simples. C'est également en utilisant ce type de codage que les premiers résultats de convergence théorique ont été obtenus.

Cependant, ce type de codage n'est pas toujours bon :

- Pour des problèmes d'optimisation dans des espaces de grande dimension, le codage binaire peut rapidement devenir mauvais. Généralement, chaque variable est représentée par une partie de la chaîne de bits et la structure du problème n'est pas bien reflétée, l'ordre des variables ayant une importance dans la structure du chromosome, alors qu'il n'en a pas forcément dans la structure du problème.

Les algorithmes génétiques utilisant des vecteurs réels [29], [30] évitent ce problème en conservant les variables du problème dans le codage de l'élément de population, sans passer par le codage binaire intermédiaire. Certains les appellent RCGA (*Real Coded Genetic Algorithms*), d'autres parlent d'algorithmes évolutionnaires. La structure du problème est conservée dans le codage.

III.2.2. Génération aléatoire de la population initiale

Le choix de la population initiale d'individus conditionne fortement la rapidité de l'algorithme. Si la position de l'optimum dans l'espace d'état est totalement inconnue, il est naturel d'engendrer aléatoirement des individus en faisant des tirages uniformes dans chacun des domaines associés aux composantes de l'espace d'état, en veillant à ce que les individus produits respectent les contraintes [25]. Si par contre, des informations a priori sur le problème sont disponibles, il paraît bien évidemment naturel d'engendrer les individus dans un sous-domaine particulier afin d'accélérer la convergence. Dans l'hypothèse où la gestion des contraintes ne peut se faire directement, les contraintes sont généralement incluses dans le critère à optimiser sous forme de pénalités.

III.2.3. Gestion des contraintes

Un élément de population qui viole une contrainte se verra attribuer une mauvaise *fitness* et aura une probabilité forte d'être éliminé par le processus de sélection. Il peut cependant être intéressant de conserver, tout en les pénalisant, les éléments non admissibles car ils peuvent permettre de générer des éléments admissibles de bonne qualité. Pour de nombreux problèmes, l'optimum est atteint lorsque l'une au moins des contraintes de séparation est saturée, c'est-à-dire sur la frontière de l'espace admissible.

Gérer les contraintes en pénalisant la fonction *fitness* est difficile, un "dosage" s'impose pour ne pas favoriser la recherche de solutions admissibles au détriment de la recherche de l'optimum ou inversement.

Disposant d'une population d'individus non homogène, la diversité de la population doit être entretenue au cours des générations, afin de parcourir le plus largement possible l'espace d'état. C'est le rôle des opérateurs de croisement et de mutation.

III.2.4. Opérateur de croisement

Le croisement a pour but d'enrichir la diversité de la population en manipulant la structure des chromosomes. Classiquement, les croisements sont envisagés avec deux parents et génèrent deux enfants.

Initialement, le croisement associé au codage par chaînes de bits est le croisement à découpage de chromosomes (slicing crossover). Pour effectuer ce type de croisement sur des chromosomes constitués de M gènes, on tire aléatoirement une position dans chacun des parents. On échange ensuite les deux sous-chaînes terminales de chacun des deux chromosomes, ce qui produit deux enfants C1 et C2 (voir figure III.3).

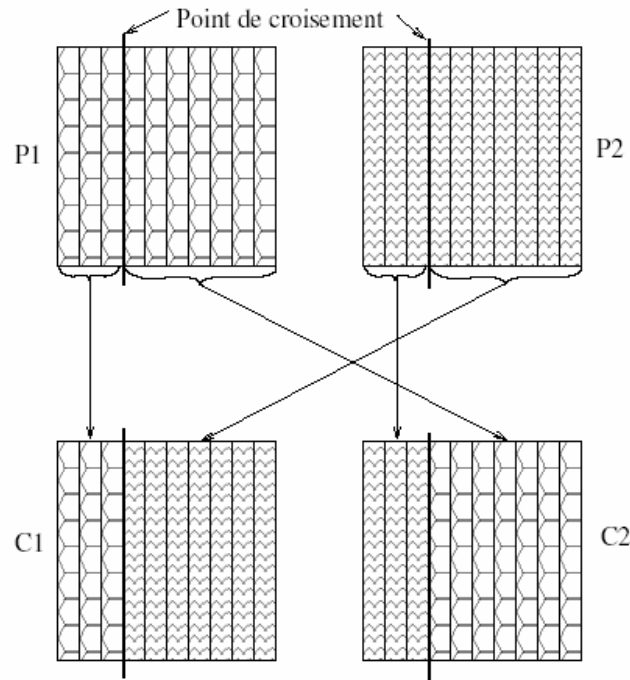


Figure III.3. Croisement à 1 point.

On peut étendre ce principe en découpant le chromosome non pas en 2 sous chaînes mais en 3, 4, etc [16], [17]. (figure III.4).

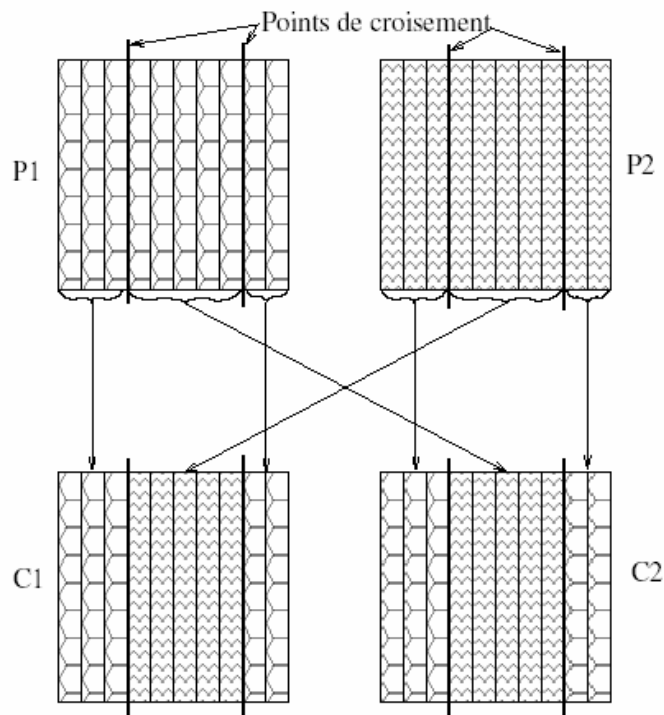


Figure III.4. Croisement à 2 points.

Ce type de croisement à découpage de chromosomes est très efficace pour les problèmes discrets. Pour les problèmes continus, un croisement “barycentrique” est souvent

utilisé : deux gènes $P1(k)$ et $P2(k)$ sont sélectionnés dans chacun des parents à la même position i . Ils définissent deux nouveaux gènes $C1(k)$ et $C2(k)$:

par combinaison linéaire :

$$\begin{cases} C1(k) = \alpha P1(k) + (1 - \alpha) P2(k) \\ C2(k) = (1 - \alpha) P1(k) + \alpha P2(k) \end{cases} \quad (\text{III.1})$$

où α est un coefficient de pondération aléatoire adapté au domaine d'extension des gènes (il n'est pas nécessairement compris entre 0 et 1, il peut par exemple prendre des valeurs dans l'intervalle $[-0.5, 1.5]$, ce qui permet d'engendrer des points entre, ou à l'extérieur des deux gènes considérés).

Dans le cas particulier d'un chromosome matriciel constitué par la concaténation de vecteurs, on peut étendre ce principe de croisement aux vecteurs constituant les gènes (figure III.5).

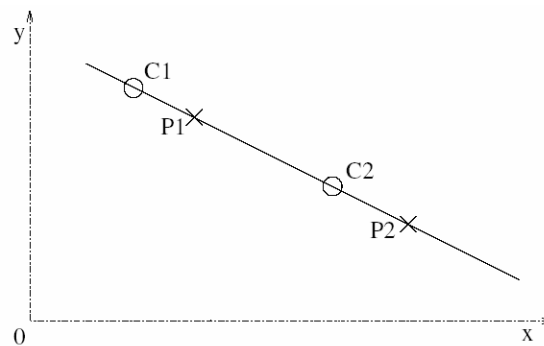


Figure III.5. Croisement barycentrique.

$$\begin{cases} \bar{C}1(k) = \alpha \bar{P}1(k) + (1 - \alpha) \bar{P}2(k) \\ \bar{C}2(k) = (1 - \alpha) \bar{P}1(k) + \alpha \bar{P}2(k) \end{cases} \quad (\text{III.2})$$

On peut imaginer et tester des opérateurs de croisement plus ou moins complexes sur un problème donné mais l'efficacité de ces derniers est souvent liée intrinsèquement au problème.

III.2.4.1. Croisement réel

III.2.4.1.1. Croisement manipulant des variables continue

Le procédé pour calculer les solutions enfants $x_i^{(1,k+1)}$ et $x_i^{(2,k+1)}$ d'après les solutions parents $x_i^{(1,k)}$ et $x_i^{(2,k)}$ est décrit comme suit [18], [20]: Un facteur de propagation β est défini comme la valeur absolue du rapport entre la différence des valeurs des enfants sur celles des parents :

$$\beta = \left| \frac{x_i^{2,k+1} - x_i^{1,k+1}}{x_i^{2,k} - x_i^{1,k}} \right| \quad (\text{III.3})$$

D'abord un nombre aléatoire u entre 0 et 1 est créé. Ensuite, à partir d'une fonction de distribution de probabilité spécifiée, l'ordonnée $\bar{\beta}$ est trouvée de sorte que la surface sous la courbe de probabilité entre 0 à $\bar{\beta}$ soit égale au nombre aléatoire choisi u . La distribution de probabilité utilisée pour créer la solution enfant est dérivée de l'analyse de recherche de puissance et est donnée comme suit [20] [18]:

$$C(\beta) = \begin{cases} 0.5(n+1)\beta^n, & \text{si } \beta \leq 1; \\ 0.5(n+1)\frac{1}{\beta^{n+2}}, & \text{autrement.} \end{cases} \quad (\text{III.4})$$

La figure III.6 montre la distribution (ci-dessus) de probabilité avec $n=2$ et 5 pour créer des enfants solutions d'après deux solutions de parent ($x_i^{(1,k)}=2.0$ et $x_i^{(2,k)}=5.0$) dans l'espace réelle. Dans les expressions ci-dessus, l'index n de distribution est tout nombre réel non négatif. Une grande valeur de n donne une probabilité plus élevée pour créer des valeurs près des valeurs de parent et une petite valeur de n permet aux points éloignés d'être choisis comme solutions enfants. En utilisant l'équation (III.4), on calcule $\bar{\beta}$ en égalisant l'aire sous la courbe de probabilité à u comme suit :

$$\bar{\beta} = \begin{cases} (2u)^{\frac{1}{n+1}}, & \text{si } u \leq 0.5; \\ \left(\frac{1}{2(1-n)}\right)^{\frac{1}{n+1}}, & \text{ailleurs.} \end{cases} \quad (\text{III.5})$$

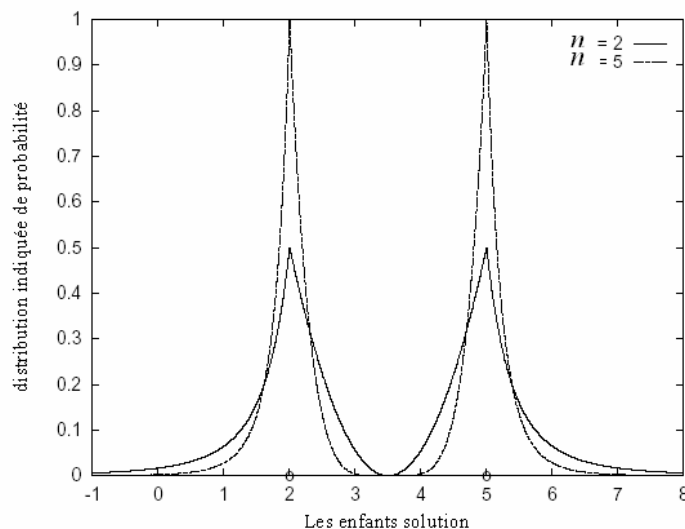


Figure III.6. Distribution de probabilité pour créer des solutions enfants des variables continues.

Après obtention de β d'après la distribution de probabilité (ci-dessus), les enfants solutions sont calculées comme suit :

$$x_i^{(1,k+1)} = 0.5[(1 + \bar{\beta})x_i^{(1,k)} + (1 - \bar{\beta})x_i^{(2,k)}] \quad (\text{III.6})$$

$$x_i^{(2,k+1)} = 0.5[(1 - \bar{\beta})x_i^{(1,k)} + (1 + \bar{\beta})x_i^{(2,k)}] \quad (\text{III.7})$$

Ainsi, le procédé étape par étape suivant est suivi pour créer deux solutions d'enfants $x_i^{(1,k+1)}$ et $x_i^{(2,k+1)}$ de deux solutions de parent $x_i^{(1,k)}$ et $x_i^{(2,k)}$:

étape 1 : choisir un nombre aléatoire $u \in [0,1]$.

étape 2 : calculer $\bar{\beta}$ en utilisant l'équation (III.5).

étape 3 : calculer les enfants solutions en utilisant les équations (III.6) et (III.7) .

III.2.5. Opérateur de mutation

L'opérateur de mutation apporte aux algorithmes génétiques la propriété d'ergodicité de parcours d'espace. Cette propriété indique que l'algorithme génétique sera susceptible d'atteindre tous les points de l'espace d'état, sans pour autant les parcourir tous dans le processus de résolution. Ainsi en toute rigueur, l'algorithme génétique peut converger sans croisement, et certaines implémentations fonctionnent de cette manière [19], [20]. Les propriétés de convergence des algorithmes génétiques sont donc fortement dépendantes de cet opérateur sur le plan théorique.

Pour les problèmes discrets, l'opérateur de mutation consiste généralement à tirer aléatoirement un gène dans le chromosome et à le remplacer par une valeur aléatoire (voir figure III.7). Si la notion de distance existe, cette valeur peut être choisie dans le voisinage de la valeur initiale.

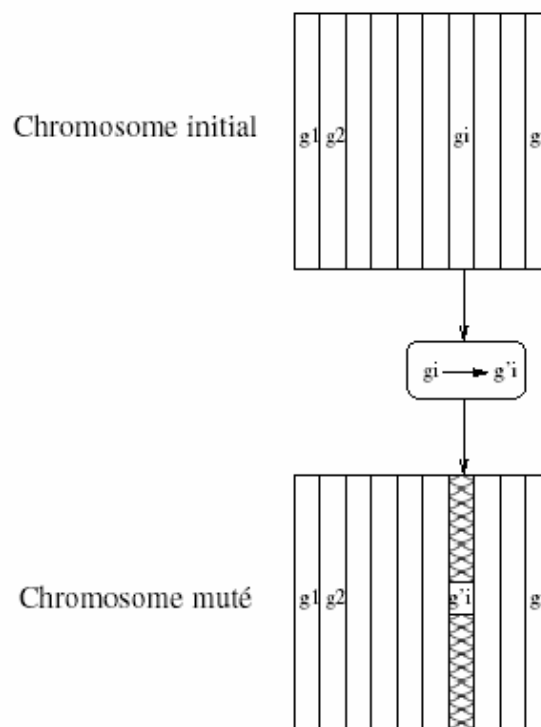


Figure III.7. Principe de l'opérateur de mutation.

Dans les problèmes continus, on procède un peu de la même manière en tirant aléatoirement un gène dans le chromosome, auquel on ajoute un bruit généralement gaussien. L'écart-type de ce bruit est difficile à choisir a priori.

III.2.5. 1. Mutation réelle

III.2.5. 1.1. La manipulation de mutation des variables continue

Après l'application des opérateurs de reproduction et de croisement, un opérateur de mutation utilisé avec une petite probabilité de mutation p_m . Pour des variables continues, les valeurs courantes des variables sont changées à une valeur voisine en utilisant le polynôme de distribution de probabilité. Celui-ci a sa moyenne comme la valeur courante et sa variance comme une fonction de l'index de distribution n . Pour exécuter la mutation un facteur δ de perturbation est défini comme suit [18], [20]:

$$\delta = \frac{c - p}{\Delta_{\max}} \quad (\text{III.8})$$

Où : Δ_{\max} est une quantité fixe représentant la perturbation maximale permis dans la valeur des parents p et c est la valeur mutée.

Similairement à l'opérateur de croisement, la valeur mutée est calculée avec une distribution de probabilité qui dépend du facteur de perturbation δ :

$$P(\delta) = 0.5(n+1)(1-|\delta|)^n \quad (\text{III.9})$$

La distribution de probabilité (ci-dessus) est valide dans la gamme $\delta \in (-1,1)$. Cette distribution est montrée sur la figure III.8 pour différentes valeurs de l'index de distribution n . Pour créer une valeur mutée un nombre aléatoire u est créé dans la gamme (0,1). Ensuite l'équation suivante peut être employée pour calculer le facteur de perturbation $\bar{\delta}$ correspondant à u en utilisant la distribution de probabilité (ci-dessus):

$$\bar{\delta} = \begin{cases} (2u)^{\frac{1}{n+1}} - 1, & \text{si } u < 0.5; \\ 1 - [2(1-u)]^{\frac{1}{n+1}}, & \text{si } u \geq 0.5. \end{cases} \quad (\text{III.10})$$

Ensuite la valeur mutée est calculée comme suit :

$$c = p + \bar{\delta}\Delta_{\max} \quad (\text{III.11})$$

Pour des variables discrètes codées directement, une valeur permise voisine est choisie avec une version discrète de la distribution de probabilité $P(\delta)$ (ci-dessus).

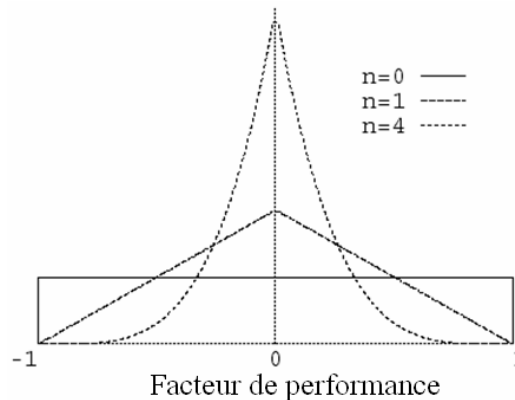


Figure III.8. La distribution de probabilité pour créer une valeur mutée pour des variables continues.

III.2.6. Principes de sélection

A l'inverse d'autres techniques d'optimisation, les algorithmes génétiques ne requièrent pas d'hypothèse particulière sur la régularité de la fonction objectif. L'algorithme génétique n'utilise notamment pas ses dérivées successives, ce qui rend très vaste son domaine d'application. Aucune hypothèse sur la continuité n'est non plus requise. Néanmoins, dans la pratique, les algorithmes génétiques sont sensibles à la régularité des fonctions qu'ils optimisent.

Le peu d'hypothèses requises permet de traiter des problèmes très complexes. La fonction à optimiser peut ainsi être le résultat d'une simulation. La sélection permet d'identifier statistiquement les meilleurs individus d'une population et d'éliminer les mauvais.

On trouve dans la littérature un nombre important de principes de sélection plus ou moins adaptés aux problèmes qu'ils traitent. Les trois principes de sélection suivants sont les plus couramment utilisés [21]:

- *Selection par Roulette (Roulette wheel selection);*
- *Stochastic remainder without replacement selection;*
- *Sélection par Tournoi*

Le principe de *selection par Roulette* consiste à associer à chaque individu un segment dont la longueur est proportionnelle à sa *fitness*. On reproduit ici le principe de tirage aléatoire utilisé dans les roulettes de casinos avec une structure linéaire. Ces segments sont ensuite concaténés sur un axe que l'on normalise entre 0 et 1. On tire alors un nombre aléatoire de distribution uniforme entre 0 et 1, puis on "regarde" quel est le segment sélectionné. Avec ce système, les grands segments, c'est-à-dire les bons individus, seront plus souvent choisis que les petits.

Lorsque la dimension de la population est réduite, il est difficile d'obtenir en pratique l'espérance mathématique de sélection en raison du peu de tirages effectués.

Un biais de sélection plus ou moins fort existe suivant la dimension de la population.

La *Stochastic remainder without replacement selection* évite ce genre de problème. Décrivons ce principe de sélection :

- Pour chaque élément i , on calcule le rapport r_i de sa *fitness* sur la moyenne des *fitness*.
- Soit $e(r_i)$ la partie entière de r_i , chaque élément est reproduit exactement $e(r_i)$ fois.
- La *roulette wheel selection* précédemment décrite est appliquée sur les individus affectés des *fitness* $r_i - e(r_i)$.

Lorsque des populations de faible taille sont utilisées, ce principe de sélection s'avère généralement efficace dans les applications pratiques.

Enfin le principe de sélection par Tournoi est comme suit :

On sélectionne $\frac{n}{2}$ individus aptes à la reproduction. Ces individus sont sélectionnés de la manière suivante :

K individus sont tirés au sort dans la population des n individus (K est un paramètre appelé taille du tournoi). Il existe différentes sélections par tournoi : déterministe ou probabiliste. Dans le cas du tournoi déterministe, le meilleur des K individus gagne le tournoi. Dans le cas probabiliste, chaque individu peut être choisi comme vainqueur avec une probabilité proportionnelle à sa fonction d'évaluation.

Il est tout à fait possible que certains individus participent à plusieurs tournois : s'ils gagnent plusieurs fois, ils auront donc droit d'être copiés plusieurs fois dans la génération intermédiaire, ce qui favorisera la pérennité de leurs gènes. La figure III.9 montre un exemple de sélection par tournoi entre deux individus (tournoi binaire).

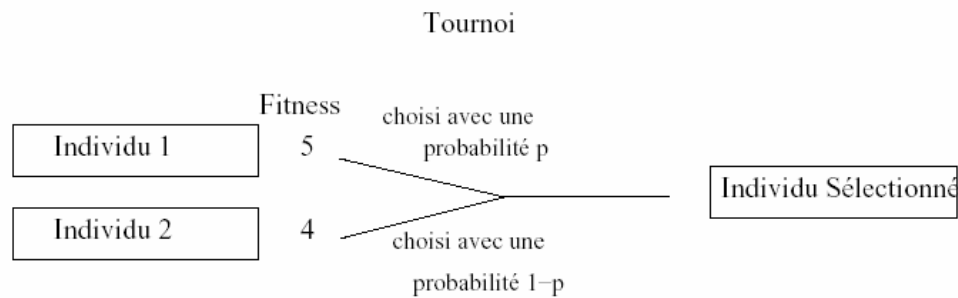


Figure III.9. Exemple de sélection par Tournoi.

III.2.7. La réinsertion

Dès qu'une nouvelle population est générée par sélection et recombinaison d'individus de la population précédente, la fitness des individus dans la nouvelle population peut être déterminée. Si moins d'individus sont produits par recombinaison par rapport à la taille de la population originale, alors la différence fractionnelle entre la nouvelle et l'ancienne taille de la population est appelée *génération gap*.

L'insertion des descendants dans la population peut être effectuée par plusieurs manières. Dans le cas le plus simple, par exemple, la population entière des parents est remplacée par leurs descendants inconditionnellement. En dehors de la supposition d'une population de taille constante, cette stratégie de remplacement montre des similarités dans beaucoup d'espèces naturelles où la reproduction prend place de façon saisonnière, et les parents meurent avant que leurs enfants ne soient nés. Ceci est connu comme le remplacement *générationnel*.

Dans la nature, il y a aussi un intérêt dans le cas des AGs où les descendants ont une chance de rivaliser avec au moins quelques-uns de leurs parents. Dans le cas de l'AG *steady-state* ou l'AG *incrémental*, un nombre minime d'enfants (typiquement un ou deux) est produit à travers recombinaison et mutation. Ces descendants sont alors évalués, et probablement réinsérés dans la population, en remplaçant:

- Des membres aléatoirement choisis de la population des parents.
- Les membres les plus anciens de la population des parents.
- Leurs propres parents.
- Les membres les moins adaptés de la population des parents.

La réinsertion actuelle peut se faire :

- Inconditionnellement.
- Seulement si les enfants sont plus adaptés que les individus qu'ils vont les remplacer.

- D'une façon probabiliste, suivant l'adaptation des descendants par rapport à celles des individus qu'ils vont les remplacer.

Si quelques individus ne montrent pas une habilité à se reproduire plus loin, la réinsertion aura finalement le même effet que la sélection. La pression sélective totale ou le biais sur le meilleur individu, imposée sur la population n'est pas déterminée seulement par la stratégie du calcul de fitness, mais elle est également influencée par l'instant et la façon d'application de la procédure de réinsertion. En particulier, remplacer toujours les individus les moins adaptés dans la population augmente fortement l'effectif, la différence des fitness entre les individus les plus forts et les plus faibles dans la population. Cela est dû, en plus

d'être moins appropriés pour la sélection, au fait que les individus les plus faibles tendent à mourir plus rapidement, donc auront moins de chance pour participer aux essais de sélection que les individus les plus forts. Les stratégies de réinsertion qui garantissent la conservation du meilleur individu sont dites *élitistes*.

L'utilisation de différentes combinaisons de sélection et de stratégies de réinsertion est largement rapportée dans la littérature des AGs. Par exemple, les AGs steady-state sont supposés fournir de meilleurs résultats que ceux générationnels, prétendument parce qu'ils exploitent les bons individus, récemment produits, d'une façon plus rapide par rapport aux autres cas. Comme il a été signalé par Harvey [1], cela se traduit par une augmentation de la pression sélective, même si des remplacements aléatoires sont utilisés. Le remplacement basé sur la fitness fournirait même de plus grande pression sélective.

La pression sélective supplémentaire fournie par la reproduction steady-state en prenant en compte sa contrepartie générationnelle peut être la simple raison pour laquelle l'AG steady-state paraît donner de meilleures performances dans certains cas. Une pression sélective suffisante est spécialement importante si de grandes erreurs de sélection peuvent se produire. Le travail de Syswerda (1991) ne révèle aucune différence fondamentale entre les AGs générationnels et les AGs steady-state avec remplacement aléatoire inconditionnel. L'augmentation de la pression sélective dans un AG générationnel, d'une façon explicite à travers le calcul de fitness peut produire des résultats semblables implicites à travers la stratégie de remplacement.

III.2.8. Les paramètres d'un AG

La convergence d'un AG dépend fortement de certains paramètres qu'on doit fixer à l'avance. Dans ce qui suit nous présentons chacun de ces paramètres et leurs rôles dans le fonctionnement d'un algorithme génétique [22].

III.2.8.1. La taille de la population

Les conditions de convergence changent avec la taille de la population ; lorsque celle-ci est grande, sa diversité augmente ce qui diminue la convergence vers un optimum local. Mais le temps d'exécution de chaque génération augmente, et la recherche risque de s'effectuer d'une façon redondante et l'efficacité de l'algorithme est globalement affectée. Cette taille selon le cas se situe entre 25 et 100 individus. Par contre, si la taille de la population est petite, alors la probabilité de s'attarder sur des minima locaux est grande.

III.2.8.2. Le taux de croisement

Le *taux de croisement* détermine la proportion des individus qui sont croisés parmi ceux qui remplaceront l'ancienne génération. L'opérateur de croisement est appliqué avec une probabilité P_c , et plus cette valeur est grande plus de nouvelles structures (individus) sont introduites dans la nouvelle génération. En effet, si ce taux a été fixé à 1, tout descendant est obtenu par croisement. Mais quand ce taux est élevé, les structures performantes sont trop fréquemment détruites. Par contre, si ce taux est trop bas, la population n'évolue pas assez vite. En général P_c varie entre 0.7 et 1.0.

III.2.8.3. Le taux de mutation

L'opérateur de mutation est appliqué avec une probabilité P_m ; si ce taux est grand alors la recherche devient purement aléatoire et la population est diversifiée, et l'algorithme génétique perd de son efficacité. Si au contraire ce taux est faible, la population est moins diversifiée et en plus il y a risque de stagnation. Des études empiriques conseillent pour l'obtention de bons résultats une fréquence qui se situe autour d'une mutation tous les 1000 bits.

A noter que ces paramètres dépendent étroitement du type de problème à résoudre.

III.3. Améliorations classiques

III.3.1. Introduction

Les processus de sélection présentés sont très sensibles aux écarts de *fitness* et, dans certains cas, un très bon individu risque d'être reproduit trop souvent et peut même provoquer l'élimination complète de ses congénères ; on obtient alors une population homogène contenant un seul type d'individu [16]. Ainsi, dans l'exemple de la figure III.10 le second mode M2 risque d'être le seul représentant pour la génération suivante et seule la mutation pourra aider à atteindre l'objectif global M1, au prix de nombreux essais successifs.

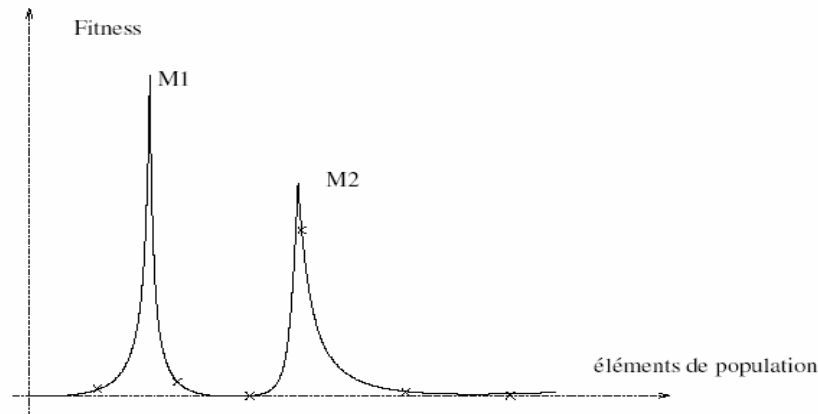


Figure III.10. Exemple où les sélections classiques risquent de ne reproduire qu'un individu.

Pour éviter ce comportement, il existe d'autres modes de sélection (*ranking*) ainsi que des principes (*scaling*, *sharing*) qui empêchent les individus "forts" d'éliminer complètement les plus "faibles". On peut également modifier le processus de sélection en introduisant des tournois entre parents et enfants, basés sur une technique proche du recuit simulé.

Enfin, on peut également introduire des recherches multiobjectif, en utilisant la notion de dominance lors de la sélection.

III.3.2. Scaling

Le *scaling* ou mise à l'échelle, modifie les *fitness* afin de réduire ou d'amplifier artificiellement les écarts entre les individus. Le processus de sélection n'opère plus sur la *fitness* réelle mais sur son image après *scaling*. Parmi les fonctions de *scaling*, on peut envisager le *scaling* linéaire et le *scaling* exponentiel. Soit f_r la *fitness* avant *scaling* et f_s la *fitness* modifiée par le *scaling*.

III.3.2.1. Scaling linéaire

Dans ce cas la fonction de *scaling* est définie de la façon suivante [31] :

$$f_s = af_r + b \quad (\text{III.12})$$

$$a = \frac{\max' - \min'}{\max - \min}, \quad b = \frac{\min' \cdot \max - \min \cdot \max'}{\max - \min} \quad (\text{II.13})$$

En règle générale, le coefficient a est inférieur à un, ce qui permet de réduire les écarts de *fitness* et donc de favoriser l'exploration de l'espace. Ce *scaling* est statique par rapport au numéro de génération et pénalise la fin de convergence, lorsque l'on désire favoriser les modes dominants.

La figure III.11 donne un exemple de l'influence de la fonction de *scaling* sur la pression de sélection.

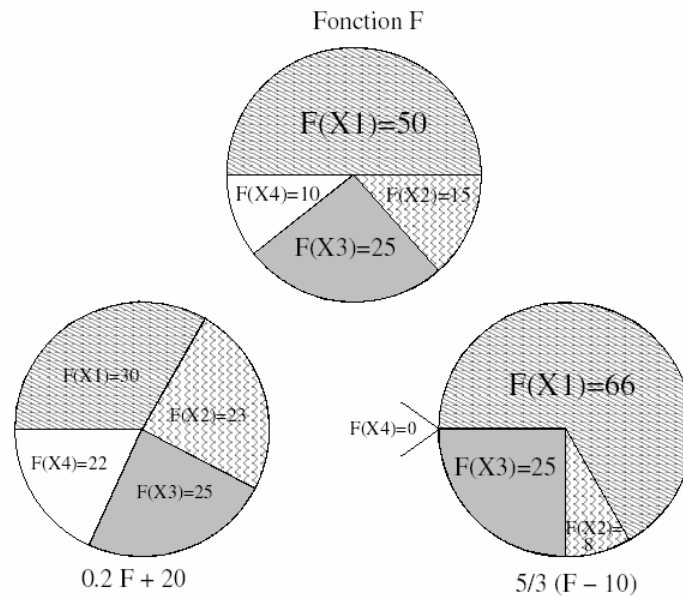


Figure III.11. Influence de la fonction de *scaling* sur la pression de sélection.

III.3.2.2. Scaling exponentiel

Il est défini de la façon suivante [31] (figure III.12) :

$$f_s = (f_r)^{k(n)} \quad (\text{III.14})$$

Où n est la génération courante.

– Pour k proche de zéro, on réduit fortement les écarts de *fitness* ; aucun individu n'est vraiment favorisé et l'algorithme génétique se comporte comme un algorithme de recherche aléatoire et permet d'explorer l'espace.

– Pour k proche de 1 : le *scaling* est inopérant.

– Pour $k > 1$ les écarts sont exagérés et seuls les bons individus sont sélectionnés, ce qui produit l'émergence des modes.

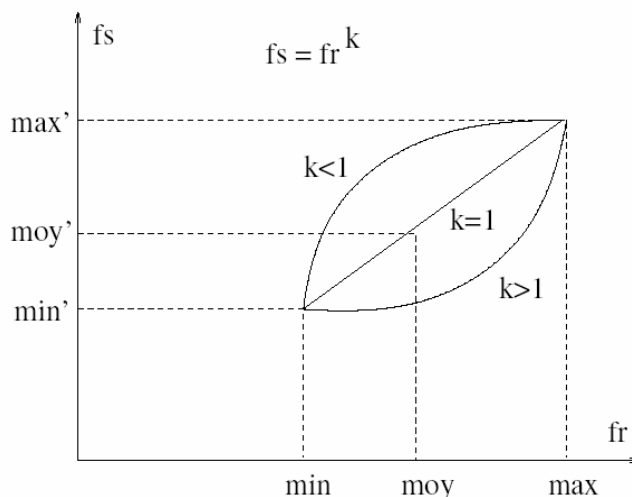


Figure III.12. Fonction de scaling exponentielle.

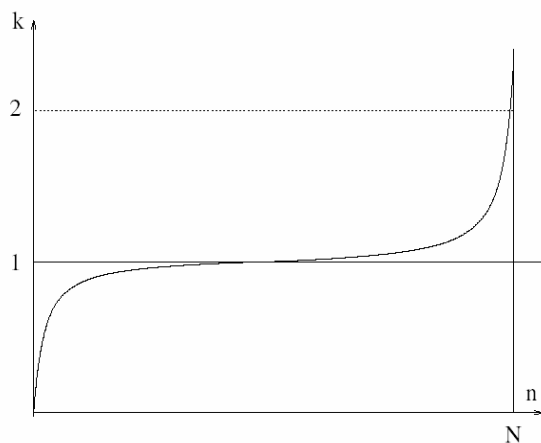


Figure III.13 Allure de l'évolution de k en fonction des générations.

Dans la pratique, on fait généralement varier k des faibles valeurs vers les fortes valeurs au cours des générations. Pour cela on peut utiliser la formule suivante :

$$k = \left(\tan \left[\frac{n}{N+1} \frac{\pi}{2} \right] \right)^p \quad (\text{III.15})$$

n : étant la génération courante, N le nombre total de générations prévues, p un paramètre à choisir. L'évolution de k en fonction de la génération n est donnée par la figure III.13.

Dans le cas des fonctions objectifs multi-modes présentant des optima quasi équivalents, cette technique de *scaling*, en amplifiant les écarts de *fitness* en fin de convergence, va effectivement favoriser le mode dominant, mais aussi masquer les modes sous-optimaux, qui peuvent tout de même présenter un intérêt. Le *scaling* permet donc une bonne exploration de l'espace d'état, mais ne favorise pas la répartition des individus sur les différents modes de la fonction objectif.

III.3.3. Sharing

III.3.3.1. Introduction

L'objectif du *sharing* est de répartir les individus de la population sur tous les sommets de la fonction à optimiser. La figure III.14 présente deux exemples de répartitions de populations dans le cas d'une fonction à cinq sommets : le premier sans *sharing*, le second avec *sharing*.

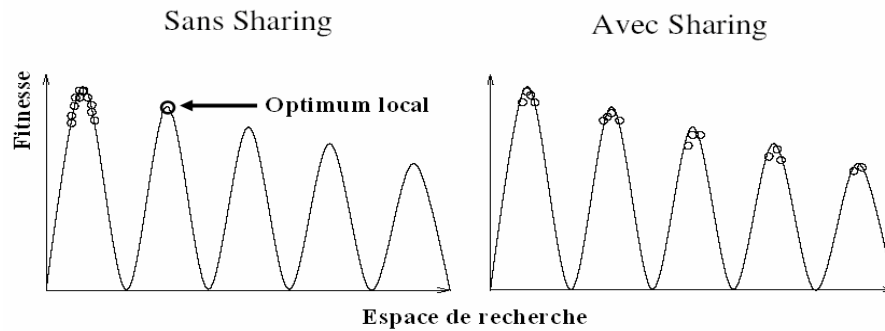


Figure III.14. Objectif du *sharing*.

III.3.3.2. Principe

De la même façon que le *scaling*, le *sharing* consiste à modifier la *fitness* utilisée par le processus de sélection. Pour éviter le rassemblement des individus autour d'un sommet dominant, le *sharing* pénalise les *fitness* en fonction du taux d'agrégation de la population dans le voisinage d'un individu. Il requiert l'introduction d'une notion de distance. Dans la pratique, il faut définir une distance indiquant la dissimilarité entre deux individus. Cette distance est alors utilisée pour calculer la nouvelle *fitness* de la façon suivante :

$$f'_i = \frac{f_i}{m_i} ; m_i = \sum_{j=i}^N S(d(x_i, x_j)) \quad (\text{III.16})$$

Avec

$$S(d) = 1 - \left(\frac{d}{\sigma_{share}} \right)^\alpha \text{ si } d < \sigma_{share} \quad (\text{III.17})$$

$$S(d) = 0 \text{ si } d > \sigma_{share} \quad (\text{III.18})$$

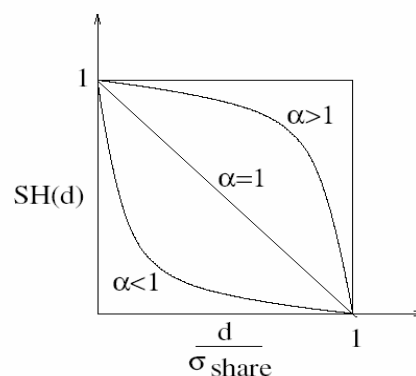


Figure III.15. Allure de $S\left(\frac{d}{\sigma_{share}}\right)$

Le paramètre σ_{share} permet de délimiter le voisinage d'un point et dépend du problème traité. La figure 2.15 donne l'allure de $S(d)$ pour différentes valeurs de α . Suivant la valeur donnée à α le *sharing* sera plus ou moins efficace. Ainsi pour $\alpha < 1$, on pénalise les groupes très agglomérés.

Dans la pratique ce type de *sharing* donne effectivement de bons résultats mais au prix de N^2 calculs de distances entre chromosomes à chaque génération pour une population de taille N . Or les algorithmes génétiques induisent une complexité en N sans *sharing* et le fait de passer en N^2 peut être pénalisant, notamment pour N grand.

Pour réduire ce nombre, on utilise un *sharing* "clusterisé".

III.3.3.3. Sharing clusterisé

Pour effectuer ce type de *sharing* [32], on commence par identifier les différents groupes d'individus dans la population. Ce dernier utilise deux paramètres d_{min} et d_{max} respectivement pour fusionner des *clusters* ou en créer de nouveaux.

Initialement, chaque individu de la population est considéré comme le centre d'un *cluster*. On applique alors successivement les deux principes suivants :

- si deux centres sont à une distance inférieure à d_{min} , on fusionne ces derniers dans un *cluster* unique dont le centre est le barycentre des deux centres initiaux.
- un nouvel individu est agrégé à un *cluster* si sa distance au centre le plus proche est inférieure à d_{max} . Dans ce cas, on recalcule le centre du *cluster* global. Sinon, on crée un nouveau *cluster* contenant ce seul individu.

Ce principe de fusion-agrégation permet d'obtenir un nombre de *clusters* fluctuant avec la répartition des individus dans l'espace d'état. On applique ensuite le principe de *sharing* en modifiant les *fitness* de la façon suivante :

$$f'_i = \frac{f_i}{m_i} ; m'_i = n_c \left(1 - \left(\frac{d_{ic}}{2d_{max}} \right)^\alpha \right) \quad (\text{III.19})$$

Avec

- n_c : nombre d'individus contenus dans le *cluster* auquel appartient l'individu i .
- α : coefficient de sensibilité.
- d_{ic} : distance entre l'individu i et le centre du *cluster* c .

On montre que ce type de *sharing* induit une complexité en $O(N \log N)$ [32] pour des résultats tout à fait comparables à ceux fournis par le *sharing* classique.

Dans la pratique, on remarque que le réglage des coefficients d_{min} et d_{max} est assez délicat car l'efficacité de ces derniers dépend essentiellement de la connaissance a priori des distances inter-modes dans l'espace d'état, distance qu'il est très difficile d'estimer.

III.4. Avantages et inconvénients des algorithmes génétiques

III.4.1. Avantages

- Facile à implémenter,
- Possibilité d'incorporer d'autres méthodes (méthodes hybrides),
- Fournissent plusieurs alternatives (solutions),
- Bon pour des problèmes avec beaucoup d'optimum locaux (évite les optima locaux),

- L'espace de recherche peut être complètement exploré,
- Conceptuellement simple et flexible.

III.4.2. Inconvénients

- Parfois difficile de trouver :
 - Un bon codage,
 - Une bonne fonction d'adaptation,
 - De bons opérateurs de croisement et de mutation,
- Méthode lente, mais au moins, donne une solution en tout temps,
- Ne garantissent pas de trouver la solution optimale dans un temps fini (Aucune garantie sur la valeur des résultats),
- Moins efficace que d'autres méthodes (hill climbing, newton raphson, recuit simulé, la recherche tabou, branch and bound,...),
- Les AGs ne peuvent pas être utilisés directement comme régulateur en temps réel parce qu'il sont trop lents, c à d qu'ils nécessitent beaucoup de temps pour fonctionner.

III.5. Conclusion

Ce chapitre constitue une introduction aux algorithmes génétiques et fournit les éléments nécessaires à leur programmation. Les Algorithmes Evolutionnaires sont basées sur la théorie de l'évolution de Darwin. Par analogie avec le monde biologique, l'algorithme fait évoluer une population de dispositifs à l'aide de divers opérateurs : sélection, croisements, mutations.

Les algorithmes génétiques possèdent des caractéristiques d'évolution qui leurs permettent d'être très utilisés à une grande classe de problèmes d'optimisation. Ces mêmes caractéristiques font distinguer les AGs des autres mécanismes de recherche conventionnels, en ce qui concerne la capacité de trouver les solutions optimales d'un problème donné, tout en ne requérant que peu d'informations sur ces dernières et en manifestant de très grande habilité d'éviter le piège des optima locaux, sans oublier la possibilité d'une implémentation parallèle. Cependant, il devrait être signalé qu'il n'y a aucune base théorique solide des AGs, et qu'il n'y a aucune garantie qu'une solution soit trouvée.

En effet, la plupart des applications pratiques dans le domaine de la commande et l'identification des systèmes font appel à des problèmes d'optimisation de plusieurs critères (performances, robustesse, énergie, etc.).

Parmi les divers champs d'applications des AGs, on cite l'optimisation de la commande par logique floue. Le prochain chapitre met en évidence cette méthode d'optimisation.

CHAPITRE IV

CONTROLEUR MPPT FLOU OPTIMISE PAR LES ALGORITHMES GENETIQUES

V.1. Introduction

Quand on conçoit un contrôleur par logique floue (CLF), une connaissance experte du processus à commander peut être employée pour établir les fonctions d'appartenance et les règles d'inférence. Cependant il n'y a aucune procédure générale pour concevoir un CLF, vu que le temps est exigü et que beaucoup d'erreurs d'essais, peuvent être rencontrées lors de la réalisation d'un CLF. Dans cette approche traditionnelle, concevoir un CLF peut être un processus laborieux et long. Ces CLF tendent également à être non exportables et non adaptables à d'autres applications.

Il serait judicieux de chercher à optimiser les différents paramètres concernant un CLF dont le choix reste plutôt subjectif. On se propose de faire appel aux algorithmes génétiques. Ce sont des algorithmes qui peuvent être employés pour chercher dans un grand espace de possibilité des solutions optimales.

Les algorithmes génétiques sont des méthodes fiables et robustes pour chercher des solutions dans des espaces immenses. Ils sont inspirés de la théorie biologique d'évolution de la sélection naturelle.

V.2. Contrôleur MPPT Flou optimisé par les Algorithmes Génétiques

Dans la théorie de la logique floue, les concepts abstraits peuvent être représentés par des variables linguistiques. L'utilisation de ce terme « flou » mène à des contrôleurs à logique floue CLF, avec une flexibilité généralement inaccessible avec des processus de contrôle conventionnels.

Les contrôleurs CLF ont été testés avec succès dans un bon nombre de situations dans le contrôle industriel. Ces systèmes flous experts incluent des règles pour diriger la décision du processus et des fonctions d'appartenances afin de convertir les variables linguistiques en valeurs numériques nécessaires aux calculateurs pour automatiser le processus. La base des règles est souvent réalisée à partir de la connaissance de l'être humain, qui est généralement basée sur son expérience.

La méthode standard pour la détermination des fonctions d'appartenances produisant une performance optimale doit accélérer le développement du CLF. Cependant, localiser les fonctions d'appartenances optimales s'avère être difficile car la performance du contrôleur reste sensible aux petits changements dans la base des règles ou dans les fonctions d'appartenances. Ce contrôle devient plus difficile lorsqu'il doit être accompli en temps réel.

L'aptitude adaptative ainsi que la nature robuste des algorithmes génétiques font d'eux un outil potentiel pour l'établissement des fonctions d'appartenances des CLF. Les algorithmes génétiques sont également efficaces dans la génération des règles floues. De plus,

ils possèdent des qualités bénéfiques pour la conception d'un CLF adaptatif qui altère ses règles et/ou ses fonctions d'appartenances en temps réel tout en tenant compte des changements physiques de l'environnement aussi petits soient-ils.

Dans le domaine de l'optimisation des CLF en utilisant les algorithmes génétiques, un très grand nombre de recherches et travaux existent. Du coup, beaucoup d'approches et de méthodes ont été adoptées. Par exemple CHEN [15] applique les AG pour optimiser les règles d'inférence d'un CLF pour lequel les fonctions d'appartenance ont été déjà créées, tandis que Lee et Tackagi [26] ont utilisé un AG qui détermine le nombre de fonctions d'appartenance et le nombre de règles floues. Dans les deux cas, ils utilisent le codage à valeurs réelles, tandis que Belarbi [27] emploie le codage binaire pour son approche, où ils met en application le CLF comme réseau neurologique et emploie l'AG pour former les poids.

V.3. Conception du contrôleur MPPT flou original

Dans un but de poursuivre le PPM, comme pour toutes les méthodes de poursuite MPPT (vues dans le premier chapitre), le CLF marche aussi suivant la même philosophie.

Un CLF se compose de trois blocs qui sont :

La Fuzzification, l'Inférence et la Défuzzification.

V.3.1. Fuzzification

On suppose que le contrôle se fait par la satisfaction de deux critères relatifs à deux variables d'entrées du contrôleur flou proposé, qui sont:

L'erreur E et le changement de l'erreur CE à des instants échantillonnés k .

Les variables E et CE sont exprimées comme suit:

$$E(k) = \frac{P_{ph}(k) - P_{ph}(k-1)}{V_{ph}(k) - V_{ph}(k-1)} \quad (V.1)$$

$$CE(k) = E(k) - E(k-1) \quad (V.2)$$

Où $P_{ph}(k)$ et $V_{ph}(k)$ sont respectivement: la puissance et la tension du générateur photovoltaïque.

L'entrée $E(k)$ montre si le point de fonctionnement de la charge est situé à gauche ou à droite du point de puissance maximale de la courbe P-V, tandis que l'entrée $CE(k)$ montre la direction du point de fonctionnement.

V.3.1.1. Variables linguistiques

En utilisant les bases des sous ensembles flou, on peut exprimer les variables d'entrées et de sortie en terme des variables linguistiques suivantes :

PG: (positif grand)

PP: (positif petit)

ZE: (Zéro)

NP: (négatif petit)

NG: (négatif grand).

V.3.1.2. Fonctions d'appartenances

La génération de fonctions d'appartenance s'effectue soit d'une manière itérative, soit avec la méthode d'essai et d'erreur ou encore en se basant sur l'expérience humaine. Il

n'existe jusqu'à présent aucune méthode généralisée pour la formulation des stratégies de la logique floue.

On définit pour la résolution des problèmes de poursuite du point de puissance maximale les fonctions d'appartenance suivantes :

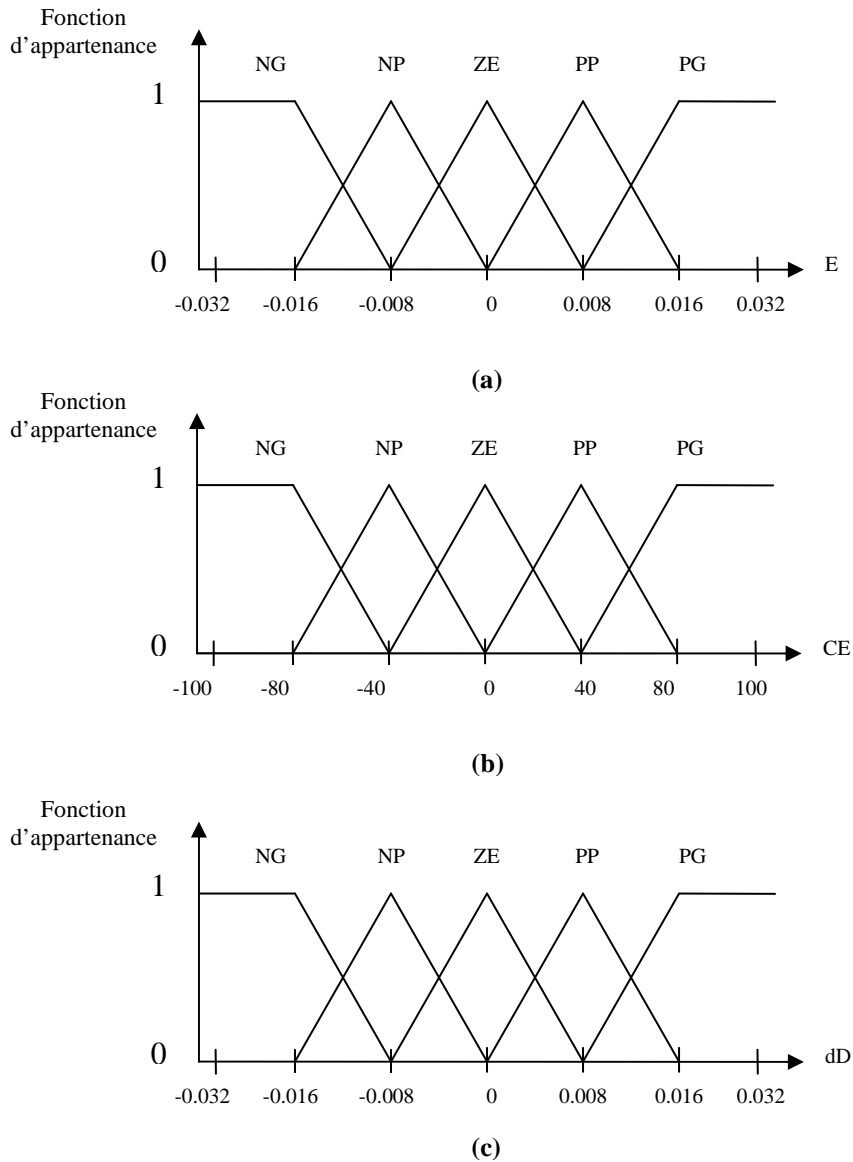


Figure V.1. Fonctions d'appartenance de :
 (a) L'entrée E (b) L'entrée CE (c) La sortie dD.

V.3.2. Méthode d'inférence

La table suivante est la table de règles du CLF où :

- Les entrées de la matrice sont: l'erreur E et le changement de l'erreur CE .
- La sortie est le changement du rapport cyclique dD .

Les règles de contrôle doivent être désignées de sorte que la variable d'entrée E doit être toujours *Zéro*.

E \ CE	NG	NP	ZE	PP	PG
NG	ZE	ZE	PG	PG	PG
NP	ZE	ZE	PP	PP	PP
ZE	PP	ZE	ZE	ZE	NP
PP	NP	NP	NP	ZE	ZE
PG	NG	NG	NG	ZE	ZE

Table V.1 Table de règles floues.

C'est la méthode de Mamdani qui a été utilisée comme méthode d'inférence floue. Elle consiste à utiliser l'opérateur MIN pour le ET et l'opérateur MAX pour le OU.

V.3.3. Déffuzzification

La méthode de défuzzification, utilisée dans le CLF est la méthode du centre de gravité.

V.4. Optimisation par les Algorithmes Génétiques

Dans ce qui suit, on applique les AGs pour concevoir le meilleur CLF pour la commande MPPT. Après la conception (MPPT flou) du CLF, et comme on sais qu'il y a toujours des défauts malgré notre expérience au niveau des règles d'inférences et des fonctions d'appartenances, on va essayer dans ce travail de trouver les fonctions d'appartenances optimales en utilisant les AGs, et cela se fait par les étapes suivantes :

V.4.1. Choix du critère d'optimisation

Il existe plusieurs critères d'optimisation [28]. Dans ce travail nous allons utiliser le critère quadratique à minimisé suivant :

$$J = \int e^2 dt \quad (V.3)$$

Avec :

$$e = P_{\max} - P \quad (V.4)$$

Où : P est la puissance désirée et P_{\max} est la puissance maximale délivrée par le module sous les conditions normalisées : $T=25^{\circ}C$ et $S=1000W/m^2$.

Ce choix a été fait dans le but d'améliorer le temps de réponse et de réduire les fluctuations.

V.4.2. Création de la population initiale

Dans la conception du contrôleur MPPT flou on a choisi trois fonctions d'appartenances, deux entrées et une sortie, E, CE et dD: respectivement. Chacune d'entre elles est composée de 5 sous ensembles flous.

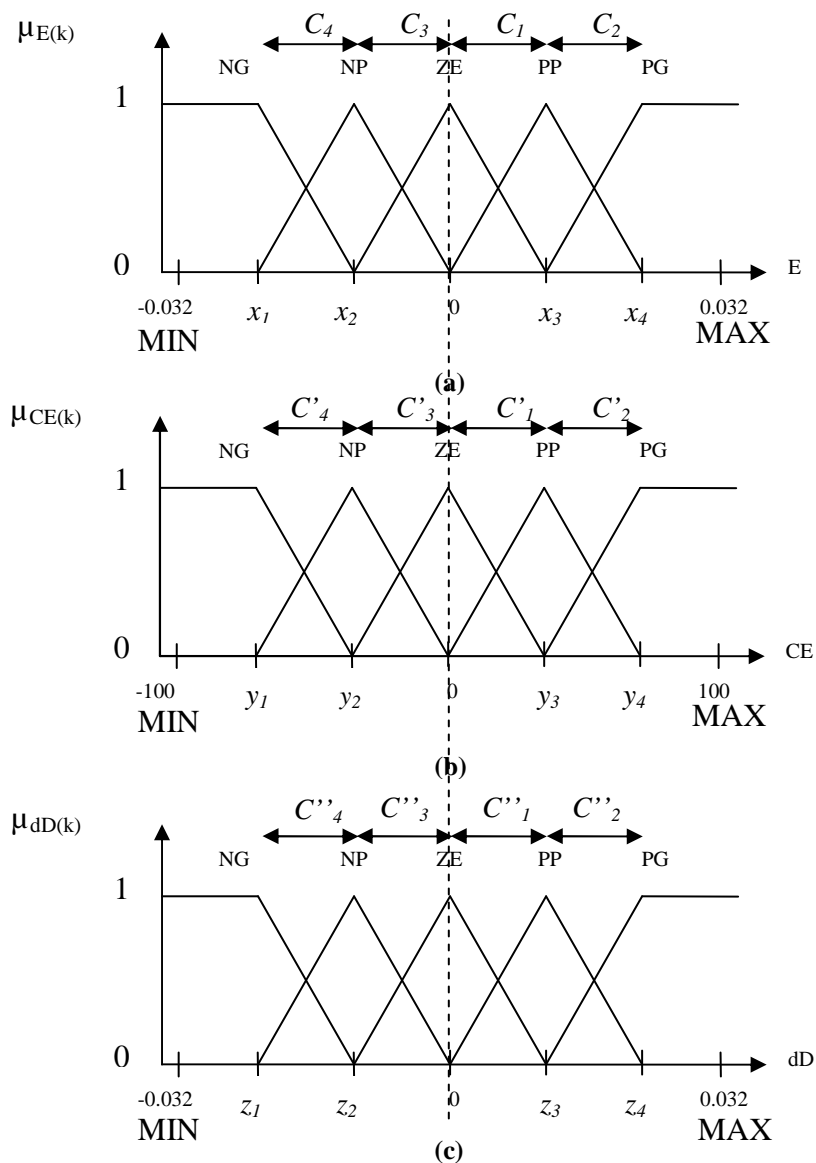


Figure V.2. Codages des fonctions d'appartenance de :

(a) L'entrée E (b) L'entrée CE (c) La sortie dD.

La population se compose d'un ensemble d'individus, où chaque individu est composé des trois chromosomes : E, CE et dD, chaque chromosome est composé d'un ensemble de gènes (figure V.2) :

- ✓ Pour le chromosome E les gènes sont : C_1, C_2, C_3, C_4 . On définit un intervalle de variation de C (qui est l'espace de recherche) ; qui varie entre $[0.01 \ 0.99]$
- ✓ Pour le chromosome CE les gènes sont : C'_1, C'_2, C'_3, C'_4 . On définit un intervalle de variation de C' ; qui varie entre $[0.01 \ 0.99]$
- ✓ Pour le chromosome dD les gènes sont : $C''_1, C''_2, C''_3, C''_4$. On définit un intervalle de variation de C'' ; qui varie entre $[0.01 \ 0.99]$

La forme d'individu qui en résulte est illustrée sur la figure V.3 :

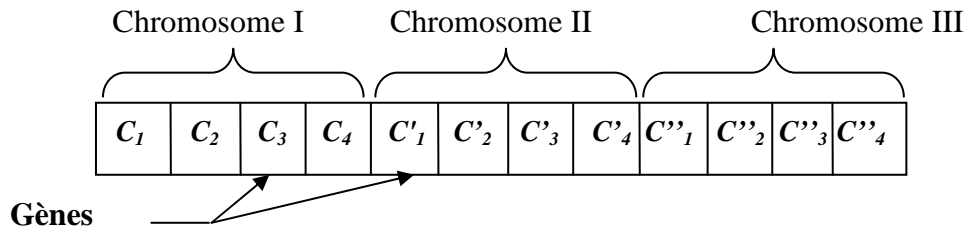


Figure V.3. Forme d'un individu.

Les relations entre les C_i , C'_i , C''_i et x'_i , y'_i , z'_i sont données par :

$$\begin{cases} x_1 = (C_4 + C_3)(-0.032) \\ x_2 = (C_3)(-0.032) \\ x_3 = (C_1)(0.032) \\ x_4 = (C_1 + C_2)(0.032) \end{cases}, \quad \begin{cases} y_1 = (C'_4 + C'_3)(-100) \\ y_2 = (C'_3)(-100) \\ y_3 = (C'_1)(100) \\ y_4 = (C'_1 + C'_2)(100) \end{cases} \quad \text{et} \quad \begin{cases} z_1 = (C''_4 + C''_3)(-0.032) \\ z_2 = (C''_3)(-0.032) \\ z_3 = (C''_1)(0.032) \\ z_4 = (C''_1 + C''_2)(0.032) \end{cases} \quad (\text{V.5})$$

Afin de trouver l'individu solution (individu optimal) on se limite à chercher les valeurs des gènes qui sont inconnus. Pour atteindre l'optimum global, on prend une grande taille de la population qui est égale à 100 individus.

V.4.3. Sélection

Dans cette étape on choisit les meilleurs individus et cela se fait par la fonction d'adaptation (fitness) suivante :

$$J = \int_0^{t_f} (P_{\max} - P)^2 dt \quad (\text{V.6})$$

Et avec la méthode de sélection par tournoi binaire, cette dernière permet d'éliminer la pression de sélection.

V.4.4. Croisement

Pour le croisement on a utilisé la méthode de croisement manipulant des variables continues (croisement réel) (déjà discuté dans la section (10.3)), avec une probabilité de croisement $P_c=1$ et un indice de distribution $n=5$.

V.4.5. Mutation

Pour la mutation on a utilisé la méthode de la manipulation de mutation des variables continues (mutation réelle) (déjà discutée dans la section (10.4)), avec une probabilité de mutation $P_m=1/NG=1/12$, et un indice de distribution $n=5$.

V.4.6. Réinsertion

Dans cette étape la réinsertion a été faite par la méthode élitiste. Dans ce cas, la population des parents et celles des descendants sont rassemblées, et les N meilleurs individus de cet ensemble sont choisis pour former la nouvelle population des parents.

V.4.7. Critère d'arrêt

Le critère d'arrêt qu'on a utilisé lors de l'exécution du programme, est quand le nombre de génération maximal atteint la valeur 50.

V.4.8. Fonctions d'appartenance du CLF optimale

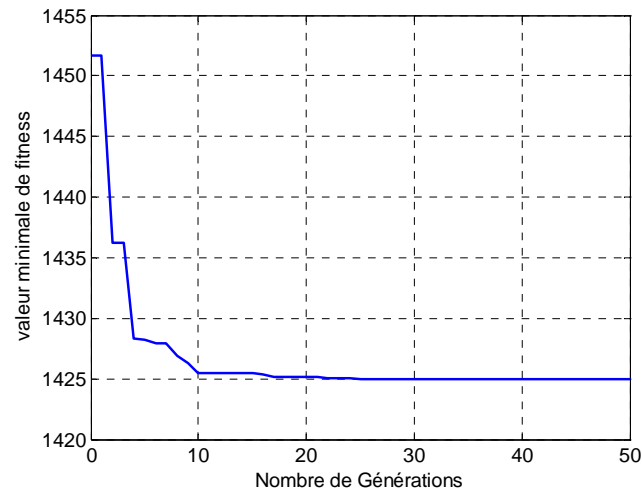


Figure V.4. Valeurs minimales de la fonction objective en fonction du nombre de générations.

Les variations de la valeur minimale de la fonction objective (fitness) en fonction du nombre de générations sont données dans la Figure V.4. Nous remarquons que l'AG a convergé progressivement vers une solution optimale représentée par le meilleur individu de la population de la dernière génération. Cet individu donne les valeurs des paramètres recherchés.

La solution optimale obtenue donne alors les formes des fonctions d'appartenance illustrées dans la figure V.5 suivante :

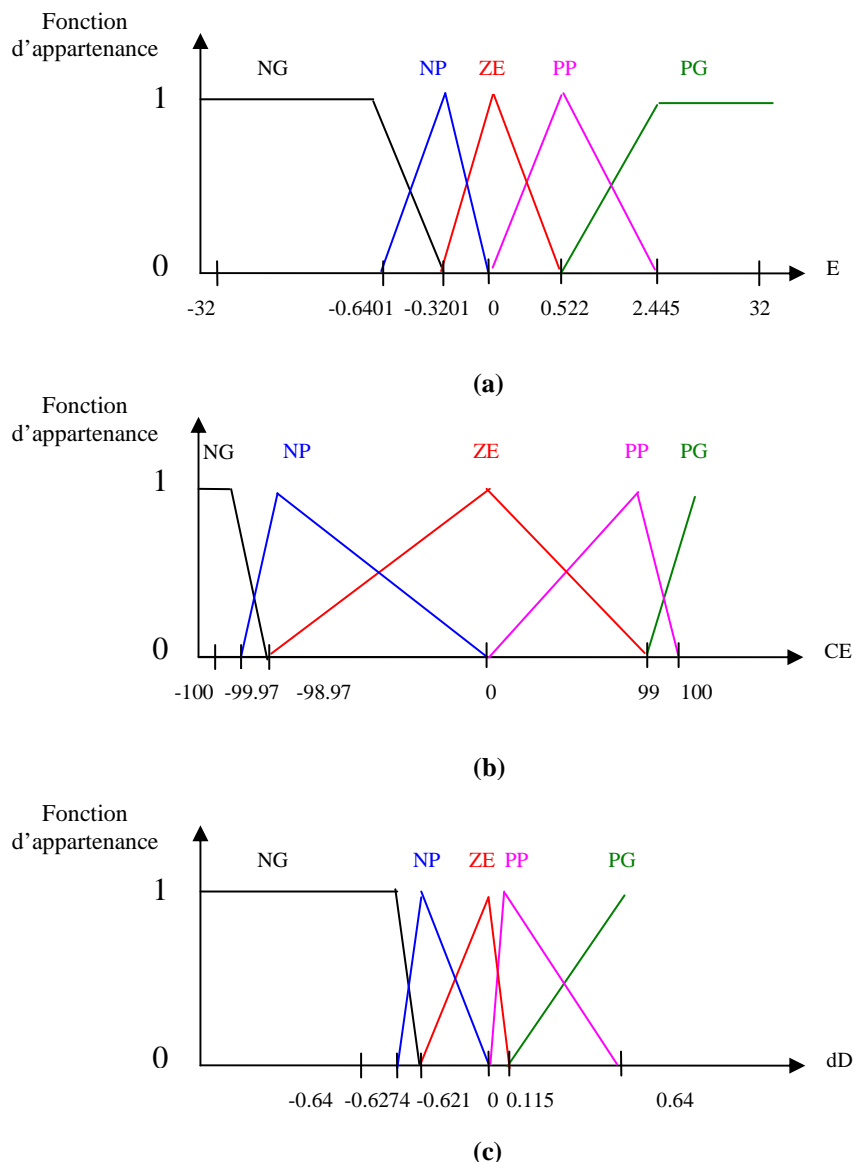


Figure V.5 Fonctions d'appartenance du CLF flou optimal.

V.5. Simulation et résultats

Pour valider le meilleur contrôleur MPPT Flou optimal obtenu (présenté sur la figure V.5), on va effectuer une série d'essais par simulation. Les simulations ont été faites en utilisant un hacheur Boost ensuite un hacheur buck, et en soumettant les deux contrôleurs MPPT : MPPT flou et MPPT flou optimal à différentes variations et combinaisons des conditions environnementales. Les résultats obtenus sont donnés dans les sections qui suivent.

V.5.1. Simulation du système de poursuite MPPT

Le schéma synoptique du contrôleur MPPT simulé sous Simulink est donnée par le bloc global représenté par la figure V.6. Nous soulignons ici que chaque bloc a été construit d'après les équations de fonctionnement déjà étudiées dans les chapitres précédents.

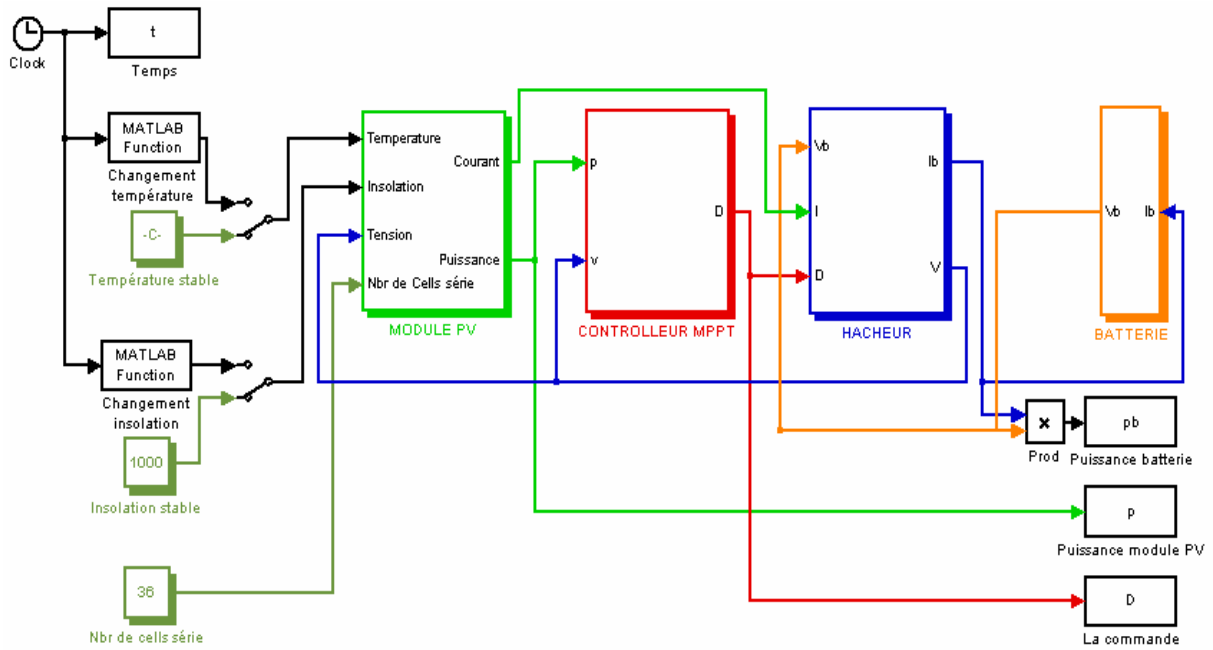


Figure V.6. Schéma du système global de simulation.

V.5.1.1. Module PV

Le sous bloc qui représente le module PV est détaillé par la figure V.7 qui suit :

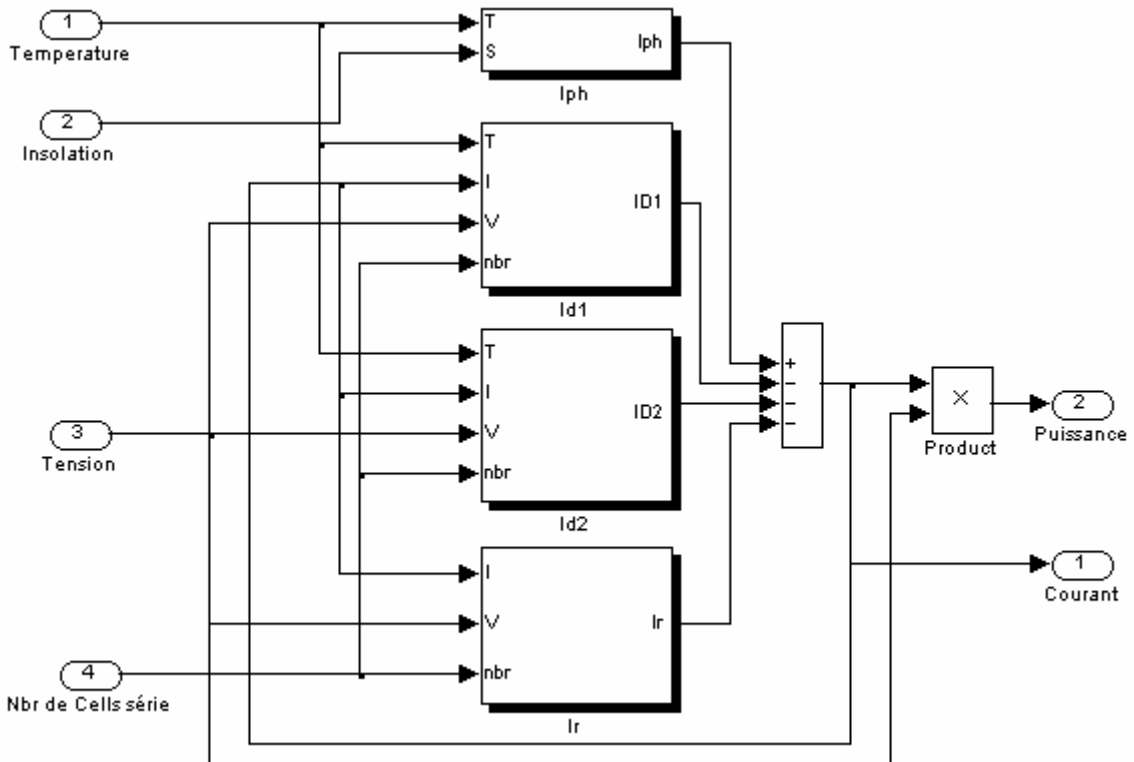


Figure V.7. Bloc de simulation du module PV.

Ce bloc a été modélisé d'après les équations mathématiques du modèle du panneau PV (I.7), décrites dans la section (I.4).

On donne les données suivantes [5]:

$$I_{ph}|_{(T=298.K)} = 3.25A$$

$$R_p = 30\Omega$$

$$R_s = 15 \cdot 10^{-3} \Omega$$

$$E_g = 1.1 \text{ eV}$$

$$n_1 = 1; n_2 = 2$$

$$k = 1.380 \cdot 10^{-23} \text{ J/K}$$

$$q = 1.602 \cdot 10^{-19} \text{ C.}$$

V.5.1.2. Batterie Pb-acide

Le modèle de batterie Pb-Acide qui a été choisi pour les simulations a été déjà étudié dans la section (I.8). C'est le modèle des équations mathématiques : (I.13) et (I.14), associées avec les données suivantes [8]:

$$R_{bs} = 0.0013\Omega$$

$$R_{b1} = 2.84\Omega$$

$$R_{bp} = 10e3\Omega$$

$$C_{b1} = 2.5mF$$

$$C_{bp} = 2 / (125^2 - 90^2) * 45 * 9 * 12 * 3600 = 4.6501kF.$$

Ces équations peuvent être implantées directement dans un bloc Simulink de fonction de transfert présenté par la figure V.8 :

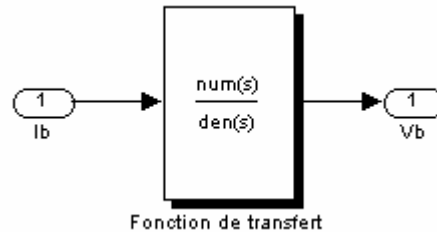


Figure V.8. Bloc de simulation de la batterie.

V.5.1.3. Hacheurs

Pour les simulations que nous avons effectuées nous avons utilisé les deux hacheurs Buck et Boost.

A. Hacheur Boost

Les équations du modèle moyen mathématique du hacheur Boost (I.39) ont été étudiées dans la section (I.9.2.2). On les représente directement par le schéma Simulink de la figure (V.9) :

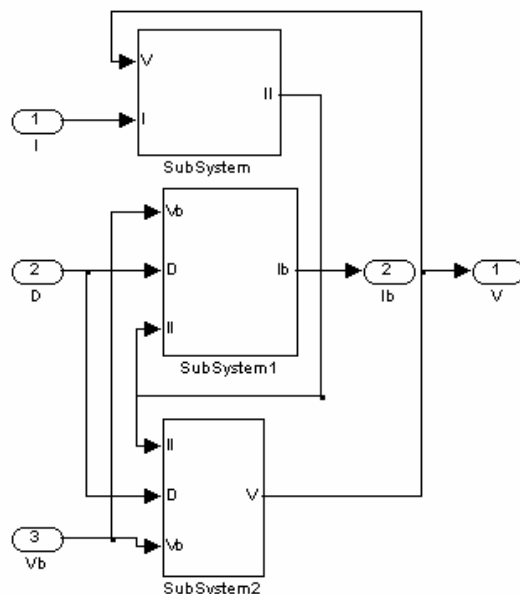


Figure V.9. Schéma Simulink représentant le hacheur Boost.

B. Hacheur Buck

Les équations du modèle moyen mathématique du hacheur Buck (I.21) ont été étudiées dans la section (I.9.1.2). On les représente directement par le schéma Simulink de la figure (V.10) :

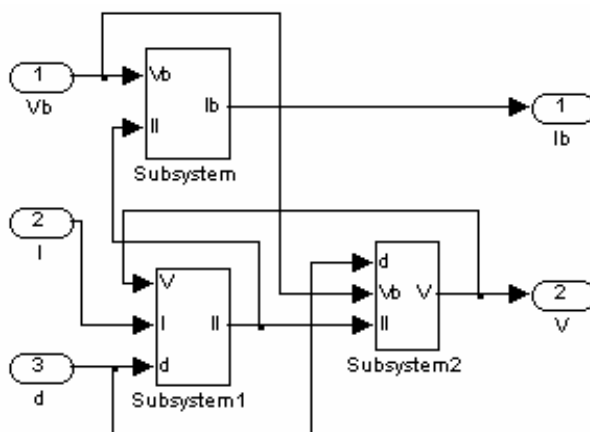


Figure V.10. Schéma Simulink représentant le hacheur Buck.

On associe pour les deux types de hacheurs les valeurs numériques suivantes [5]:

$$C_1 = 5,6\text{mF}$$

$$C_2 = C_1$$

$$L = 3,5\text{mH}$$

V.5.1.4. Contrôleurs MPPT

Comme déjà discuté, on a effectué des simulations en comparant les résultats des deux types de contrôleur : Le contrôleur MPPT flou et le contrôleur MPPT flou optimisé par les AGs (MPPT AGF).

A. Contrôleur MPPT flou

Le contrôleur MPPT flou qui a été discuté dans la section (V.3) est programmé et simulé en utilisant la boîte à outil Fuzzy sous Simulink.

B. Contrôleur MPPT AGF

Le contrôleur MPPT AGF qui a été discuté dans la section (V.4) est programmé et simulé en utilisant la boîte à outil Fuzzy sous Simulink représenté globalement par la figure V.11 suivante :

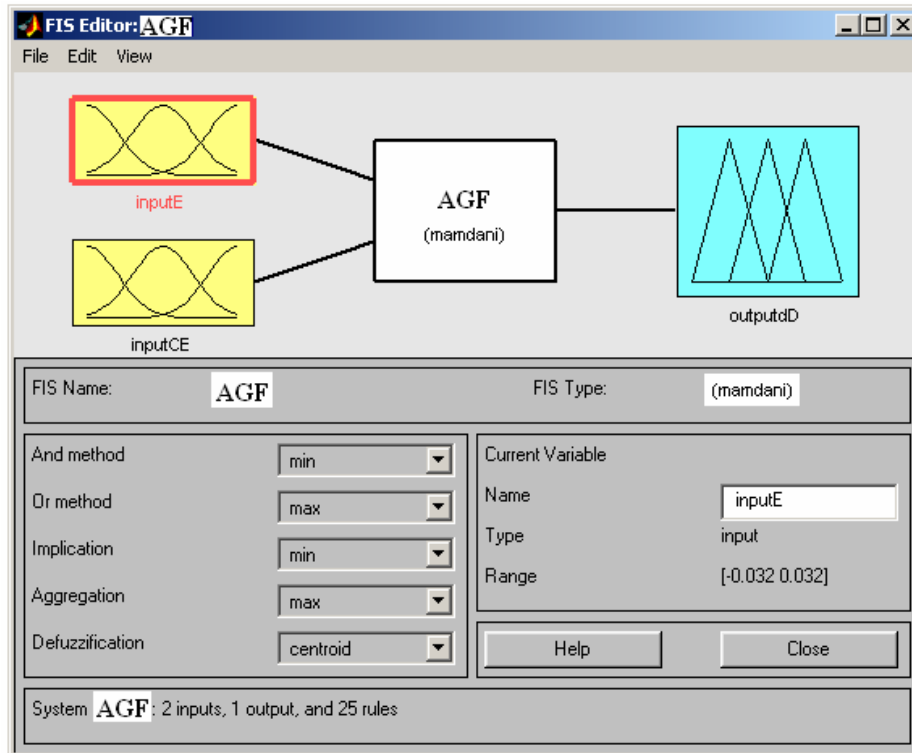


Figure V.11. Schéma général du contrôleur MPPT AGF.

Remarque

Pour la discussion des résultats de simulation on a choisi d'utiliser un hacheur boost, pour cela on utilise les éléments suivants :

- ✓ 14 modules solaires en parallèle, chaque module contenant 36 cellules en série.
- ✓ 9 batteries au plomb en série, chaque batterie a une tension de fonctionnement de 12V.

Ensuite on a donné les résultats des simulations du système avec un hacheur buck et on a utilisé :

- ✓ 1 module solaire, contenant 36 cellules en série.
- ✓ Une batterie au plomb de même caractéristique que la précédente.

V.5.2. Résultats de simulation du système MPPT en utilisant un hacheur boost

V.5.2.1. Test de performances

Pour faire une étude des performances des deux systèmes contrôlés : MPPT flou et MPPT AGF, et voir la différence entre eux on a effectué un test où les deux systèmes subissent les mêmes conditions environnementales standards, c'est-à-dire un éclairage solaire constant qui est égal à 1000 W/m^2 et une température égale à 25°C .

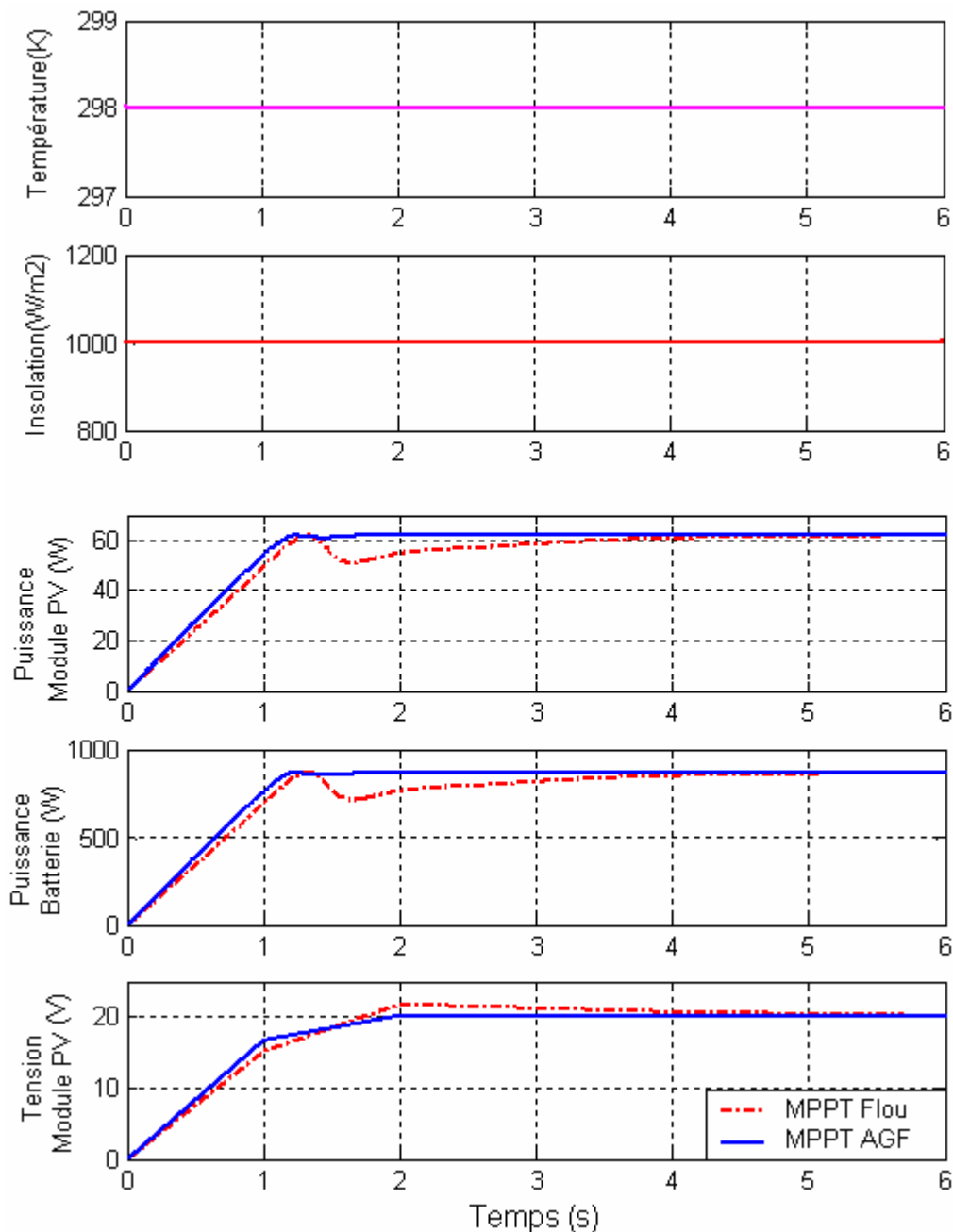


Figure V.12. Variation de la puissance du module PV, de la puissance de la batterie et de la tension du module, des deux contrôleurs MPPT flou et MPPT AGF, sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

La figure V.12 montre la réponse des deux contrôleurs MPPT flou et MPPT AGF avec conditions environnementales standards. Cette figure nous montre la différence entre la rapidité des deux contrôleurs. Le contrôleur optimisé par les AGs est plus rapide que le contrôleur flou original, qui lui-même a prouvé une bonne rapidité [7]. Et cela s'exprime bien sur les courbes de caractéristiques P-V illustrés sur les figures V.13 et V.14.

Les deux contrôleurs ont donné de très bonnes améliorations contre les ondulations en régime permanent, elles ont presque été éliminées.

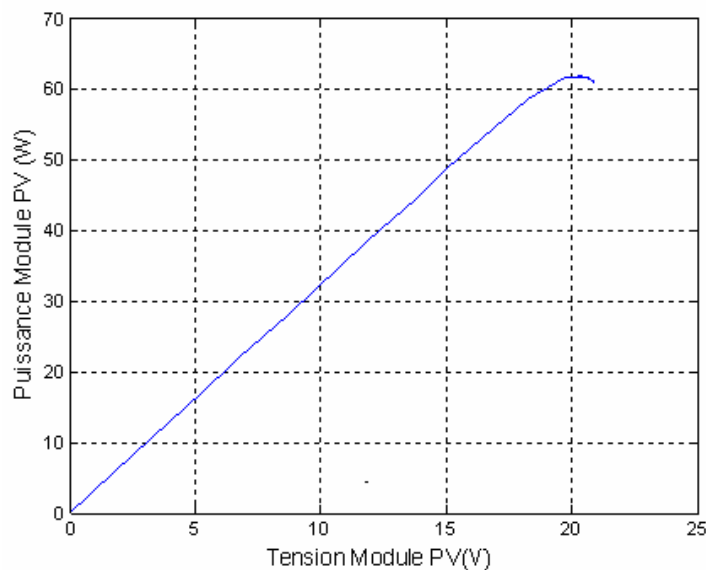


Figure V.13 Courbe de simulation de la caractéristique P-V d'un module PV contrôlé par le contrôleur MPPT AGF, sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

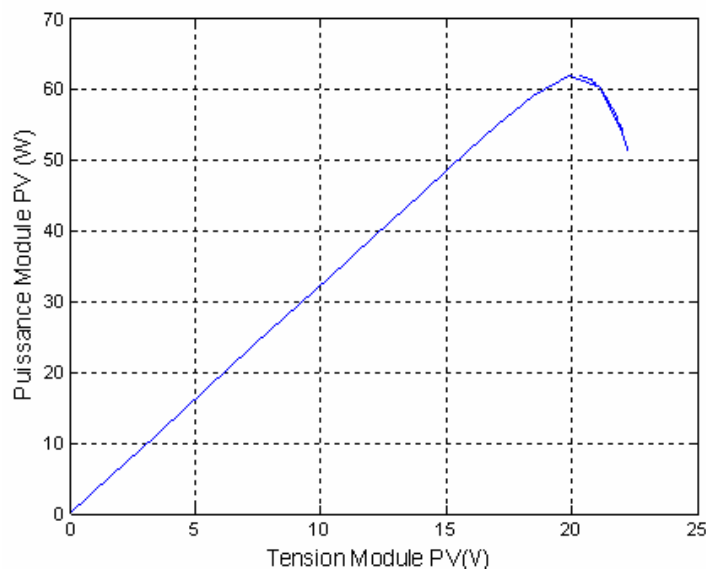


Figure V.14 Courbe de simulation de la caractéristique P-V d'un module PV contrôlé par le contrôleur MPPT Flou, sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

Les figures V.13 et V.14 montrent que le contrôleur MPPT AGF cherche dans le bon sens et atteint facilement et directement le PPM, tandis que le contrôleur MPPT Flou perd un peu de temps avant d'atteindre le PPM, cela est dû aux erreurs effectuées lors du choix des valeurs d'intervalle de fonctions d'appartenance.

V.5.2.2. Tests de robustesse

Pour faire une étude de la robustesse des deux contrôleurs vis-à-vis des différentes conditions environnementales on effectue les tests suivants :

Nous soumettons les systèmes contrôlés à différentes conditions environnementales :

- ✓ d'abord on maintient une température constante ($T=25^{\circ}\text{C}$) et on varie l'éclairement solaire suivant une variation rapide ou bien lente, et dans les deux sens de variation : soit une augmentation ou bien une diminution.
- ✓ Ensuite on maintient l'éclairement solaire à une valeur fixe (1000 W/m^2) et on varie la valeur de la température suivant une variation rapide ou bien lente, et dans les deux sens de variation : soit une augmentation ou une diminution.
- ✓ Et enfin on soumet les deux contrôleurs MPPT à des changements aléatoires des variables de température et de l'ensoleillement.

On commence par la figure V.15 qui nous montre le comportement des deux contrôleurs MPPT flou et MPPT AGF, pour une température fixe et une augmentation rapide linéaire de l'ensoleillement de 400 à 1000 W/m^2 sur une période de deux secondes. L'augmentation de l'ensoleillement implique une augmentation de la puissance.

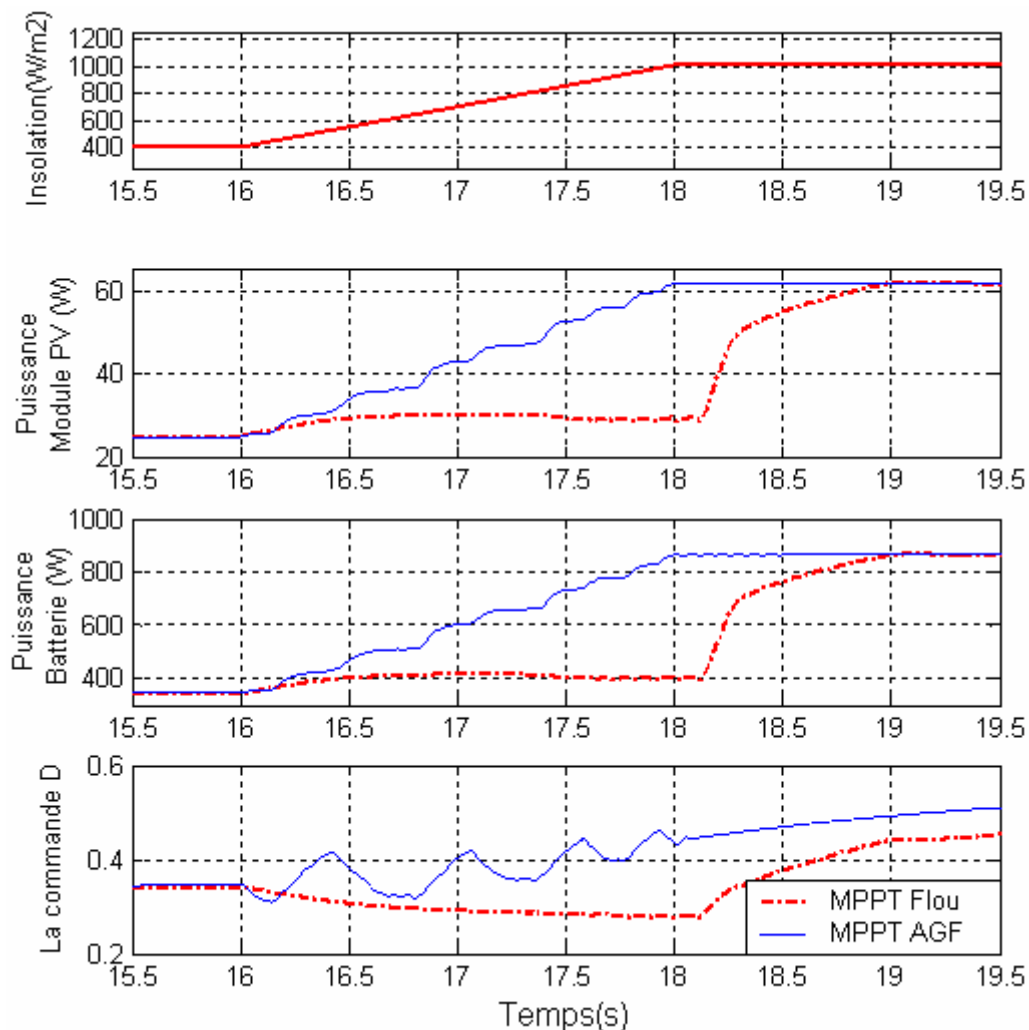


Figure V.15. Réponses des deux contrôleurs MPPT Flou et MPPT AGF pour une augmentation rapide d'ensoleillement de 400W/m^2 à 1000 W/m^2 en 2 secondes et une température constante de 25°C .

Les résultats obtenus lors de ce test montrent que le contrôleur MPPT AGF suit la consigne (variation de l'ensoleillement) avec de très faibles fluctuations tandis que le contrôleur MPPT flou a un retard.

Dans le cas du test présenté par la figure V.16, on effectue une augmentation lente de l'ensoleillement de 700W/m^2 à 800W/m^2 qui dure 120 sec. On remarque d'après cette figure que les deux contrôleurs MPPT donnent des résultats presque identiques. Les deux contrôleurs présentent une très bonne poursuite.

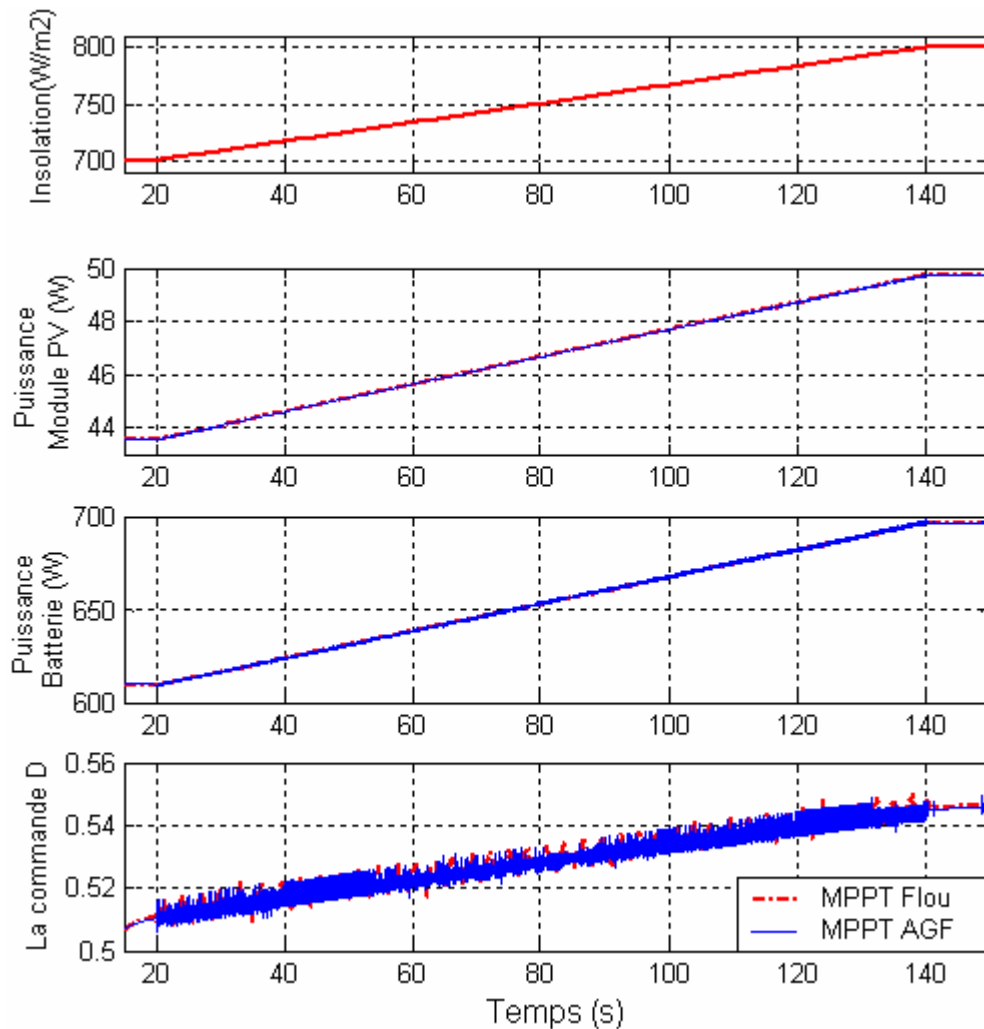


Figure V.16. Réponse des deux contrôleurs MPPT Flou et MPPT AGF lorsqu'une augmentation lente de l'ensoleillement de 700W/m^2 à 800W/m^2 dans un intervalle de 120sec.

La même conclusion est tirée en soumettant les deux contrôleurs MPPT à une diminution lente de l'ensoleillement de 800 à 700 W/m^2 dans une période de temps égale à 120 secondes. (Figure V.17).

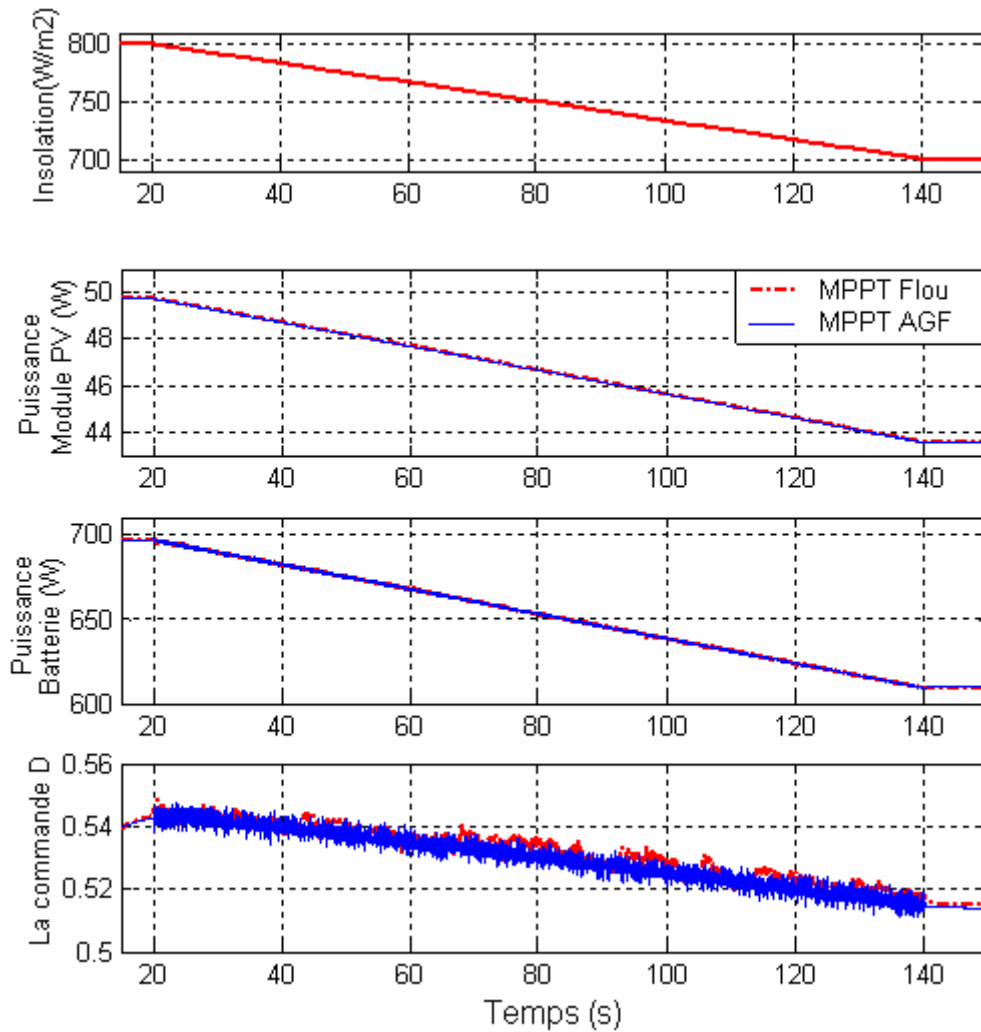


Figure V.17. Réponse des deux contrôleurs MPPT Flou et MPPT AGF pour une diminution lente de l'ensoleillement de 800 W/m^2 à 700 W/m^2 dans un intervalle de 120 s ($T=25^\circ \text{C}$).

Pour le test sur la variation de la température on effectue d'abord une augmentation rapide de 30°C à 60°C durant 2 secondes, figure V.18. L'augmentation de la température implique toujours une diminution de la puissance. Le contrôleur MPPT AGF présente une réponse en poursuite presque parfaite alors que le contrôleur MPPT flou est en retard aussi il présente quelques fluctuations dans la phase transitoire.

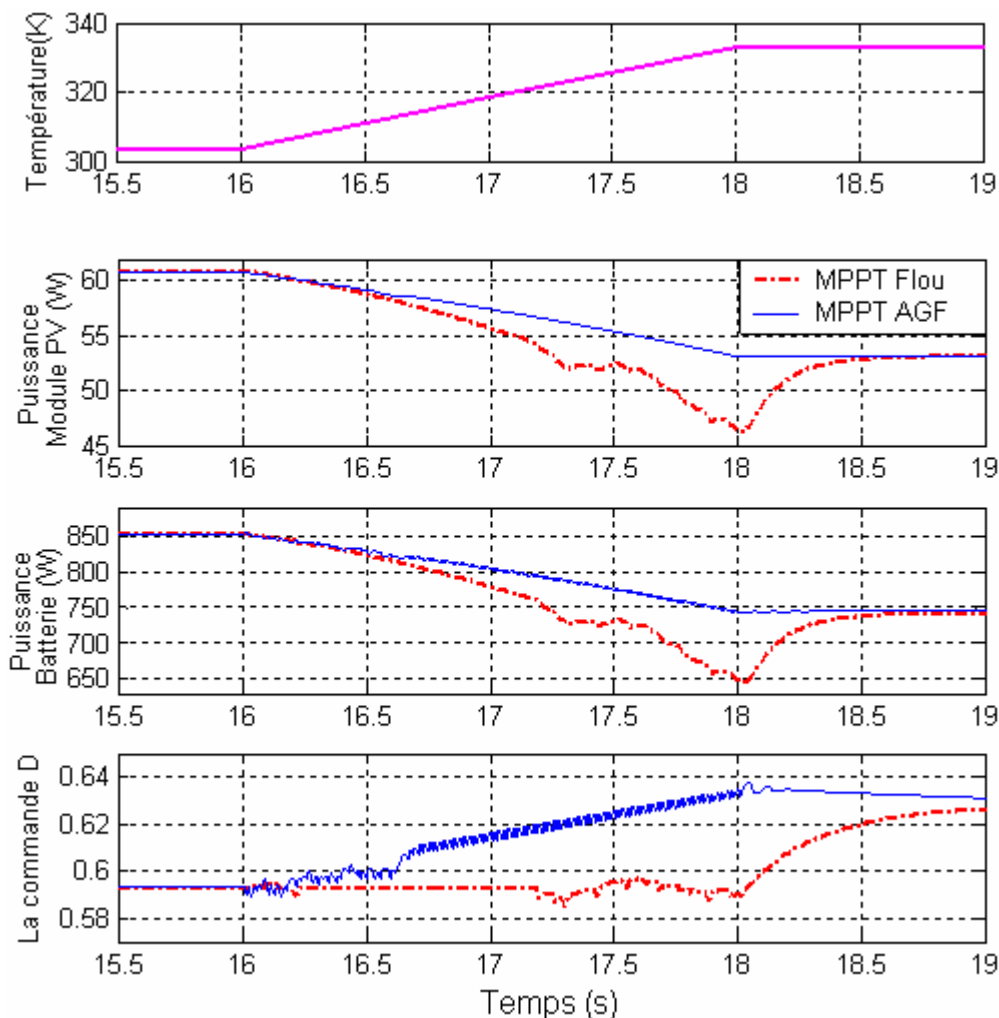


Figure V.18. Réponse des contrôleurs MPPT Flou et MPPT AGF pour une augmentation rapide de la température de 30°C à 60°C durant un temps de 2s avec un ensoleillement de 1000W/m².

La courbe de la figure V.19 qui est une courbe de la caractéristique P-V des deux contrôleurs MPPT Flou et MPPT AGF pour une augmentation rapide de la température de 15°C à 60°C sur une période de 2sec, sous un ensoleillement de 1000W/m². Cette courbe montre que le contrôleur MPPT flou effectue des variations avant d'atteindre le nouveau MPP, alors que le contrôleur MPPT AGF tend directement vers ce MPP. Ce qui prouve la recherche qui s'effectue dans le bon sens ainsi que la rapidité du contrôleur MPPT AGF.

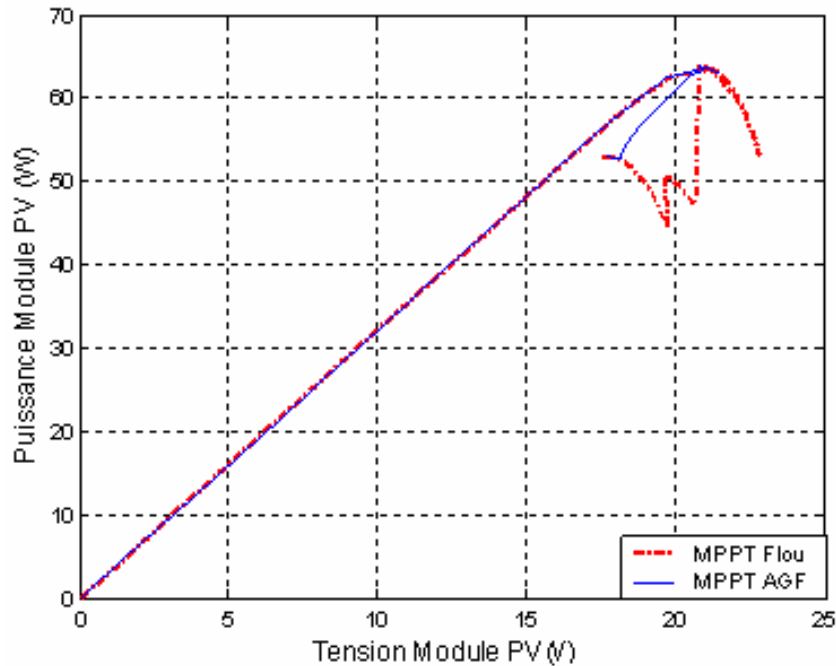


Figure V.19. Courbe de la caractéristique P-V des contrôleurs MPPT Flou et MPPT AGF pour une augmentation rapide de la température de 15°C à 60°C durant un temps de 2s avec un ensoleillement de 1000W/m².

Toujours pour le test de l'augmentation de la température mais dans un intervalle de temps qui est un peu plus long 120 secondes. Les résultats obtenus par ce test sont présentés sur la figure V.20. Ils ne sont pas trop différents pour les deux contrôleurs MPPT car tous les deux effectuent une très bonne poursuite.

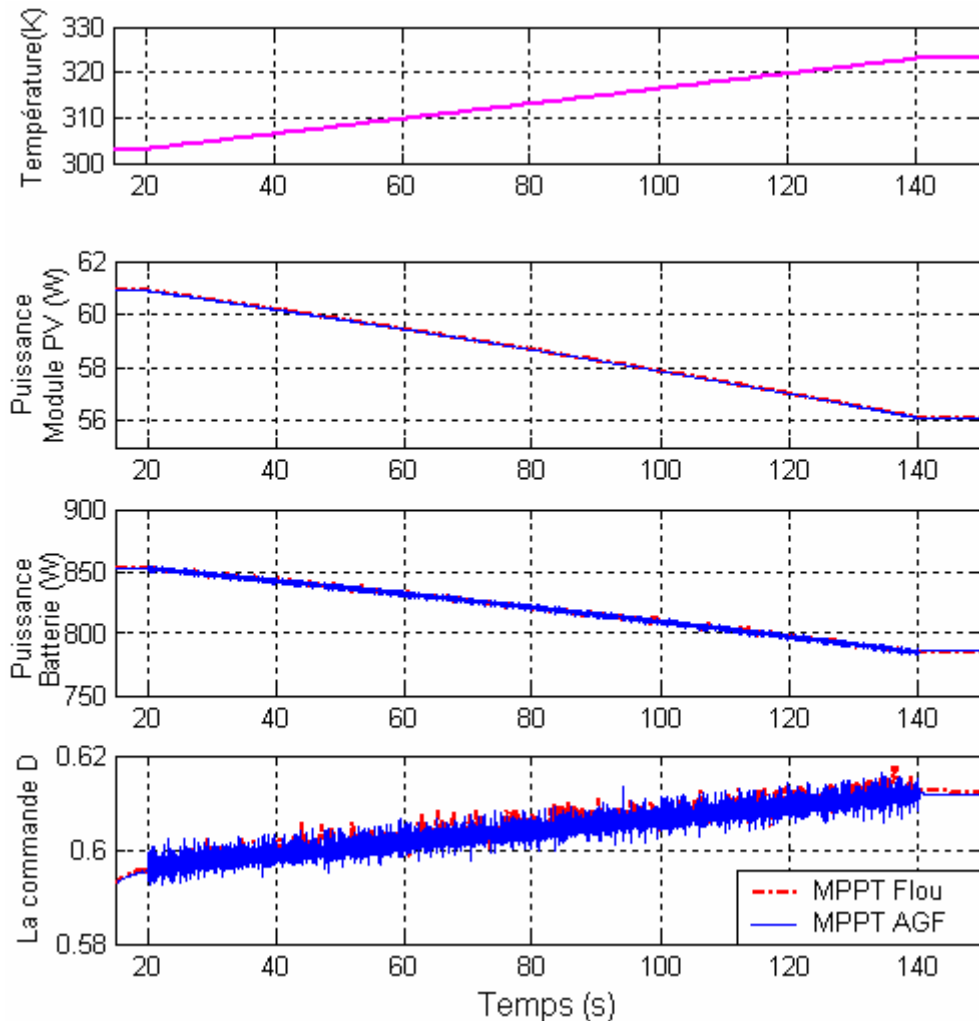


Figure V.20. Réponse des deux contrôleurs MPPT Flou et MPPT AGF pour une augmentation lente de la température de 30°C à 50°C durant un temps de 120s avec un ensoleillement de 1000W/m².

Une diminution rapide de la température de la valeur 60°C à 30°C durant un temps de 3 sec est effectuée et les résultats sont présentées par la figure V.21. Cette diminution rapide produit une augmentation de la puissance mais avec une poursuite qui est parfaite dans la zone transitoire pour le contrôleur MPPT AGF tandis que le MPPT flou est toujours en retard.

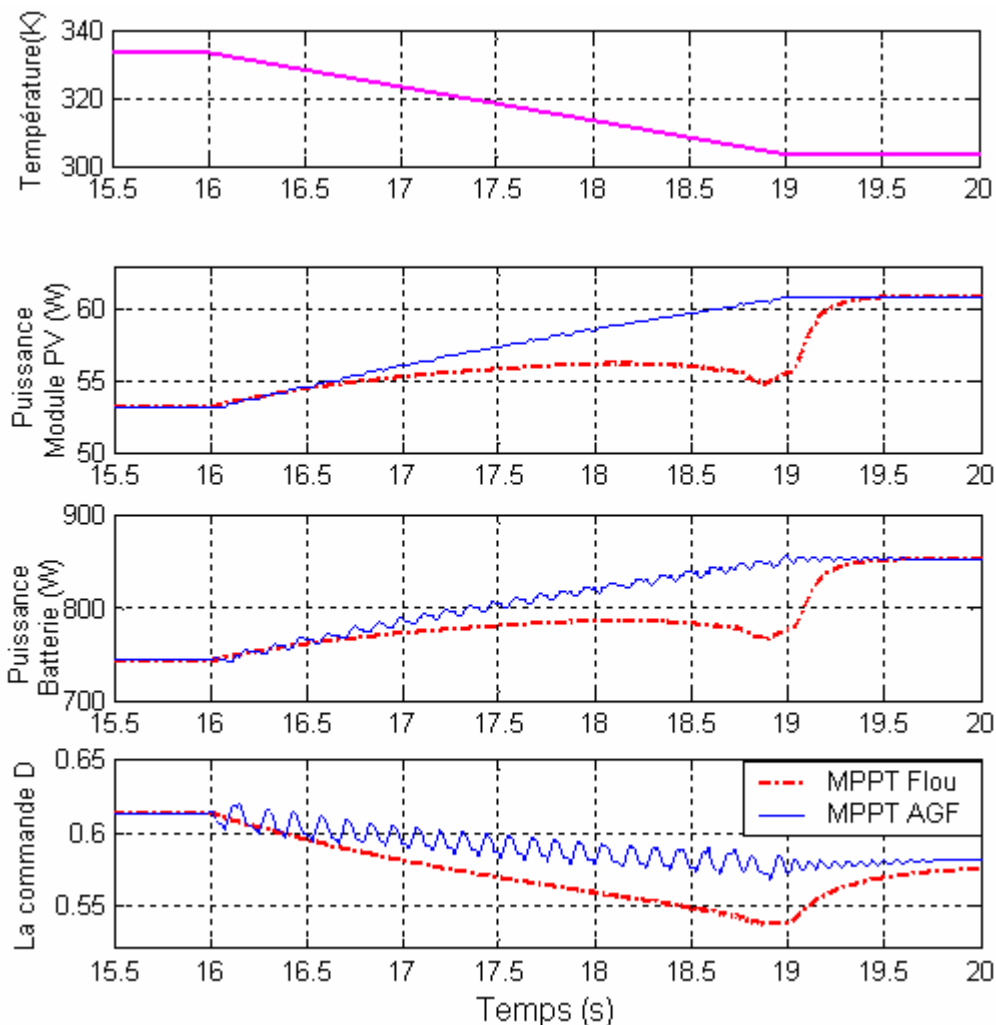


Figure V.21. Réponse des contrôleurs MPPT Flou et MPPT AGF pour une diminution rapide de la température de 60°C à 30°C durant un temps de 3sec avec un ensoleillement de $1000\text{W}/\text{m}^2$.

La courbe de la figure V.22 est une courbe de la caractéristique P-V des deux contrôleurs MPPT Flou et MPPT AGF pour une diminution rapide de la température de 50°C à 30°C durant un temps de 2s de la seconde 18 à la seconde 16, avec un ensoleillement de 1000W/m². Cette courbe nous montre toujours la recherche qui s'effectue rapidement avec le contrôleur MPPT AGF par rapport au contrôleur MPPT Flou.

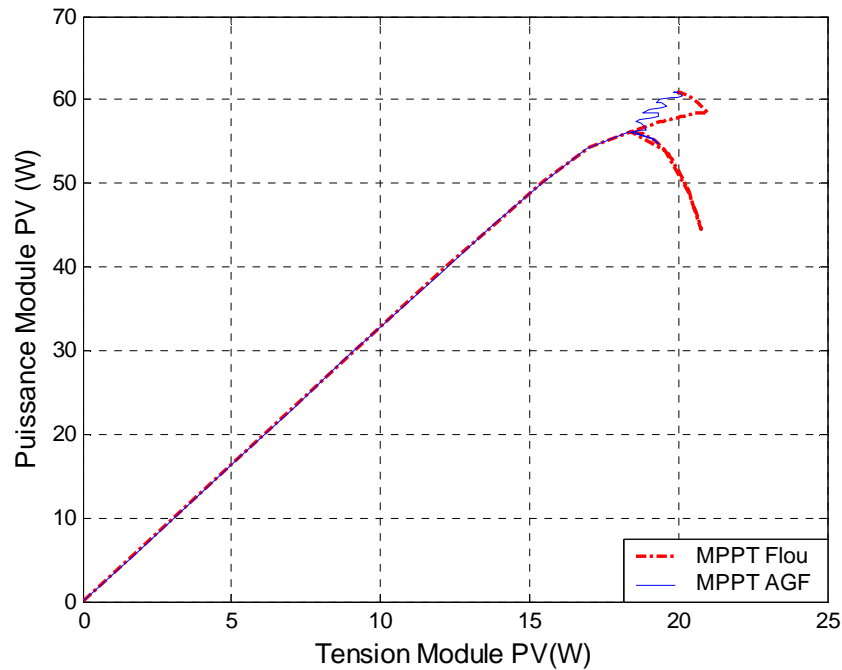


Figure V.22. Courbe de la caractéristique P-V des contrôleurs MPPT Flou et MPPT AGF pour une diminution rapide de la température de 50°C à 30°C durant un temps de 2s avec un ensoleillement de 1000W/m².

Mais pour une diminution lente de la température de 60°C à 30°C durant un temps de 120s présenté par la figure V.23, les deux contrôleurs MPPT étudiés se comporte parfaitement face à ce changement.

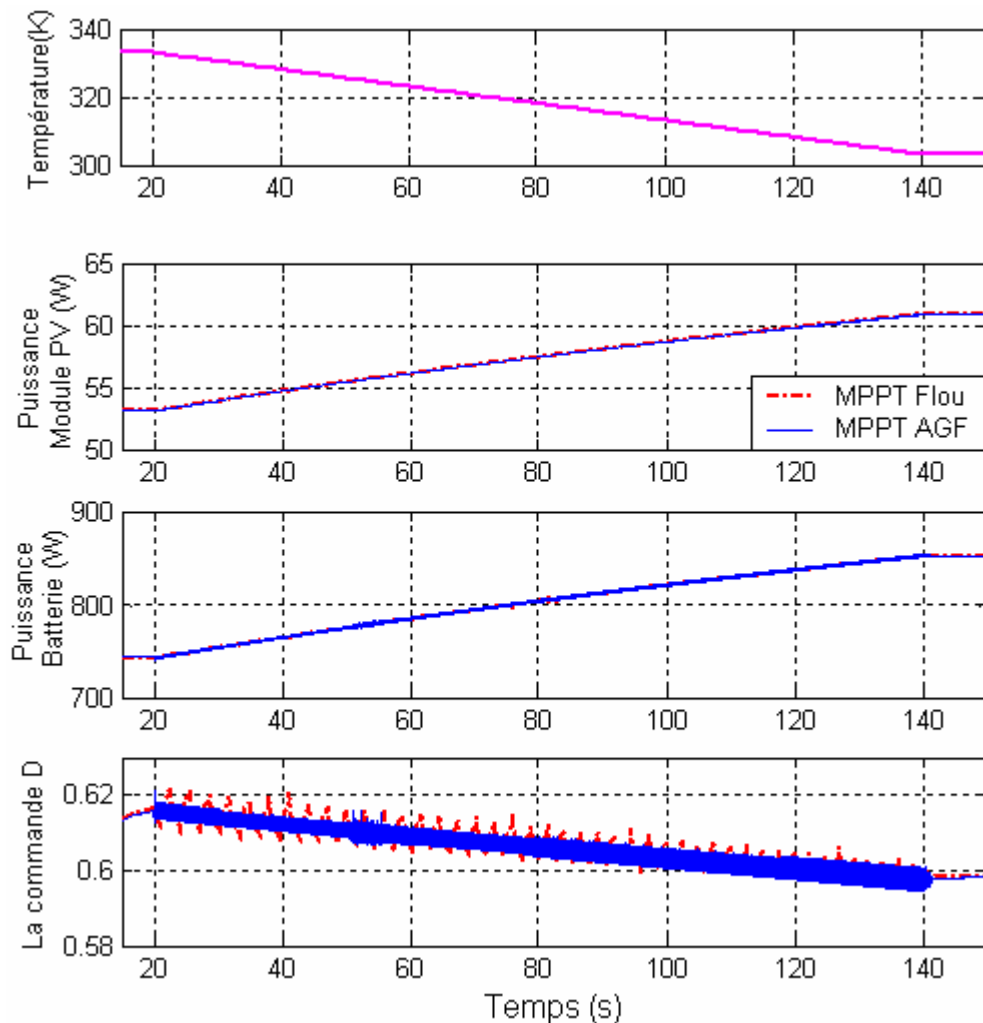


Figure V.23. Réponse des deux contrôleurs MPPT Flou et MPPT AGF pour une diminution lente de la température de 60°C à 30°C durant un temps de 120s avec un ensoleillement de 1000W/m².

Le dernier test présenté par la figure V.25, qui est une figure agrandie de la figure V.24, est un test où on soumet les deux systèmes contrôlés sous différentes conditions de changements aléatoires de la température et de l'ensoleillement.

La marge ou l'ensoleillement varie entre 200W/m² et 1000 W/m², tandis que la variation de la température s'effectue aléatoirement dans un intervalle de [30 60]°C.

Les résultats obtenus par cette expérience nous montre la bonne poursuite des deux contrôleurs mais avec une rapidité et un rendement plus élevé pour le contrôleur MPPT AGF par rapport au contrôleur MPPT flou.

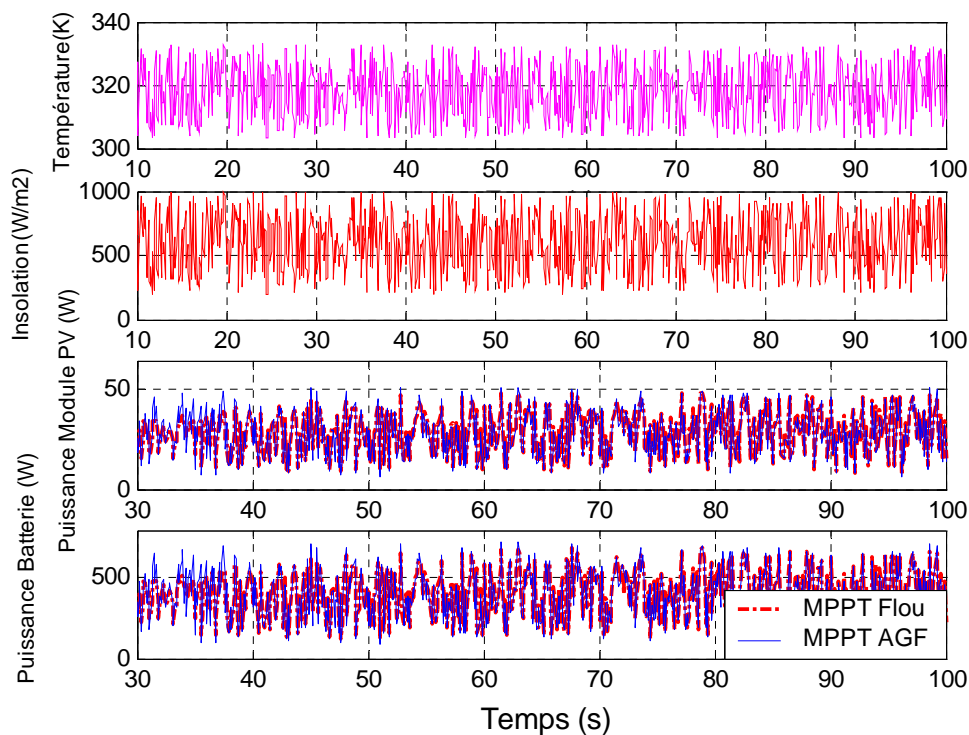


Figure V.24. Réponse des deux contrôleurs MPPT flou et MPPT AGF face aux changements aléatoire de l'ensoleillement et de la température.

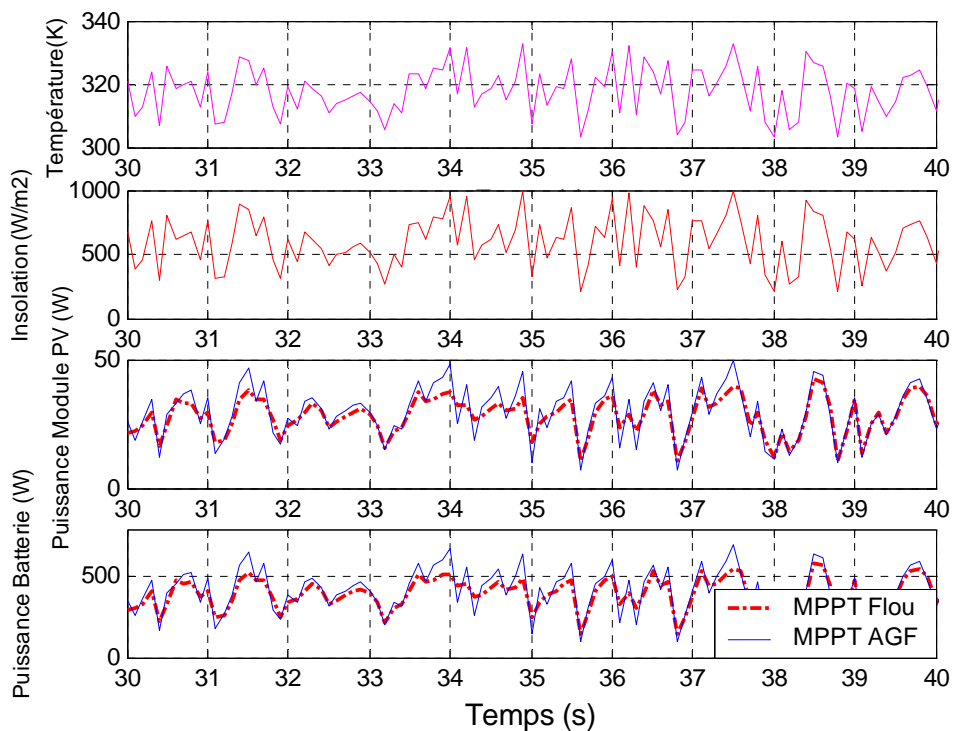


Figure V.25. Réponse agrandie des deux contrôleurs MPPT flou et MPPT AGF face aux changements aléatoire de l'ensoleillement et de la température.

Remarque 1

Lors de l'exécution du programme qui effectue la recherche du contrôleur MPPT AGF optimal on a fait un choix du nombre de générations qui est égal à 50 et le choix du nombre d'individus qui est égal à 100 individus. Ce choix a été effectué après plusieurs et différents tests ; par exemple si on commence par un nombre de générations de 1 et un nombre d'individus de 5, on aura un contrôleur optimal MPPT AGF représenté par les fonctions d'appartenance de la figure V.26 qui donne le résultat de la puissance donné par la figure V.27 :

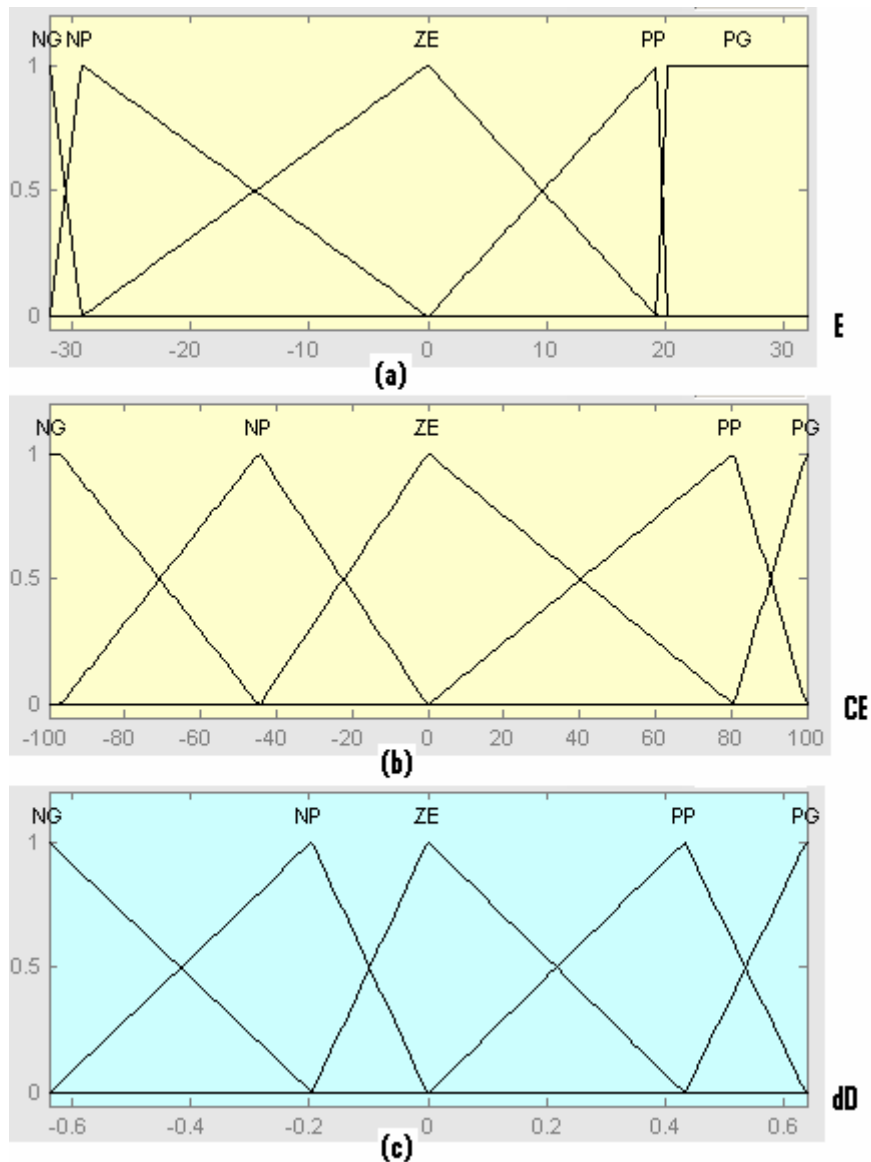


Figure V.26 Fonctions d'appartenance du CLF flou optimal obtenu avec un nombre de générations de 1 et un nombre d'individus de 5.

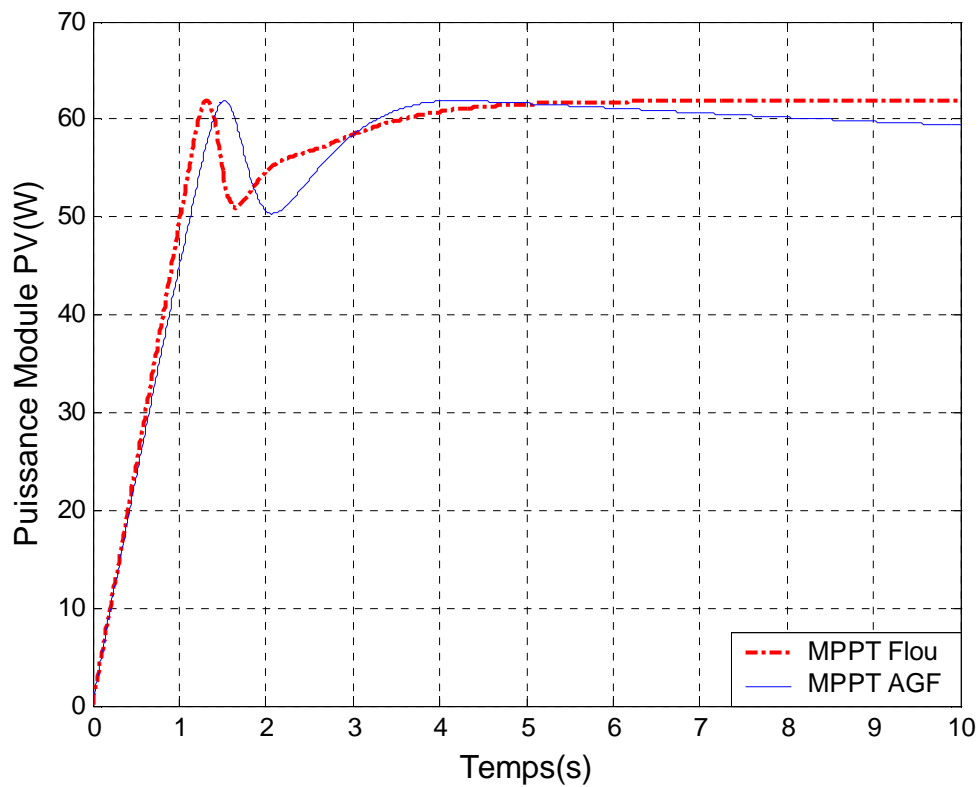


Figure V.27. Variation de la puissance du module PV, des deux contrôleurs MPPT flou et MPPT AGF (nombre de générations de 1 et nombre d'individus de 5), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

Les figures ci-dessous montrent le comportement du système PV commandé par le contrôleur MPPT AGF pour différents nombres de génération et d'individu :

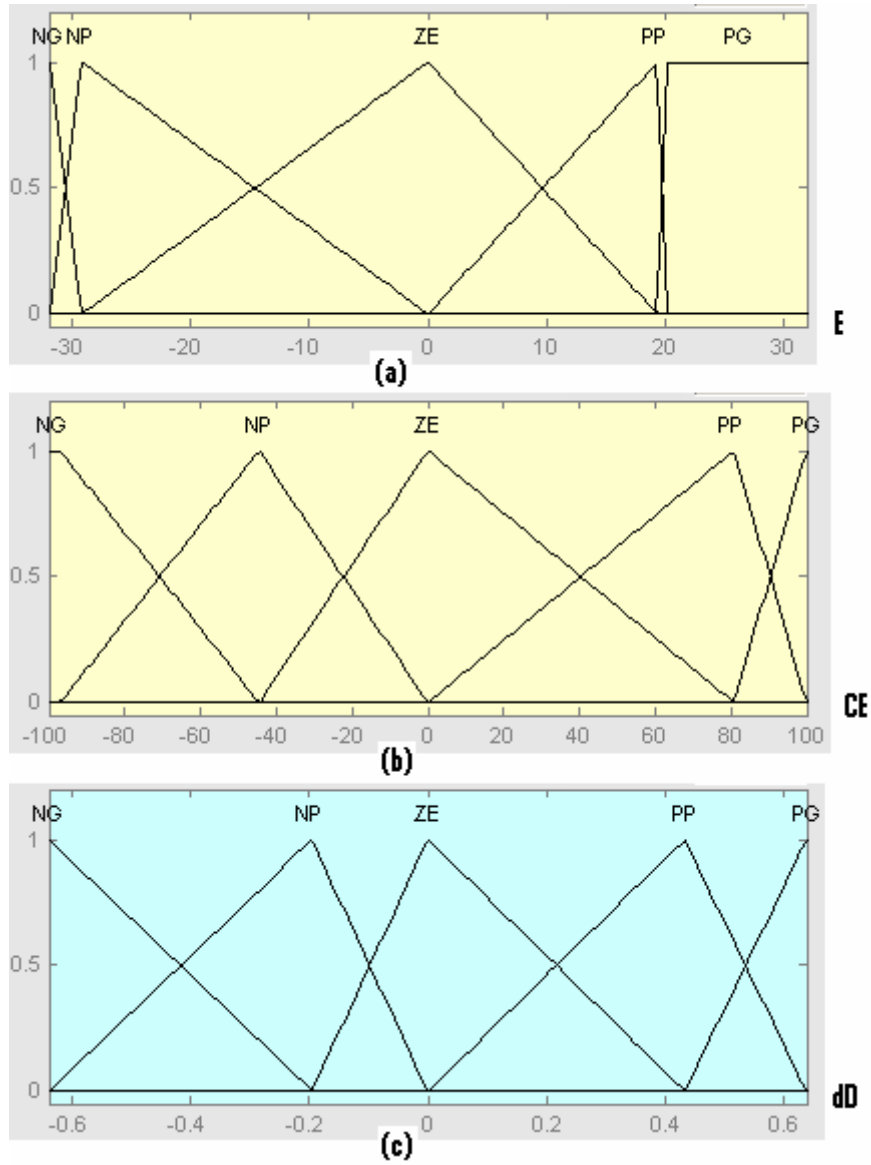


Figure V.28. Fonctions d'appartenances du CLF flou optimal obtenu avec un nombre de générations de 10 et un nombre d'individus de 5.

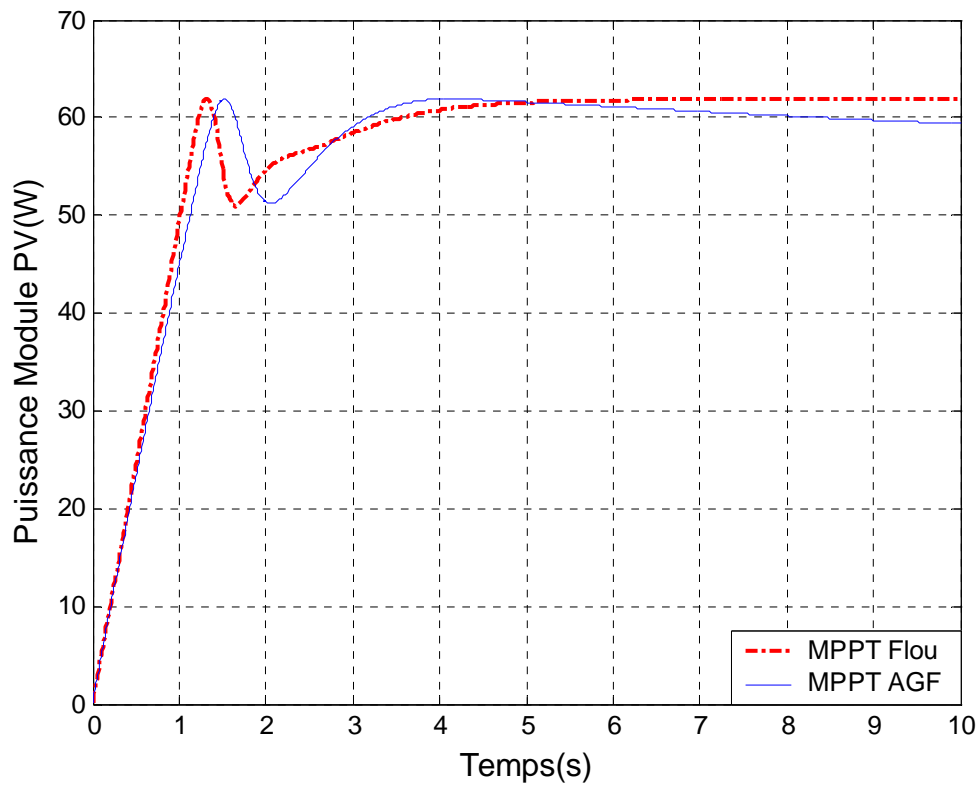


Figure V.29. Variation de la puissance du module PV, des deux contrôleurs MPPT flou MPPT AGF (nombre de générations 10 et nombre d'individus 5), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

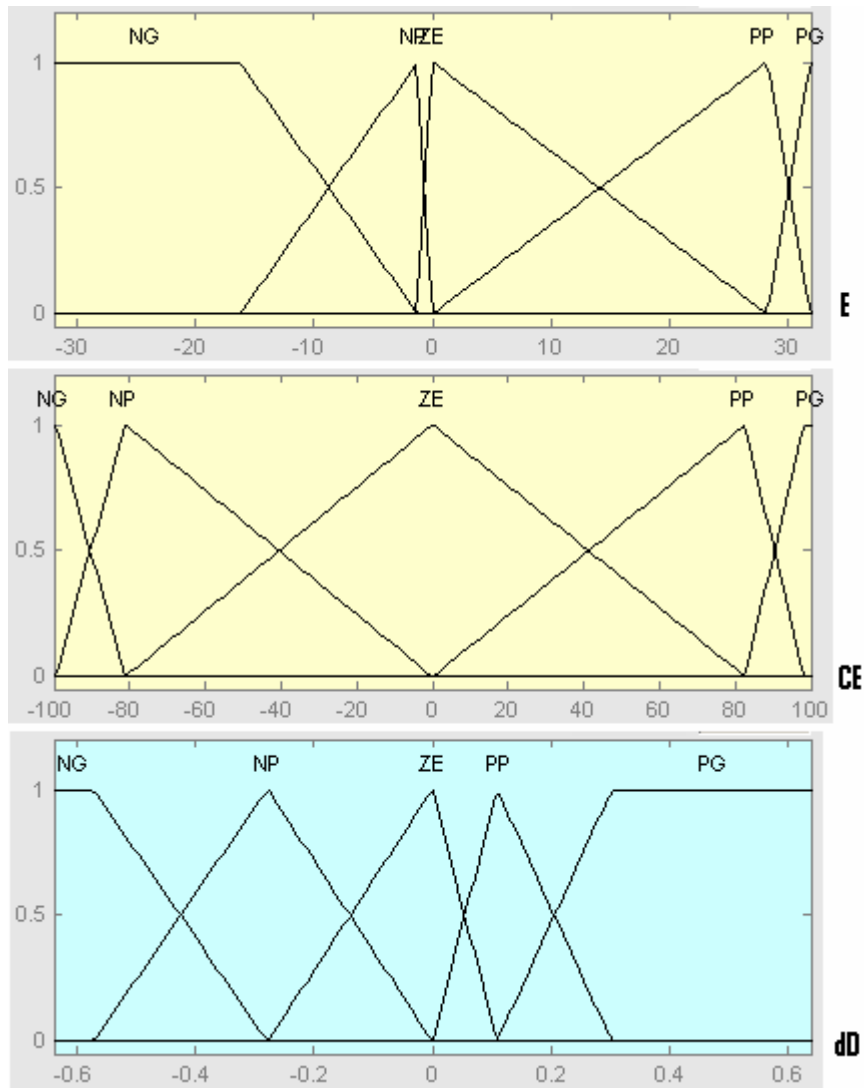


Figure V.30. Fonctions d'appartenances du CLF flou optimal obtenu avec un nombre de générations de 20 et un nombre d'individus de 20.

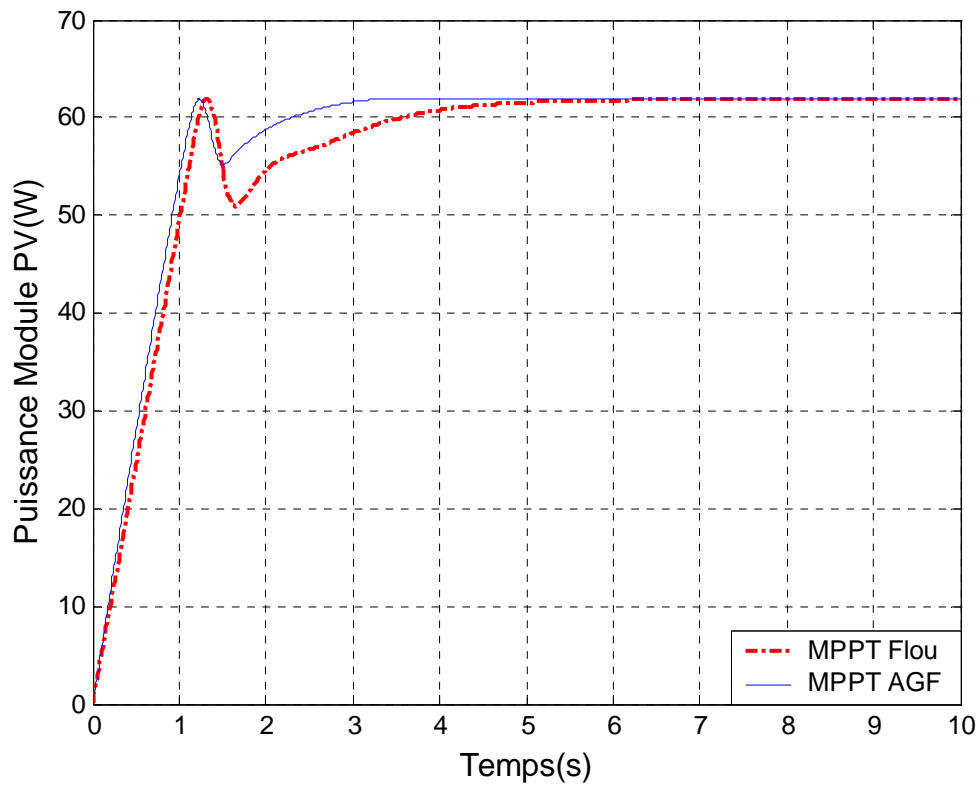


Figure V.31. Variation de la puissance du module PV, des deux contrôleurs MPPT flou et MPPT AGF (nombre de générations de 20 et nombre d'individu 20), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

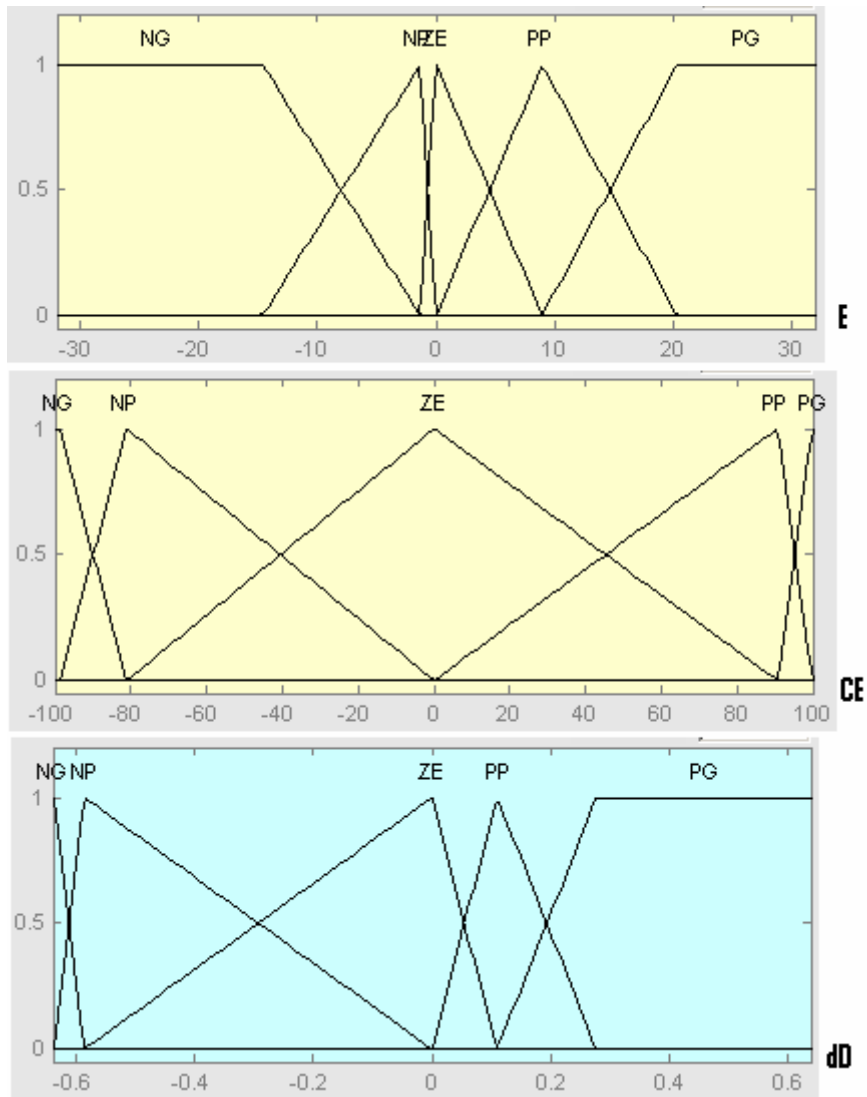


Figure V.32 Fonctions d'appartenances du CLF flou optimal obtenu avec un nombre de générations de 50 et un nombre d'individus de 20.

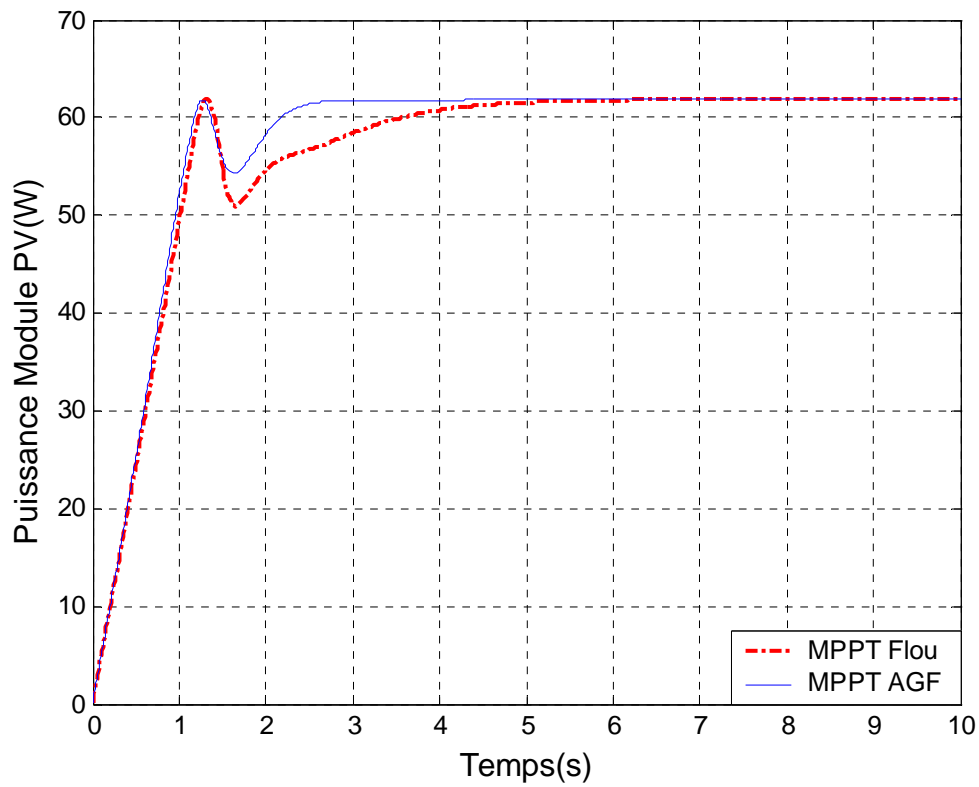


Figure V.33. Variation de la puissance du module PV, des deux contrôleurs MPPT flou et MPPT AGF (nombre de générations 50 et nombre d'individus 20), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

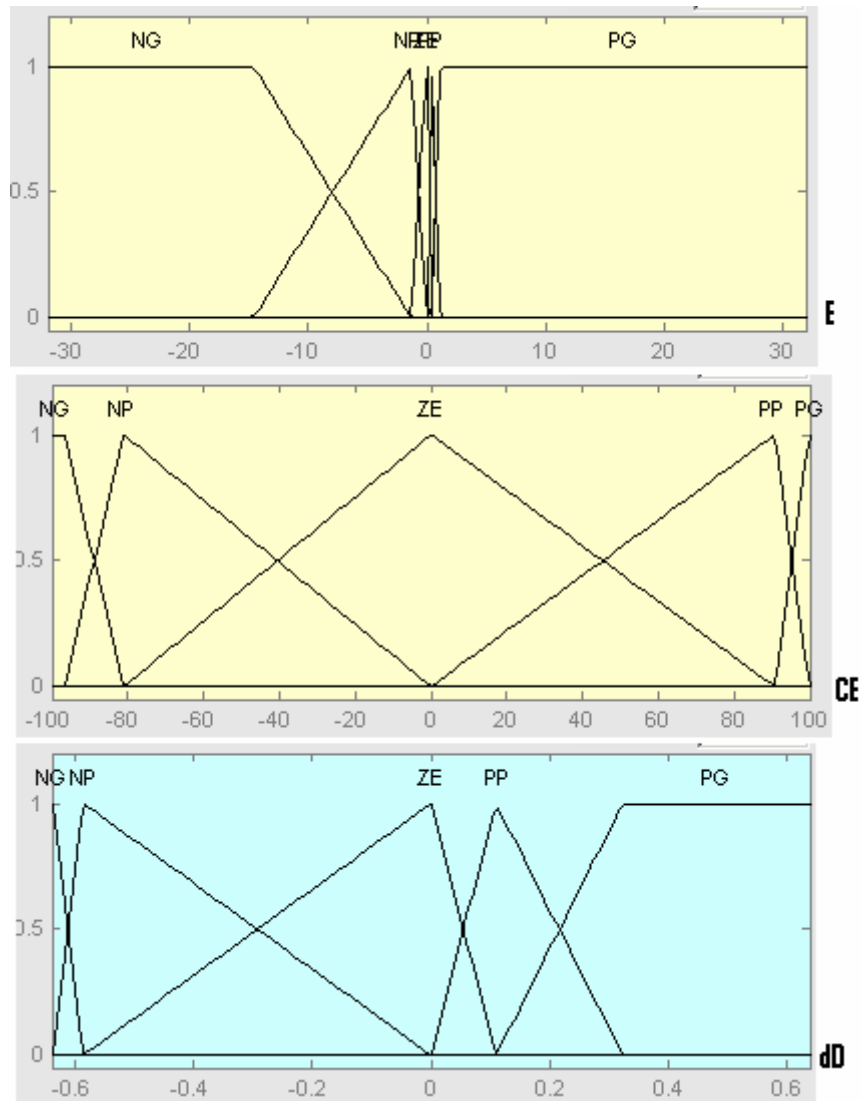


Figure V.34. Fonctions d'appartenances du CLF flou optimal obtenu avec un nombre de générations de 15 et un nombre d'individus de 70.

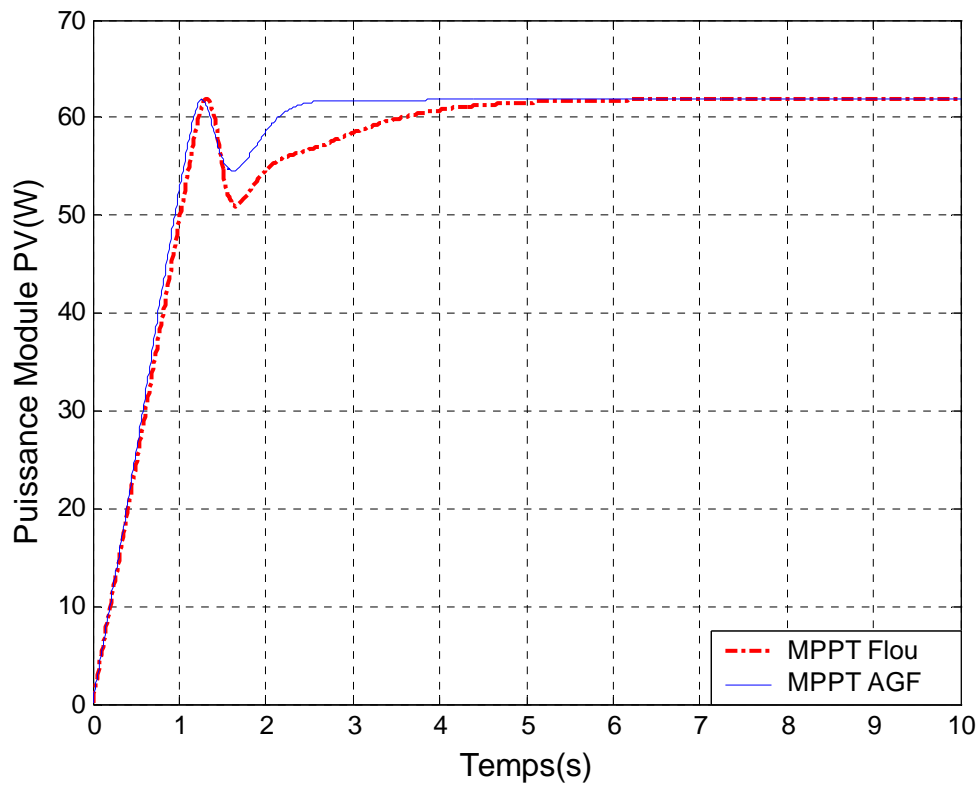


Figure V.35. Variation de la puissance du module PV, des deux contrôleurs MPPT flou et MPPT AGF (nombre de générations 15 et nombre d'individus 70), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

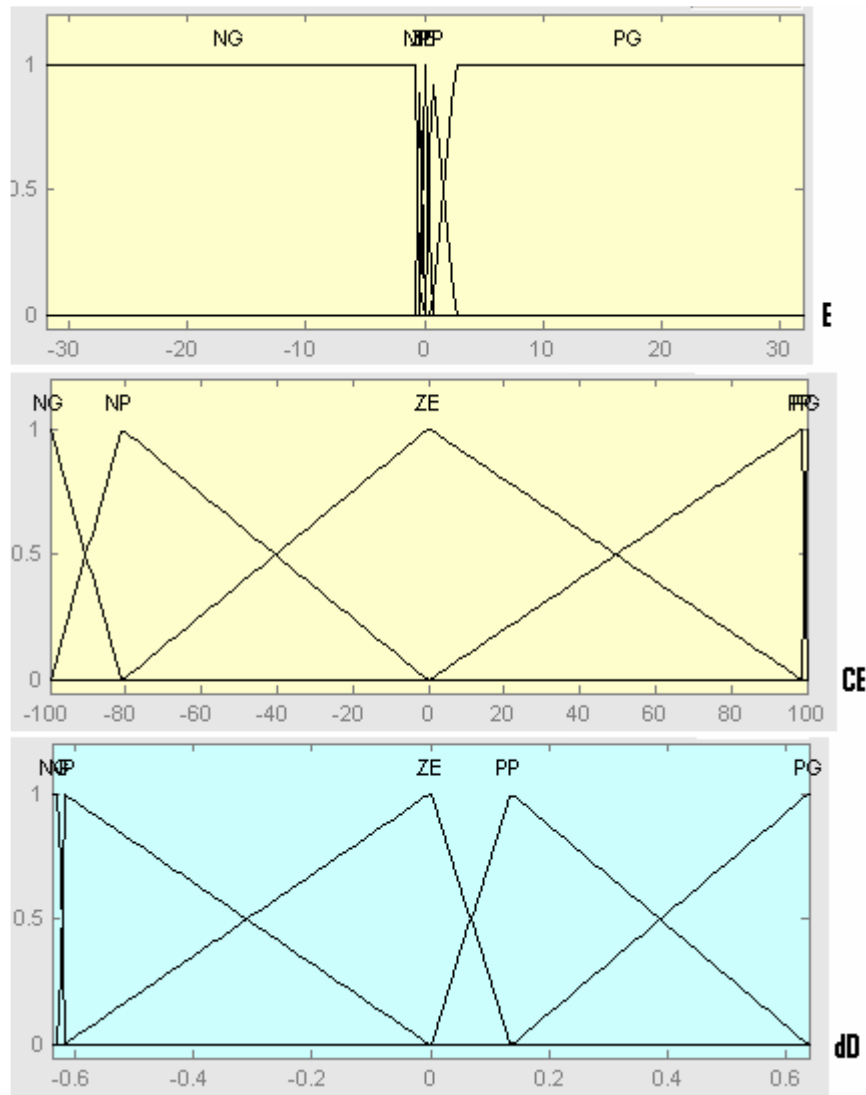


Figure V.36. Fonctions d'appartenances du CLF flou optimal obtenu avec un nombre de générations de 25 et un nombre d'individus de 70.

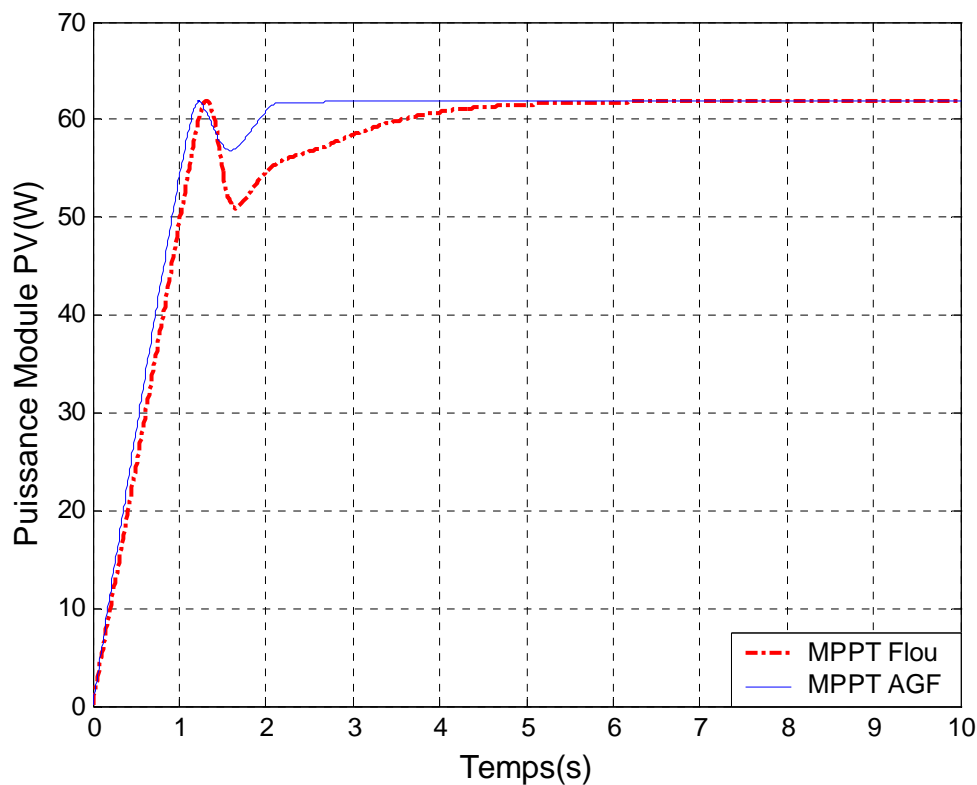


Figure V.37. Variation de la puissance du module PV, des deux contrôleurs MPPT flou et MPPT AGF (nombre de générations 25 et nombre d'individus 70), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

D'après les simulations présentées par les figures V.27, V.29, V.31, V.33, V.35 et V.37, on peut dire que lorsque que le nombre de générations ou le nombre d'individus augmente on obtient une amélioration de la sortie du système commandé par le contrôleur MPPT AGF.

Remarque 2

Un autre test a été fait avant le choix du contrôleur optimal obtenu dans la section V.4. Dans ce test on a fait l'optimisation du contrôleur MPPT Flou par les algorithmes génétiques avec un autre type de codage qui est le codage binaire (voir chapitre III). Pour cela on a utilisé les paramètres suivants :

- ✓ Nombre de bits=8
- ✓ Espace de recherche, qui est l'intervalle de variation de C , C' et C'' ; $[0.01 \ 0.99]$
- ✓ Le nombre de génération=20
- ✓ Nombre d'individu=100
- ✓ Probabilité de croisement $P_c=0.9$
- ✓ Probabilité de mutation $P_m=0.01$
- ✓ Sélection par tournoi
- ✓ Croisement à 1 point
- ✓ Mutation binaire

✓ Réinsertion par méthode élitiste

Pour cela on a obtenu des variations de la valeur minimale de la fonction objective (fitness) en fonction du nombre de générations qui sont données dans la Figure V.38. Nous remarquons que l'AG a convergé progressivement vers une solution optimale représentée par le meilleur individu de la population de la dernière génération.

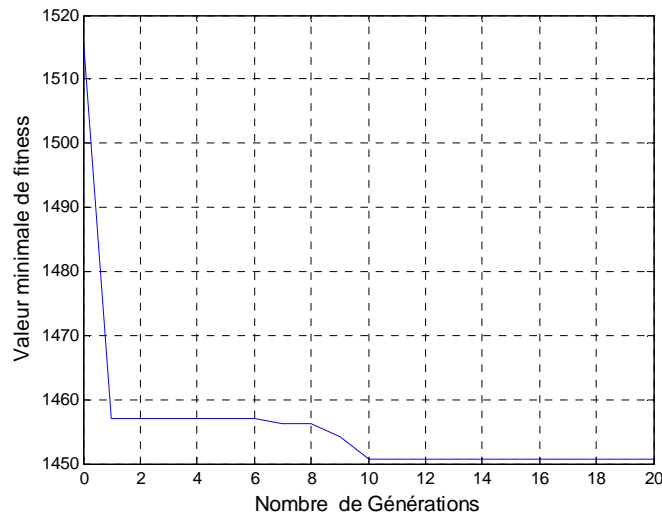


Figure V.38. Valeurs minimales de la fonction objective en fonction du nombre de générations pour un contrôleur MPPT AGF obtenu par le codage binaire.

La solution optimale obtenue par le codage binaire permet d'obtenir alors un contrôleur optimal MPPT AGF ayant les fonctions d'appartenances présentées par la figure V.39, et qui permet au système PV d'atteindre le MPP décrit par la figure V.40.

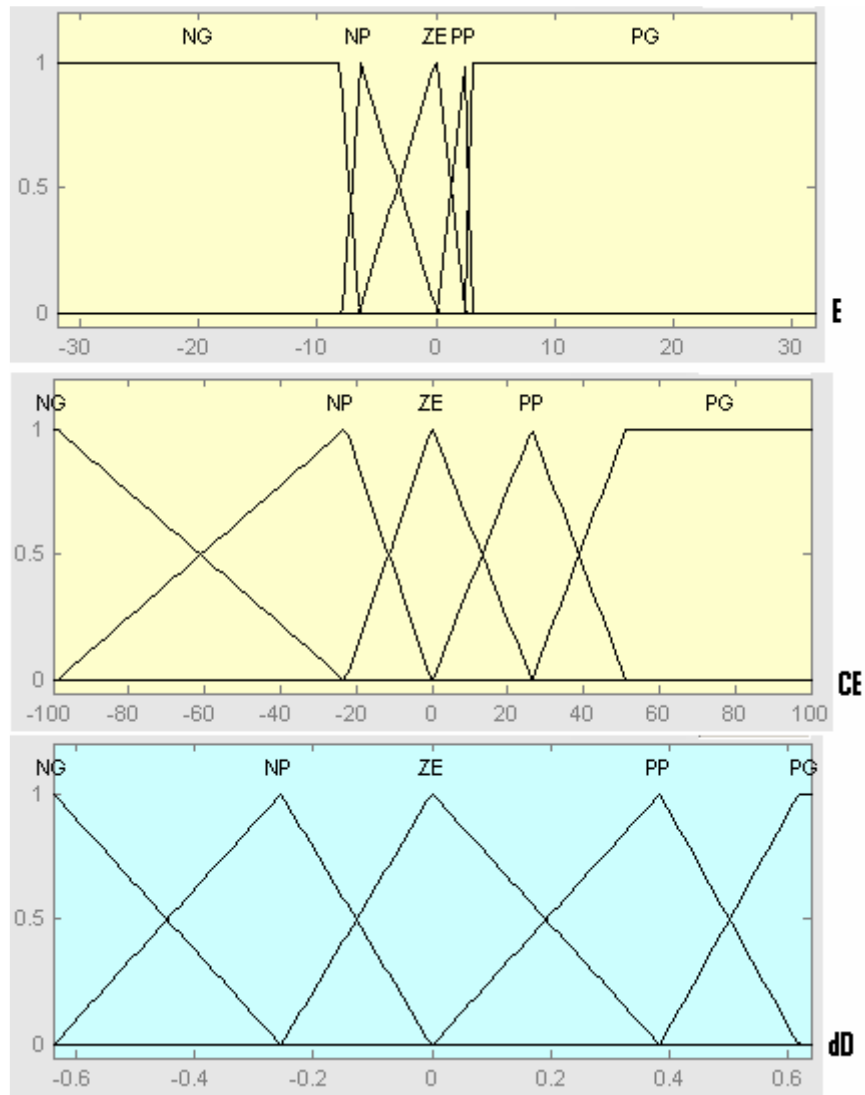


Figure V.39. Fonctions d'appartenances du CLF flou optimal obtenu avec un codage binaire.

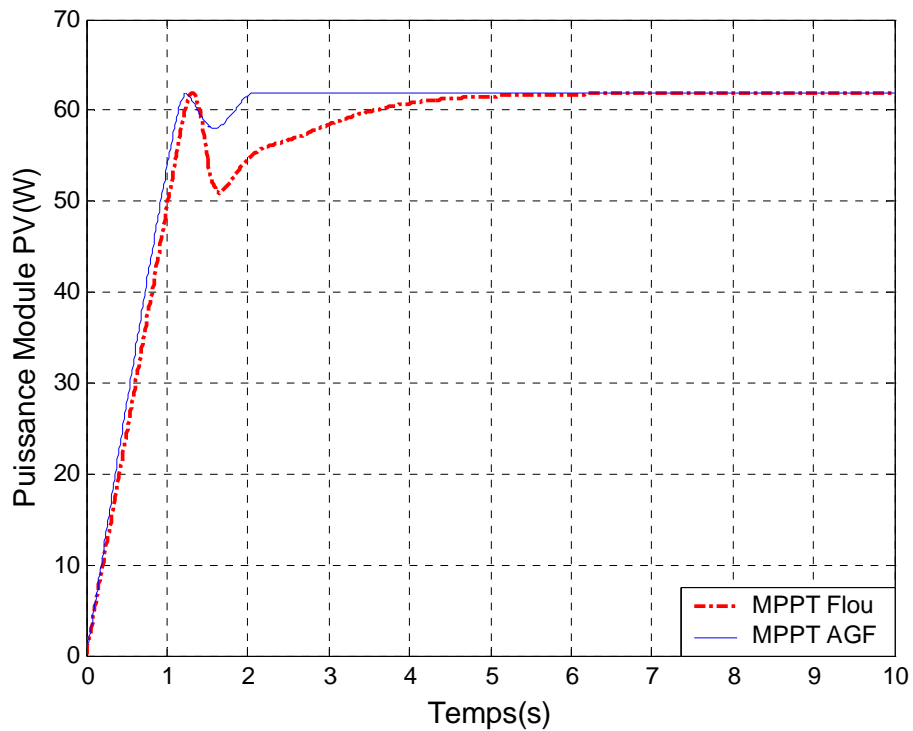


Figure V.40. Variation de la puissance du module PV, des deux contrôleurs MPPT flou et MPPT AGF (obtenu par un codage binaire), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

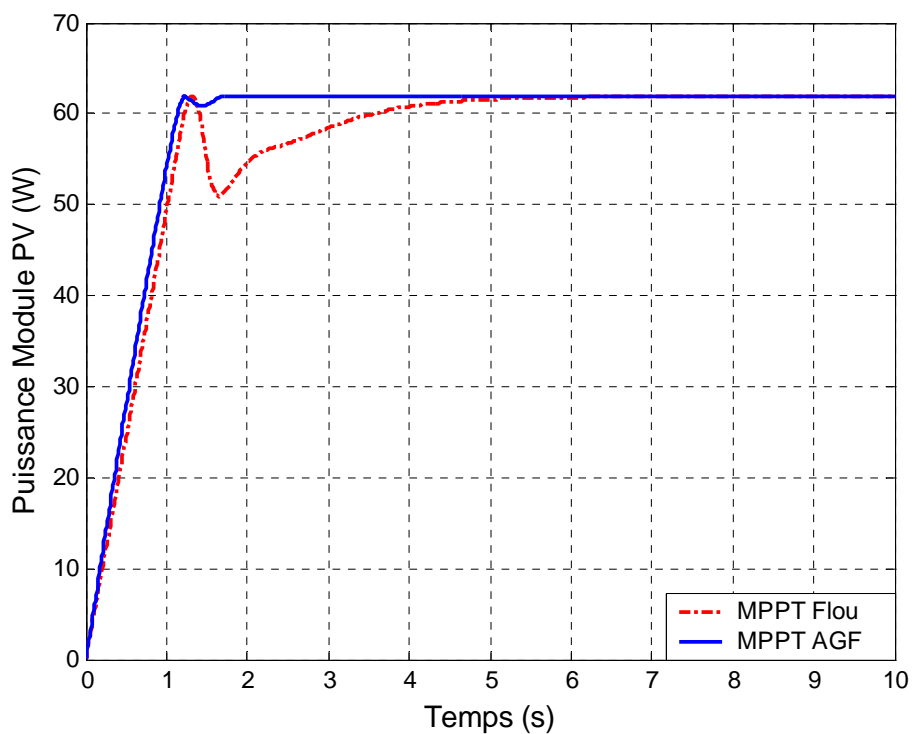


Figure V.41. Variation de la puissance du module PV, des deux contrôleurs MPPT flou et MPPT AGF (obtenu par un codage réel), sous des conditions constantes : la température=25°C et l'ensoleillement=1000W/m².

- ✓ En comparant la figure V.40 avec les résultats de la solution optimale obtenus avec le codage réel présentés sur la figure V.41, utilisant les mêmes paramètres on conclut que c'est le codage réel qui a donné les meilleurs résultats.

V.5.3. Résultats de simulation du système MPPT en utilisant un hacheur buck

Dans le cas du contrôle du système MPPT utilisant un hacheur buck, on a effectué la recherche du contrôleur MPPT optimisé par les AGs avec le même principe que celui du hacheur boost mais avec un nombre de génération égal à 60.

Les valeurs des fonctions d'appartenance du contrôleur obtenu sont illustrées par la figure suivante :

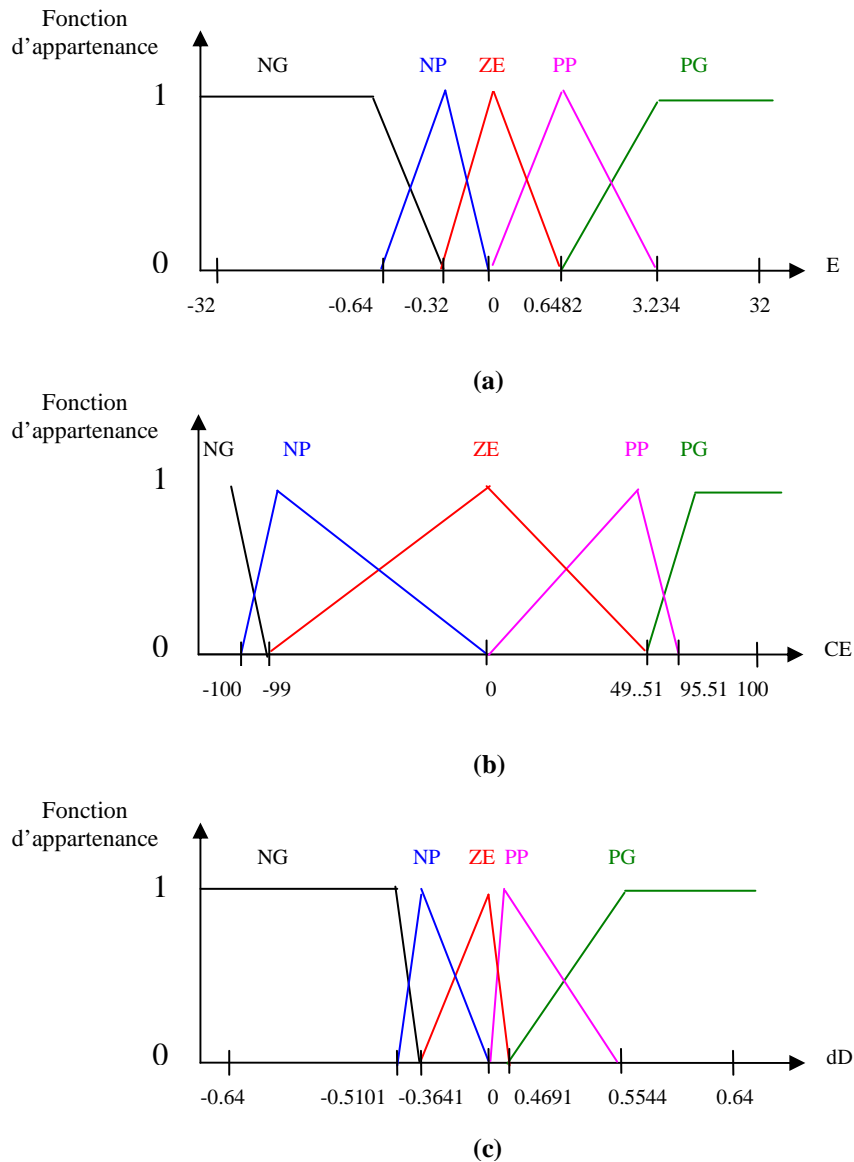


Figure V.42. Fonctions d'appartenance du contrôleur MPPT flou optimal utilisé avec un hacheur buck.

Les variations de la valeur minimale de la fonction objective (fitness) en fonction du nombre de générations dans le cas du contrôleur MPPT optimisé par les algorithmes génétique pour un système avec un convertisseur buck sont données dans la Figure V.43.

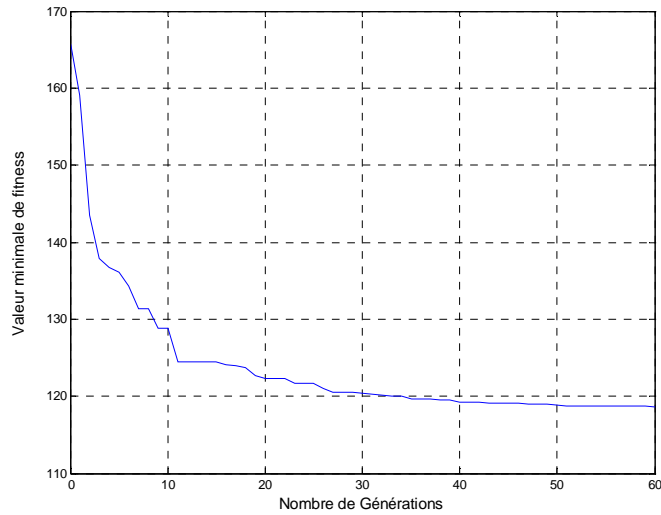


Figure V.43. Valeurs minimales de la fonction objective en fonction du nombre de générations.

Ainsi on donne les résultats de simulation qui sont présentés par les figures (V.44, V.45, V.46, V.47, V.48, V.49 et V.50) :

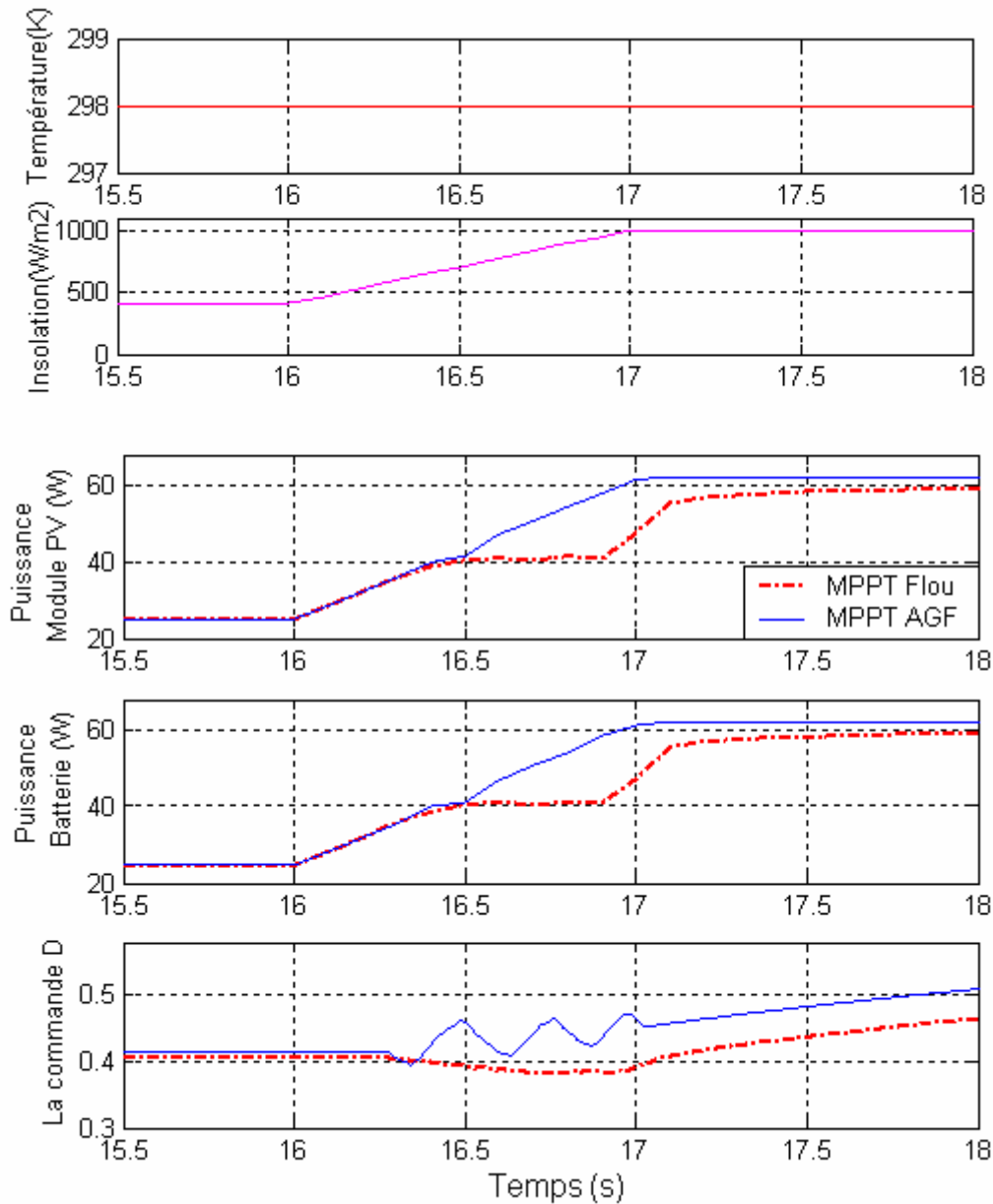


Figure V.44. Réponses des deux contrôleurs MPPT Flou et MPPT AGF pour une augmentation rapide d'ensoleillement de 400W/m² à 1000 W/m² en 1 seconde à une température constante de 25°C. Cas d'un hacheur buck.

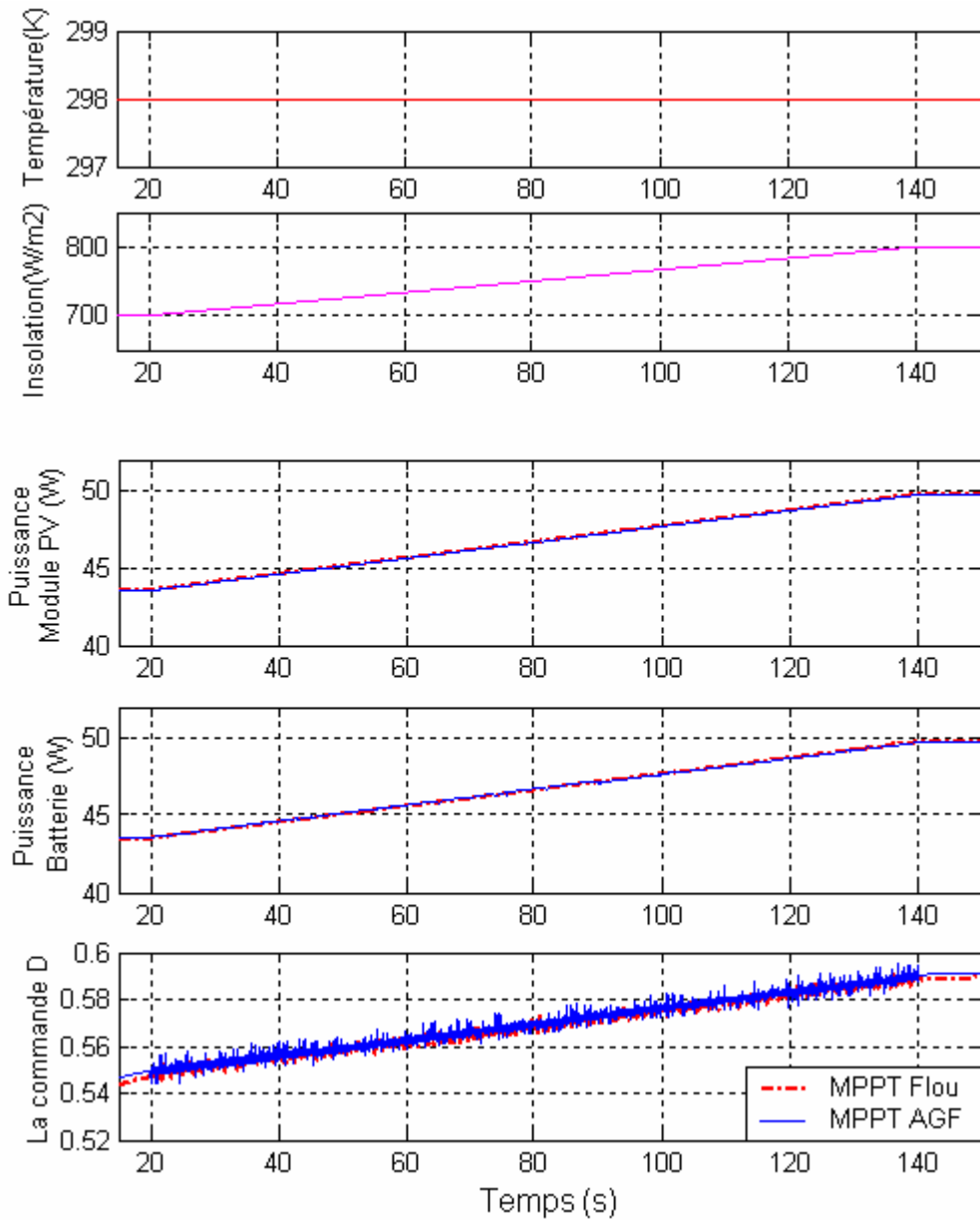


Figure V.45. Réponses des deux contrôleurs MPPT Flou et MPPT AGF pour une augmentation lente de l'ensoleillement de 700W/m² à 800W/m² dans un intervalle de 120s. Cas d'un hacheur buck.

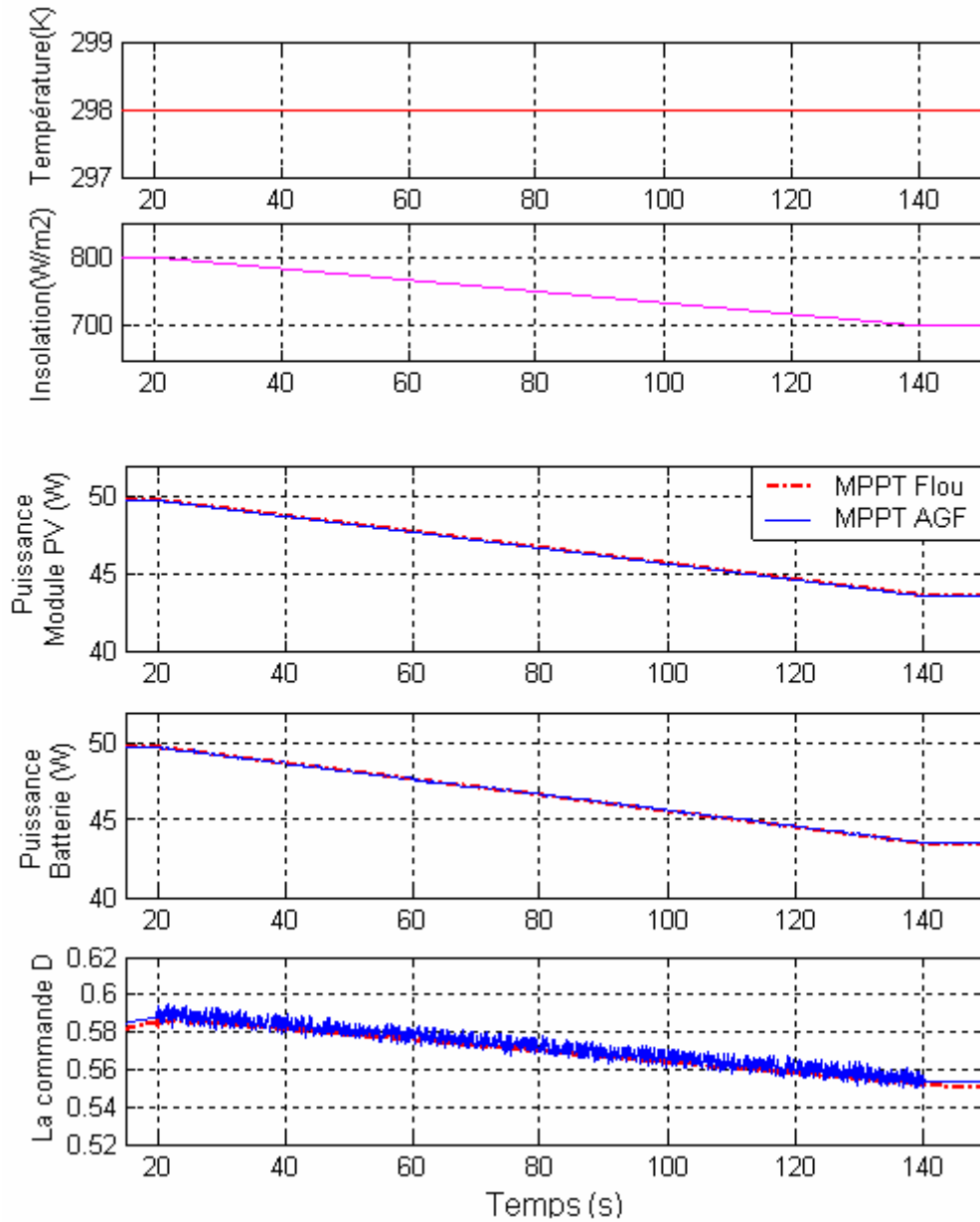


Figure V.46. Réponses des deux contrôleurs MPPT Flou et MPPT AGF pour une diminution lente de l'ensoleillement de 800W/m² à 700W/m² dans un intervalle de 120s (T=25°C). Cas d'un hacheur buck.

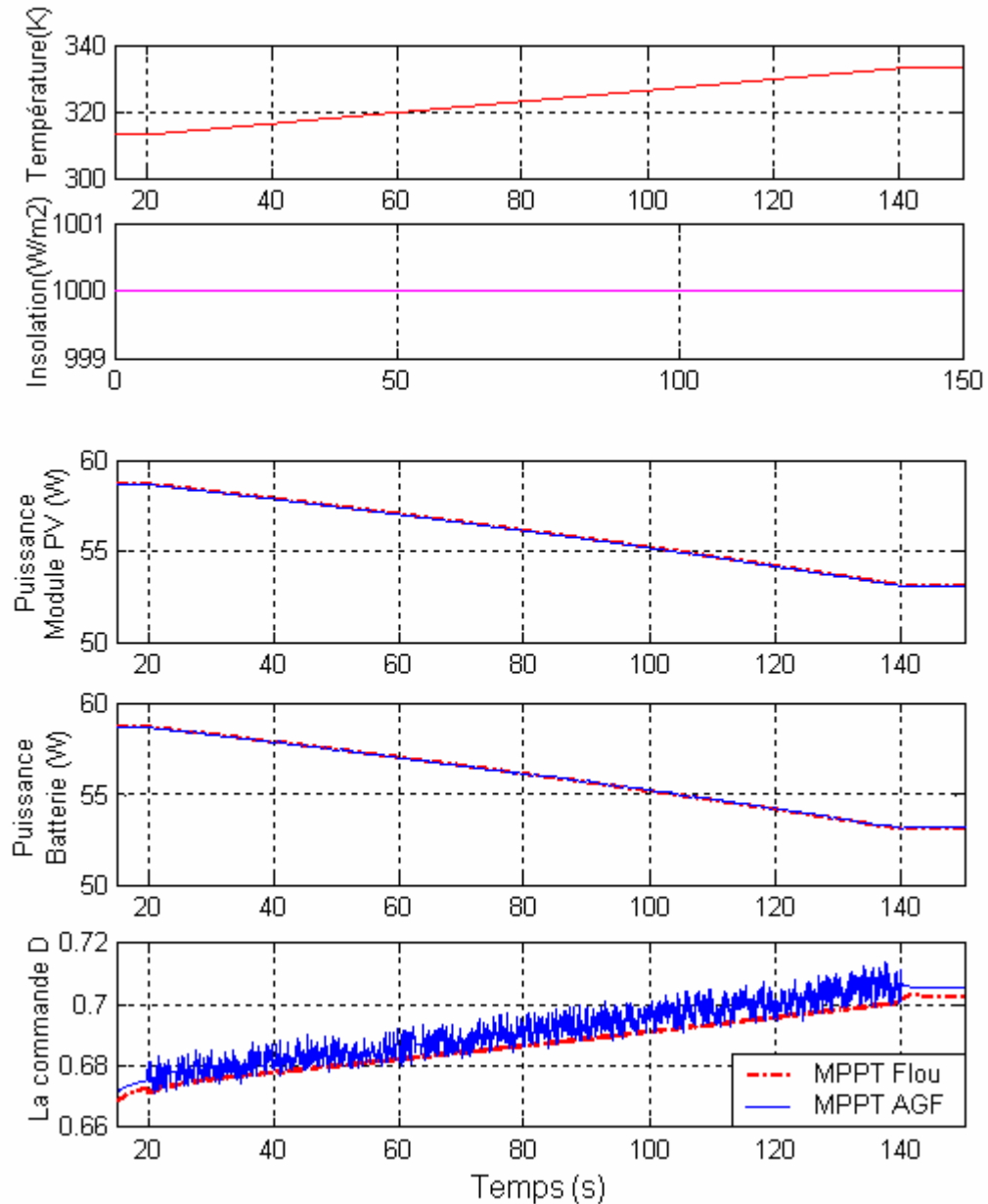


Figure V.47. Réponses des deux contrôleurs MPPT Flou et MPPT AGF pour une augmentation lente de la température de 40°C à 60°C durant un temps de 120s avec un ensoleillement de 1000W/m². Cas d'un hacheur buck.

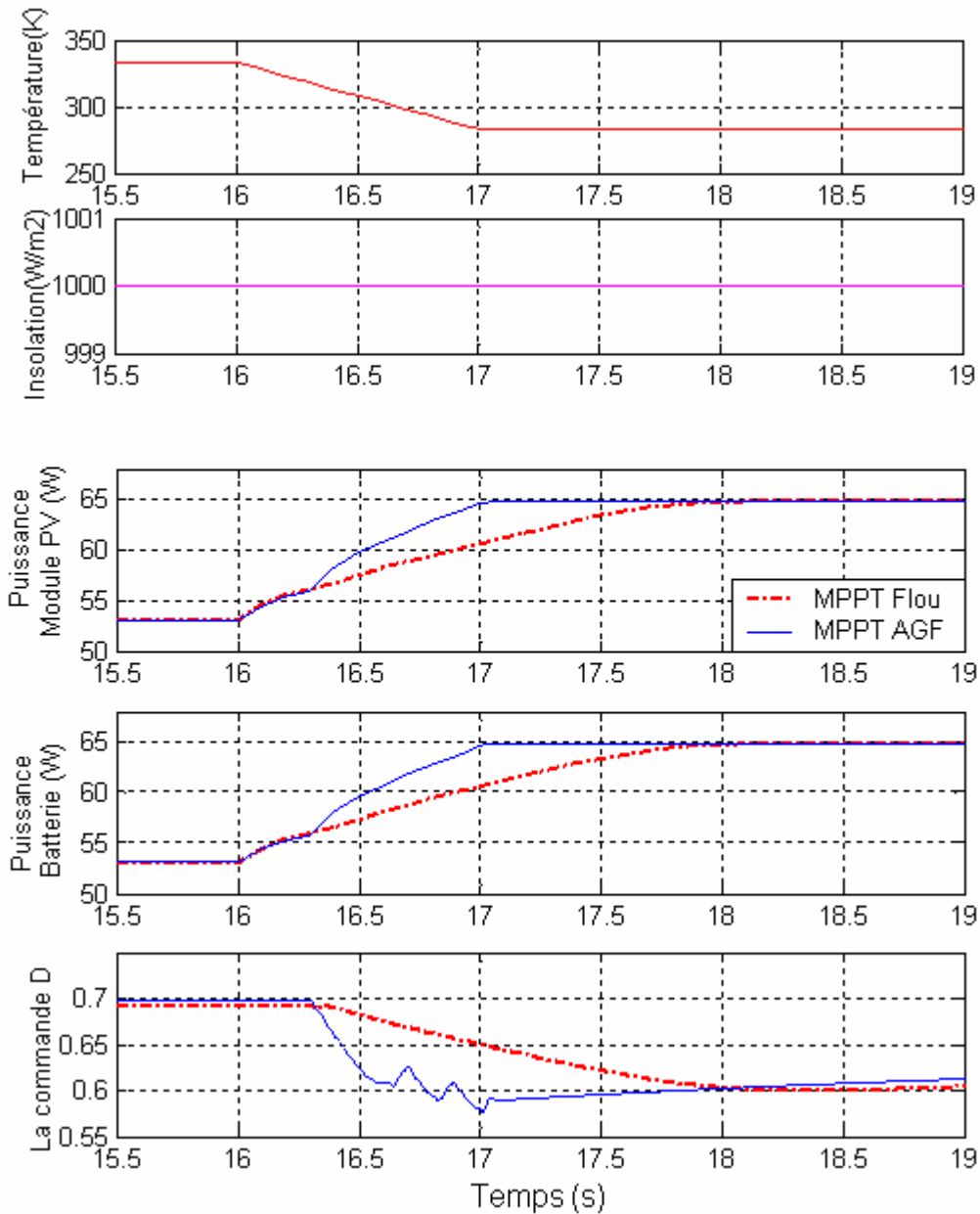


Figure V.48. Réponses des deux contrôleurs MPPT Flou et MPPT AGF pour une diminution rapide de la température de 60°C à 10°C durant un temps de 2s avec un ensoleillement de 1000W/m². Cas d'un hacheur buck.

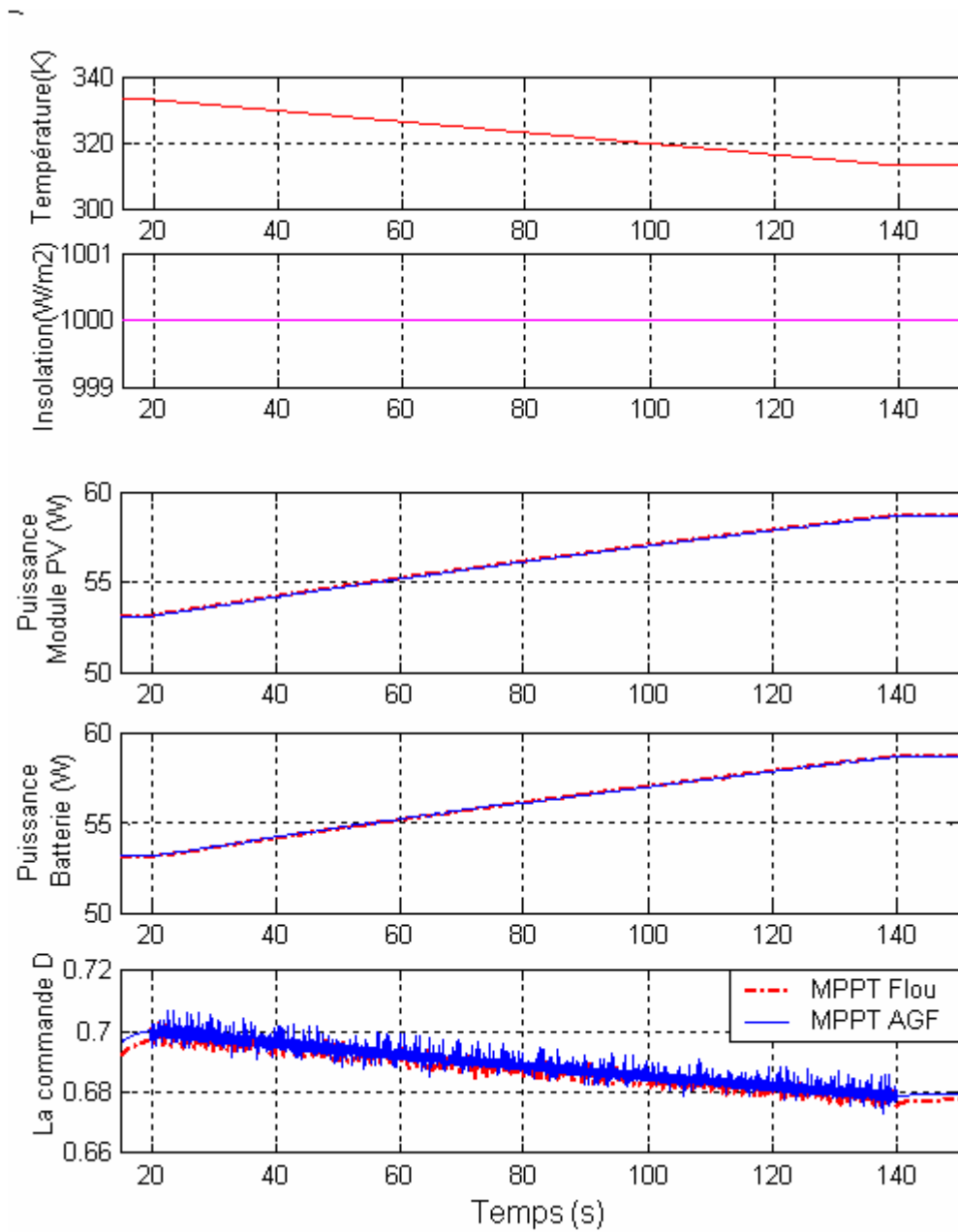


Figure V.49. Réponses des deux contrôleurs MPPT Flou et MPPT AGF pour une diminution lente de la température de 60°C à 40°C durant un temps de 120s avec un ensoleillement de 1000W/m². Cas d'un hacheur buck.

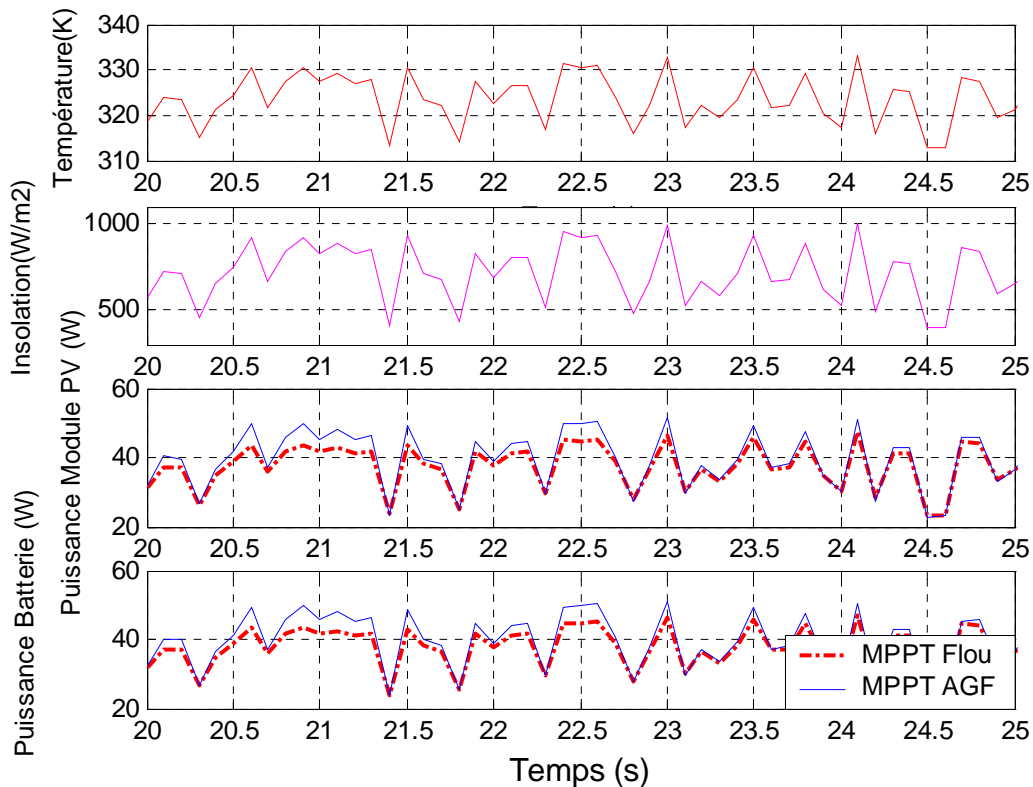


Figure V.50. Réponse des deux contrôleurs MPPT flou et MPPT AGF face aux changements aléatoire de l'ensoleillement et de la température. Cas d'un hacheur buck.

V.6. Conclusion

On a vu dans ce chapitre dans le détail la conception d'un contrôleur MPPT flou. Ensuite nous avons appliqué les algorithmes génétiques pour optimiser de ce dernier pour deux types de convertisseurs le boost et le buck.

Nous avons pu ainsi, comparer les résultats de simulations obtenus par les deux contrôleurs MPPT AGF et MPPT Flou, en les soumettant aux mêmes conditions environnementales dans un but de voir leurs performances et robustesse.

Nous pouvons conclure qu'au travers des résultats obtenus, le MPPT AGF est celui qui est de loin le plus performant et le plus robuste.

CONCLUSION GÉNÉRALE

Dans ce travail, nous avons comme but de réguler la tension du module solaire afin de se positionner au PPM de la caractéristique P-V quelles que soient les conditions de température et d'ensoleillement. Pour cela une commande qui maintient le point de fonctionnement du module précisément au PPM de la caractéristique P-V est nécessaire.

Dans notre travail on a choisi d'utiliser les algorithmes génétiques afin d'aboutir à un CLF optimisé et amélioré pour faire une parfaite poursuite du point de puissance maximale.

Une revue bibliographique a été effectuée concernant l'application des algorithmes génétiques pour l'optimisation des contrôleurs par logique floue CLF. Les résultats de cet effort ont été entièrement satisfaisants.

Les CLF fournissent une commande plus efficace que les contrôleurs classiques pour les systèmes non linéaires. La conception des CLF exige la connaissance de l'expert, mais dans notre travail nous avons montré que nous pouvons avec l'application des AGs construire un contrôleur efficace d'une façon rapide et sûre sans avoir recours à l'expérience et à la connaissance du système. Cette technique peut mener à une plus large utilisation des CLF, alors que le procédé traditionnel rendait leur utilisation inadéquate.

Le CLF optimisé par les AGs est très rapide, il permet effectivement d'aller chercher le point de puissance maximale en un temps assez court. Il permet de faire cela pour plusieurs configurations selon la température ou l'ensoleillement des panneaux solaires et même de réagir à une combinaison de variations intervenant en même temps.

Ceci nous amène à affirmer que l'optimisation par algorithmes génétiques s'avère être une méthodologie d'optimisation très efficace et offre un outil puissant pour le développement des contrôleurs flous.

Perspectives

Il serait très intéressant de réaliser la carte du circuit de commande proposé, afin de pouvoir valider ce contrôleur MPPT par des mesures réelles.

BIBLIOGRAPHIE

- [1] L. Hassaine, *Modélisation et Simulation d'un Système de Conditionnement de Puissance pour la Poursuite de Puissance Maximale dans les Systèmes Photovoltaïques*, Mémoire de Magister, ENP Algérie, Juin 2003.
- [2] K. Sobaihi, *Etude et Réalisation d'un Hacheur MPPT à contre réaction de tension*, Mémoire de Magister, ENP Algérie, Septembre 2003.
- [3] H.J. Möller, *Semiconductors for Solar Cells*, Artech House, Inc, Norwood, MA, 1993.
- [4] A.F. Boehinger. *Self Adaptive DC Converter for Solar Spacecraft Power Supply*. IEEE Transaction on Aerospace and Electronic Systems, AES-4, n°1, pp 102-111, 1968.
- [5] H. Knopf, *Analysis, Simulation, and Evaluation of Maximum Power Point Tracking (MPPT) Methods for a Solar Powered Vehicle*, Master of Science in Electrical and Computer Engineering, Portland State University, 1999.
- [6] R. Gottschalg, M. Rommel, D.G. Ineld, and H. Ryssel, *Comparison of different methods for the parameter determination of the solar cell's double exponential equation*, In 14th European Photovoltaic Science and Engineering Conference (PVSEC), Barcelona, Spain, 1997.
- [7] G.F. Tchoketch Kebir, *Commande des hacheurs MPPT par la logique floue*, Magister, ENP Algérie, février 2006.
- [8] C.F. Lu, C.C. Liu, and C.J. Wu, *Dynamic modeling of battery energy storage system and application to power system stability*, In IEE Proceedings Generation, Transmission and Distribution, Volume 142(4), pages 429-435, IEE, Stevenage, Herts., U.K., July 1995.
- [9] Z.M. Salameh, M.A.Casacca, A.W.Lynch, *A mathematical model for lead-acid batteries*, IEEE Transactions on Energy Conversion, 7(1):93-98, March 1992.
- [10] C. Hua, J. Lin, C. Shen. *Implementation of DSP-controlled photovoltaic system with peak power tracking*.
- [11] A. IDRI, *Un Modèle Intelligent d'Estimation des Coûts de Développement de Logiciels* Doctorat en Informatique, Université du Québec à Montréal, Septembre 2003,
- [12] K.M. Passino, S. Yurkovich, *Fuzzy Control*, Department of Electrical Engineering, Ohio State University.
- [13] H. Buhler, *Réglage Par Logique Floue*, Presses Polytechniques et Universitaires Romandes CH-1015 Lausanne. 1994.
- [14] K. Chafaa, *Etude de la poursuite d'une Cible*, Magister, Université de Batna, 1999.
- [15] C. Chen, C Wong, *Self-generating rule-mapping fuzzy controller design using a genetic algorithm*, IEEE Proc.-Control Theory Appl. No. 2, pp 149, March 2002
- [16] S. Benhamida, *Algorithmes évolutionnaires « prise en compte contraintes et application réelle »*, thèse Doctorat, février 2001.
- [17] T. Murata, *Génétic algorithms for multi-objecve optimization*, thèse Doctorat, Osaka University, Japon, février 1997
- [18] K. Deb, M. Goyal, *A Combined genetic adaptive search (gene AS) for engineering design*, pages 1-20.
- [19] A.Wright, *genetic algorithms for real parameter optimization, in foundations of genetic algorithms*, edited by G.J.E.Rawlins, Morgan Kaufmann, San Mateo, 1991.
- [20] K. Deb, H. George Beyer. *Self adaptive genetic algorithms with simulated Binary crossover*, Technical report N°.CI-61/99, University of Dortmund, Germany, Mars 1999.

- [21] L.JOURDAN, Métaheuristiques pour L'extraction de connaissances : Application à la Génomique ». Thèse Doctorat. Université des Sciences et Technologies de LILLE U.F.R. D'I.E.E.A. Numéro d'Ordre : 3368 . Année : 2003.
- [22] N. LASSOUAOUI, *Segmentation des Images par Différentes Approches et Optimisation avec les Algorithmes Génétiques*. Thèse Doctorat. ENP Alger. Juillet 2004.
- [23] E-G.Talbi."Méthodes d'optimisation avancées ». Cours Polytech'Lille, France.
- [24] Jerzy Korczak, '*Algorithmes Génétiques*' Cours Université Louis Pasteur, Strasbourg, France.
- [25] A. Dipanda, 'Algorithmes génétiques'. Cours algorithmes génétiques - 2004.
- [26] M. Lee, A. Takagi, *Integrating Design Stages of Fuzzy Systems Using Genetic Algorithms*, IEEE, 2nd Conference international of fuzzy systems, San Francisco, pages 612-617, 1993.
- [27] K. Belarbi, F. Titel, *Genetic algorithm for the design of a class of fuzzy controllers: An alternative approach*, IEEE Transactions on fuzzy Systems, Vol 8, pp 398-405, 2000.
- [28] P. Siarry, *Automatique de Base*, 1993, Berti Editions.
- [29] D.E. Goldberg. *Real-coded genetic algorithms, virtual alphabets and blocking*. *Complex Systems*, 5 : pages 139–167, 1991.
- [30] A.H.. Wright. *Genetic algorithms for real parameter optimization*. In *Proceeding of the Foundation Of Genetic Algorithms*. FOGA, 1991.
- [31] Z. Michalewicz. *Genetic algorithms + Data Structures = Evolution Programs*. Springer-Verlag, 1992.
- [32] X. Yin, N. Gernay. *A fast genetic algorithm with sharing scheme using cluster analysis methods in multimodal function optimization*. In *Proceedings of the Artificial Neural Nets and Genetic Algorithms*, 1993.
- [33] P. J. Fleming, R. C. Purshouse. *Genetic Algorithms in Control Systems Engineering*. Research Report No. 789, University of Sheffield, 2001.