



Mise en garde

La bibliothèque du Cégep de l'Abitibi-Témiscamingue et de l'Université du Québec en Abitibi-Témiscamingue (UQAT) a obtenu l'autorisation de l'auteur de ce document afin de diffuser, dans un but non lucratif, une copie de son œuvre dans [Depositum](#), site d'archives numériques, gratuit et accessible à tous. L'auteur conserve néanmoins ses droits de propriété intellectuelle, dont son droit d'auteur, sur cette œuvre.

Warning

The library of the Cégep de l'Abitibi-Témiscamingue and the Université du Québec en Abitibi-Témiscamingue (UQAT) obtained the permission of the author to use a copy of this document for nonprofit purposes in order to put it in the open archives [Depositum](#), which is free and accessible to all. The author retains ownership of the copyright on this document.



**CONCEPTION D'UN BANC D'ESSAI POUR LES CONTROLEURS MPPT
INTELLIGENTS DES SYSTÈMES PHOTOVOLTAÏQUES. APPLICATION AU
CONTRÔLEUR À BASE D'ALGORITHME GÉNÉTIQUE**

MÉMOIRE
PRÉSENTÉ
COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN INGÉNIERIE

PAR
LOUIS BOURNIVAL

Avril 2025

REMERCIEMENTS

Je tiens à exprimer ma profonde gratitude envers l'Université du Québec en Abitibi-Témiscamingue pour m'avoir offert les ressources et l'encadrement nécessaires à la réalisation de ce projet. Mes sincères remerciements vont également à mon directeur de recherche, le professeur Tahar Tafticht, pour son soutien, ses précieux conseils et son engagement tout au long de cette étude. Son expertise et son encouragement ont été essentiels à l'aboutissement de ce travail.

TABLE DES MATIÈRES

REMERCIEMENTS.....	III
TABLE DES MATIÈRES.....	IV
LISTE DES FIGURES.....	VI
LISTE DES SIGLES ET DES ABRÉVIATIONS.....	VIII
LISTE DES SYMBOLES ET DES UNITÉS	IX
RÉSUMÉ	XI
INTRODUCTION	13
1.1 Revue de littérature	14
1.2 Objectifs.....	16
1.3 Méthodologie et structure.....	17
1.3.1 L'approche scientifique	17
1.3.2 Le Matériel	17
1.3.3 La structure du mémoire	18
2. SYSTÈME PHOTOVOLTAÏQUE	19
2.1 Introduction du système	19
2.2 Le principe de fonctionnement.....	19
2.3 La modélisation de la cellule Photovoltaïque	20
2.4 Le Convertisseur DC/DC.....	26
2.4.1 Phase de conduction (ON) :	26
2.4.2 Phase de blocage (OFF) :	26
2.5 Le Contrôleur MPPT	28
3. LES ALGORITHMES MPPT.....	31
3.1 Les algorithmes génétiques	31
3.1.1 Étape 1 : Population initiale de départ	32
3.1.2 Étape 2 : Évaluation de la formule de sélection	33
3.1.3 Étape 3 : Sélection des individus	33
3.1.4 Étape 4 : Croissement des caractéristiques	33
3.1.5 Étape 5 : Mutations	33
3.2 L'algorithme Perturbation et observation	34
3.2.1 Principe de Fonctionnement	34
3.2.2 Étapes de l'Algorithme	35
4. CONCEPTION DU BANC D'ESSAI ET IMPLÉMENTATION	37
4.1 Interface homme/machine avec Matlab	37

4.2	Algorithme génétique	38
4.2.1	Détermination des paramètres du panneau par les algorithmes génétiques	39
4.2.2	Détermination de la tension et courant au maximum de puissance	40
4.2.3	Création de la population	41
4.2.4	Calcul des performances	41
4.2.5	Sélection des meilleurs individus	41
4.2.6	Croisement des membres de la population	42
4.2.7	Mutation de la population	42
4.2.8	Régulation de la tension VMPPT avec un contrôleur PID	42
4.3	La carte de commande	44
4.3.1	Le modèle d'interface ESP32 et Matlab	46
4.3.2	Communication analogique du ESP32	47
4.4	Le montage du panneau solaire et de la source lumineuse	48
4.5	Le montage du convertisseur DC/DC	48
4.5.1	Source lumineuse pour les tests expérimentaux	49
4.5.2	Capteur de courant	49
4.5.3	Capteur de tension	50
4.5.4	Charge du système	51
5.	ESSAIS EXPÉRIMENTAUX ET COMPARAISON DES ALGORITHMES MPPT	53
5.1	Comparaison des algorithmes génétique et P&O dans les conditions normales (éclairage total)	53
5.1.1	Courant et tension mesuré au panneau solaire dans des conditions normales	54
5.1.2	Puissance au panneau solaire dans des conditions d'illumination	54
5.1.3	Courant et tension de la charge dans des conditions normales	55
5.1.4	Puissance de la charge et rapport cyclique dans des conditions normales	56
5.2	Comparaison des algorithmes GA et P&O dans des conditions ombragées (éclairage partiel)	57
5.2.1	Courant et tension au panneau solaire dans des conditions d'ombragées	57
5.2.2	Puissance du panneau solaire dans des conditions ombragées	57
5.2.3	Courant et tension à charge dans des conditions ombragées	58
5.2.4	Puissance à la charge et rapport cyclique dans des conditions ombragées	59
	CONCLUSION	62
	ANNEXE A – PROGRAMME ESP32	65
	BIBLIOGRAPHIE	74

LISTE DES FIGURES

Figure 1 : Démarche schématisée pour réaliser le projet de recherché.	17
Figure 2 : Le diagramme du système photovoltaïque.	19
Figure 3 : Fonctionnement d'un panneau solaire	20
Figure 4 : Schéma d'une jonction d'un panneau solaire	21
Figure 5 : modèle de la cellule photovoltaïque.	21
Figure 6 : modèle de la cellule photovoltaïque.	23
Figure 7 : Courbe I-V et P-V d'une cellule photovoltaïque.	24
Figure 8 : Influence de l'irradiation lumineuse et la température sur le point de puissance maximale dans un panneau solaire	25
Figure 9 : Convertisseur de tension Buck.....	28
Figure 10 : Variation du point de puissance maximale dans le cas de variation de l'éclairement (la température et la charge constants).	29
Figure 11 : Variation du point de puissance maximale dans le cas de variation de (la température (l'éclairement et la charge constants).	29
Figure 12 :Variation du point de puissance maximale dans le cas de variation de la charge (l'éclairement et la température constants).	29
Figure 13 : Organigramme de la méthode MPPT à base d'algorithme génétique.	32
Figure 14 : représentation du système de croisement de gènes.	33
Figure 15 : Étape de l'algorithme P&O	35
Figure 16 : Schéma synoptique du banc d'essai pour les tests des algorithmes MPPT.	37
Figure 17 : Interface HMI "APP designer " de Matlab.	38
Figure 18 : Fonction de performance et paramètres du panneau solaire obtenus par les algorithmes génétiques.	40
Figure 19 : Identification du système avec l'outil de PIDTuner de Matlab.	43
Figure 20 : Diagramme des Entrées/Sorties du esp32.	45
Figure 21 : Représentation du branchement des GPIO du ESP32.	47
Figure 22 : Banc des tests expérimentaux.	48
Figure 23 : Schéma électrique du convertisseur DC/DC et de la carte commande.	49
Figure 24 : Source lumineuse pour les tests expérimentaux (lampe halogène de 500w).	49
Figure 25 : Capteur de courant ACS711	50
Figure 26 : Charge électronique utilisée dans le banc d'essai	51
Figure 27 : Le courant (gauche) et la Tension (droite) mesurés au panneau solaire lors d'essai obtenus avec les algorithmes P&O et génétique.	54

Figure 28 : Comparaison de la puissance d'entrée entre l'algorithme P&O et génétique.	55
Figure 29 : Graphique du courant à la charge (gauche) et de la tension a la charge (droite) pour les algorithmes P&O et génétique.	55
Figure 30 : Puissance de charge (gauche) et rapport cyclique (droite) du système pour chaque l'algorithme P&O et génétique.	56
Figure 31 : Graphique du courant (gauche) et de la Tension (droite) mesuré au panneau solaire lors d'essai fait avec les algorithmes P&O et GA dans des conditions d'ombrages.....	57
Figure 32 : Comparaison de la puissance d'entrée entre l'algorithme P&O et GA pour des conditions d'illumination ombragé.....	58
Figure 33 : Graphique du courant à la charge (gauche) et de la tension a la charge (droite) pour les algorithmes P&O et génétique avec des conditions ombragées.....	58
Figure 34 : Puissance de charge (gauche) et rapport cyclique (droite) du système pour chaque l'algorithme P&O et génétique dans des conditions ombragées.....	59

LISTE DES SIGLES ET DES ABRÉVIATIONS

MPPT : Maximum Power Point Tracking (Poursuite du point de puissance maximale)

PV : Photovoltaïque

ES : Entrées et Sorties

ADC : Analog-to-Digital Converter (Convertisseur Analogique-Numérique)

PID : Proportionnel-Intégral-Dérivé

P&O : Perturb and Observe (Perturbation et Observation)

INC : Incremental Conductance (Conductance Incrémentale)

FOCV : Fractional Open Circuit Voltage (Tension de circuit ouvert fractionnaire)

INC-FL : incremental Conductance with Fuzzy Logic (Conductance Incrémentale avec logique floue)

HMI : Human-Machine Interface (Interface Homme-Machine)

IoT : Internet of Things (Internet des Objets)

PWM : Pulse Width Modulation (Modulation de largeur d'impulsion)

MPP : Maximum Power Poin (Point de Puissance Maximale)

ESP32 : Microcontrôleur populaire pour applications IoT

DC/DC : Convertisseur de courant continu à courant continu

R_s : Résistance série

R_{sh} : Résistance shunt

V_{oc} : Tension en circuit ouvert

I_{sc} : Courant en court-circuit

LISTE DES SYMBOLES ET DES UNITÉS

A	ampère
GHz	gigahertz
kW	kilowatt
VA	Volt Ampère
T	Tension
W	Watt

RÉSUMÉ

À une époque où les applications industrielles progressent à un rythme rapide, le développement durable occupe une place centrale dans tout projet d'ingénierie. Ce concept englobe, entre autres, l'utilisation de sources d'énergie renouvelables, comme l'énergie solaire. Le rendement d'une source d'énergie solaire est mesuré en fonction de la quantité d'énergie produite qui peut être effectivement utilisée pour des opérations pratiques. Ce projet explore l'optimisation de l'extraction de puissance dans un système photovoltaïque à l'aide d'une technique de suivi du point de puissance maximale (Maximum Power Point Tracking) basée sur un algorithme génétique (GA). L'objectif principal est de comparer les performances de cet algorithme avec une méthode classique, l'algorithme de Perturbation et Observation (P&O), tout en concevant et validant un banc de test fonctionnel pour l'expérimentation. Les résultats expérimentaux cherchent à vérifier les performances du GA par rapport à l'algorithme P&O, tout en mettant l'accent sur la mise en place d'un banc d'essai robuste et stable. Ces travaux ouvrent la voie à des recherches futures visant à optimiser davantage les algorithmes MPPT pour une extraction d'énergie encore plus efficace dans des conditions réelles.

Mots-clés : Maximum Power Point Tracking (MPPT), Photovoltaïque, Perturbation et Observation, Algorithmes génétiques, microcontrôleur EPS32, PID.

INTRODUCTION

Présentement, une course aux énergies renouvelables se déroule dans le monde entier afin de trouver une alternative aux énergies fossile. Avec cet engouement vient aussi une course à l'optimisation de procédé de récupération d'énergie des sources d'énergies renouvelables.

Les systèmes photovoltaïques ont démontré qu'ils étaient une option intéressante dans cette course pour trouver une énergie alternative au combustible fossile. Cependant, l'énergie solaire vient avec ses difficultés d'exploitation. En effet, l'énergie recueillie par les cellules photovoltaïques dépend directement de la luminosité et de la température ambiante. Deux paramètres qui varient rapidement au cours d'une journée et qui rend l'extraction optimale d'énergie d'un panneau solaire non linéaire et difficile à obtenir. Afin d'optimiser leurs exploitations, les panneaux solaires sont mis en paire avec une commande qui effectue la recherche et la poursuite du point de puissance maximale (MPPT). Les commandes MPPT peuvent venir avec différents algorithmes, chacune ayant ses avantages et ses inconvénients dans la poursuite et l'extraction d'énergie la plus efficace.

Pour synthétiser ce mémoire, il est possible de simplifier la démarche en quelques points. D'abord, il faut faire une étude de transfert d'énergie entre le panneau solaire et la batterie. Ensuite, concevoir la commande MPPT avec l'algorithme génétique. Puis, valider expérimentalement que la conception du convertisseur MPPT et sa commande.

Notre travail sera présenté dans ce mémoire en 5 chapitres. Dans le premier chapitre, on présente une introduction au monde de l'énergie solaire et une revue de littérature présente dans les différentes revues scientifiques sur le sujet et la présentation des objectifs de ce mémoire.

Le deuxième chapitre présente le système photovoltaïque et détaille chacun de ses composants, soit le panneau photovoltaïque, le convertisseur DC/DC, le contrôleur MPPT.

Le troisième chapitre consiste à détailler le fonctionnement de l'algorithme génétique utilisé pour la recherche du point de puissance maximale et voir comment l'impact des différents paramètres de l'algorithme sur la vitesse de convergence et la robustesse de l'algorithme.

Le quatrième chapitre est consacré à la conception d'un banc d'essai des algorithmes MPPT et de l'implémentation fonctionnelle du système. Les protocoles de communication pour l'échange de données entre les différents sous-systèmes seront aussi présentés dans ce chapitre.

Au cinquième chapitre, une validation expérimentale sera présentée et les résultats obtenus ainsi que les performances du système seront discutés.

Enfin, une conclusion pour le mémoire.

1.1 *Revue de littérature*

Un système photovoltaïque utilise des panneaux solaires pour convertir la lumière du soleil en énergie électrique. C'est une forme de production d'électricité qui devient de plus en plus populaire dans les zones urbaines et rurales. En effet, la production d'énergie à partir de la lumière du soleil est plus accessible et plus fiable que l'obtention de carburant pour un générateur électrique. De plus, un système PV peut fonctionner seul sans nécessiter d'autres sources d'énergie. Cela en fait une excellente option pour les personnes ayant un accès limité à l'électricité ou qui souhaitent réduire leur dépendance aux énergies fossiles. La recherche du maximum de puissance devient importante dans les zones urbaines où l'ombrage des édifices et des arbres peuvent venir affecter le rendement des panneaux solaires. La recherche sur les panneaux solaires a permis de faire des avancées dans le domaine de l'exploitation de l'énergie depuis son apparition. [1].

Les chercheurs se sont principalement concentrés sur le MPPT en choisissant le module et/ou la charge afin d'assurer que le point d'intersection des courbes caractéristiques courant-tension soit proche du point de puissance maximale (MPP). Cependant, cette correspondance n'est pas toujours garantie en raison des variations d'irradiation et de température. Pour résoudre ce problème, certains chercheurs ont essayé de déterminer des paramètres adéquats pour les batteries connectées aux systèmes photovoltaïques afin d'améliorer l'ajustement entre les caractéristiques du module et de la charge [2].

La méthode MPPT classique la plus étudiée dans la littérature et la plus utilisée est basée sur l'algorithme Perturbation et Observation. [3]. Cette méthode perturbe la tension ou le courant d'entrée du système photovoltaïque et observe l'impact sur la puissance de sortie du système pour déterminer la direction dans laquelle se trouve le point de puissance maximale [4].

Les avantages de cette technique sont :

- Simple à mettre en œuvre.
- Faible coût de calcul.
- Fonctionne bien dans des conditions de luminosité stables.

Et les inconvénients :

- Tendance à osciller autour du point de puissance maximale en raison de perturbations continues.
- Peut converger vers un maximum local plutôt que global. [4].

La deuxième méthode MPPT la plus étudiée dans la littérature est basée sur l'algorithme de l'incrémental de la conductance. Cet algorithme utilise les changements de conductance pour déterminer le point de puissance maximale en ajustant la tension ou le courant d'entrée [5]. Les avantages de cette technique sont :

- Réponse plus rapide aux variations de conditions d'éclairage par rapport à P&O.
- Moins susceptible d'osciller autour du point de puissance maximale.

Et les inconvénients :

- Plus complexe à mettre en œuvre que P&O.
- Exige une surveillance continue de la conductance. [5].

Un autre algorithme qui est basée sur la mesure de la tension en circuit ouvert et utilise une linéarisation entre la tension en circuit ouvert et la tension de fonctionnement optimale du système pour déterminer le point de fonctionnement optimal du système photovoltaïque [6].

Les avantages de cette technique sont :

- Utilise une mesure unique de tension à circuit ouvert, réduisant ainsi la complexité du matériel.
- Convient aux systèmes avec des panneaux photovoltaïques à faible coût.

Et les inconvénients :

- Peut-être moins précis dans des conditions de variation rapide de l'ensoleillement.
- Sensible aux variations de température. [6].

Chaque algorithme MPPT a ses avantages et ses inconvénients, et le choix de l'algorithme dépend souvent des caractéristiques spécifiques du système photovoltaïque et des exigences de performance. D'autres méthodes MPPT intelligentes sont étudiées dans la littérature pour y remédier aux inconvénients des méthodes classiques précédentes, Ces méthodes intelligentes sont souvent basées sur les réseaux de neurones, la logique floue, les algorithmes génétiques, etc. Ces méthodes cherchent à optimiser les performances du système photovoltaïque dans diverses conditions [7- 8]. Les avantages des méthodes intelligentes sont :

- La recherche efficace des paramètres optimaux dans un espace de recherche complexe et non linéaire.
- L'adaptabilité à différents types de systèmes et conditions environnementales.

Et les inconvénients des méthodes intelligentes sont:

- Coût de calcul plus élevé par rapport aux autres méthodes.
- Nécessite une connaissance approfondie du système pour choisir et définir les paramètres à optimiser.
- Complexité accrue d'implémentation. [9].

1.2 Objectifs

L'objectif principal de notre projet est de proposer une méthode d'optimisation d'un système photovoltaïque embarqué avec Matlab qui va effectuer le calcul pour la recherche du maximum de puissance du système à base de l'algorithme génétique. De plus, concevoir un banc d'essai pour valider expérimentalement la commande MPPT proposée et comparer les résultats obtenus avec ceux obtenus via un algorithme classique. Les algorithmes utilisés seront implémentés à l'aide d'une carte de commande prototype basée sur un modèle de microcontrôleur ESP32.

Pour atteindre cet objectif, il faut réaliser sous-objectifs suivants:

- L'étude et analyse des systèmes photovoltaïques.
- L'étude des modèles de commande de MPPT les plus répandus actuellement.
- Analyse du transfert d'énergie dans une source photovoltaïque.
- Modéliser le fonctionnement du MPPT sous MATLAB.
- Concevoir la commande MPPT de telle sorte qu'elle soit paramétrable en temps réel.
- Réaliser le banc d'essai pour valider expérimentalement la commande MPPT.
- Implémentation de la commande de contrôle MPPT proposée sur la carte de commande ESP32.
- Validation expérimentale de la méthode MPPT proposée.
- Comparer les performances de la méthode proposée et les performances de la méthode classique.

1.3 Méthodologie et structure

1.3.1 L'approche scientifique

La méthodologie utilisée sera l'approche expérimentale, en effet, la démonstration de l'hypothèse sera faite par la réalisation d'un système fonctionnel de commande MPPT sur une carte de commande ESP32 et en comparant les résultats avec un algorithme de type observation et perturbation. L'approche expérimentale utilisera la division du problème en sous-système indépendant pour les tests initiaux. Lorsque chaque sous-système fonctionnera indépendamment, l'intégration de chacun de ce sous-système sera effectuée.

Le diagramme suivant montre la démarche qui sera emprunté pour atteindre les objectifs de notre projet de recherche.

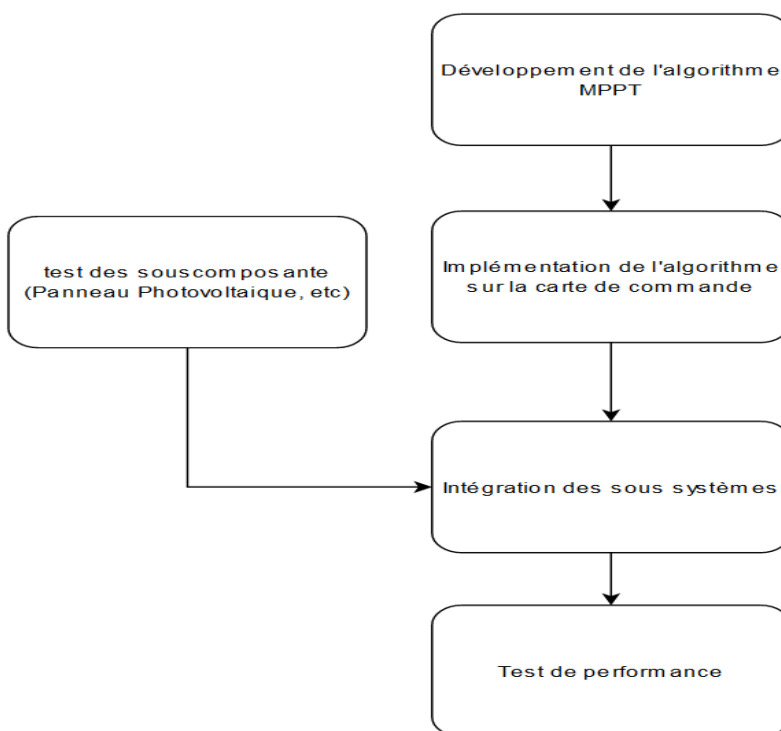


Figure 1 : Démarche schématisée pour réaliser le projet de recherche.

1.3.2 Le Matériel

Un système photovoltaïque sera conçu pour mener à bien notre projet de recherche. Un panneau solaire sera utilisé pour fournir l'énergie photovoltaïque et un convertisseur DC/DC ainsi

que sa carte de commande seront conçus. Les capteurs de tension et de courant seront utilisés pour mesurer la puissance du système via la carte de prototypage ESP32. Le logiciel MATLAB sera utilisé pour obtenir un système paramétrable en temps réel.

1.3.3 La structure du mémoire

La présentation du mémoire s'effectuera par chapitre. Dans le chapitre I, on présente la théorie autour des systèmes photovoltaïques et des recherches de maximum de puissance. Plus précisément, la présentation de la physique d'un panneau photovoltaïque, les convertisseurs de puissance seront présentés dans le chapitre II. Pour le chapitre III, une présentation plus détaillée de l'algorithme génétique et des différentes méthodes d'application dans le contexte du projet. Ensuite, le chapitre IV portera sur la conception et les paramètres du système. Enfin, une présentation des résultats ainsi que la conclusion du projet de recherche seront présentées dans le chapitre V.

2. SYSTÈME PHOTOVOLTAÏQUE

2.1 Introduction du système

Le système photovoltaïque peut être divisé en 3 majeurs parties : le panneau solaire, le convertisseur DC/DC et le contrôleur MPPT. Cette structure permet plusieurs choix d'association de composante qui fait varier grandement l'efficacité du système. Le panneau solaire sert de source d'énergie et le convertisseur DC/DC permet de varier la tension d'opération du panneau solaire. Le contrôleur MPPT effectuer les calculs pour donner le point d'opération du convertisseur DC/DC afin maximiser l'énergie du panneau. Le diagramme de la figure.1 montre la structure générale du système photovoltaïque [11].

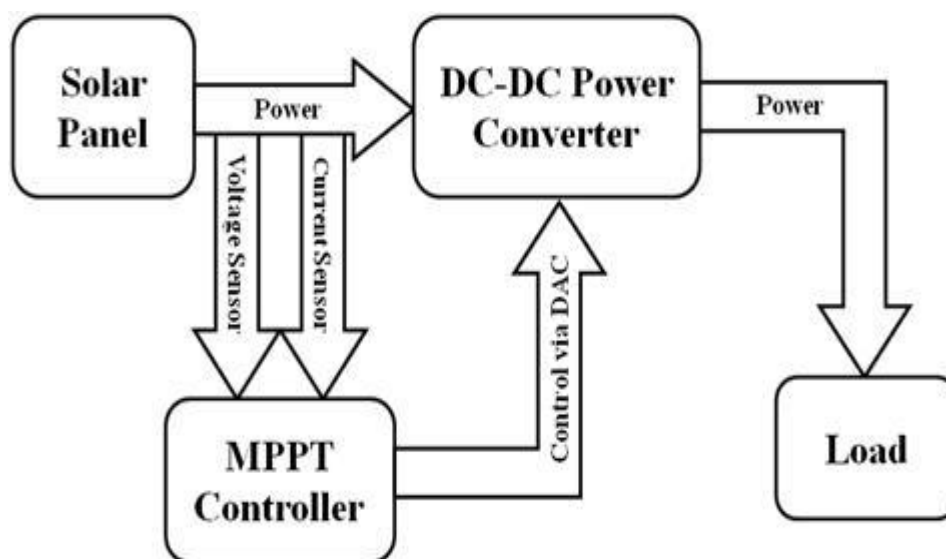


Figure 2 : Le diagramme du système photovoltaïque [12]

2.2 Le principe de fonctionnement

Les cellules photovoltaïques sont mises en série et en parallèle selon un arrangement déterminé et forment un module qui va permettre de transformer l'énergie du soleil en électricité (figure.2). Les cellules sont produites à partir de silicium, un semi-conducteur qui a un comportement assez spécifique lorsque soumis au rayonnement solaire. En effet, les photons qui viennent frapper les cellules vont transmettre leurs énergies aux matériaux de silicium et il va produire un courant électrique [13].

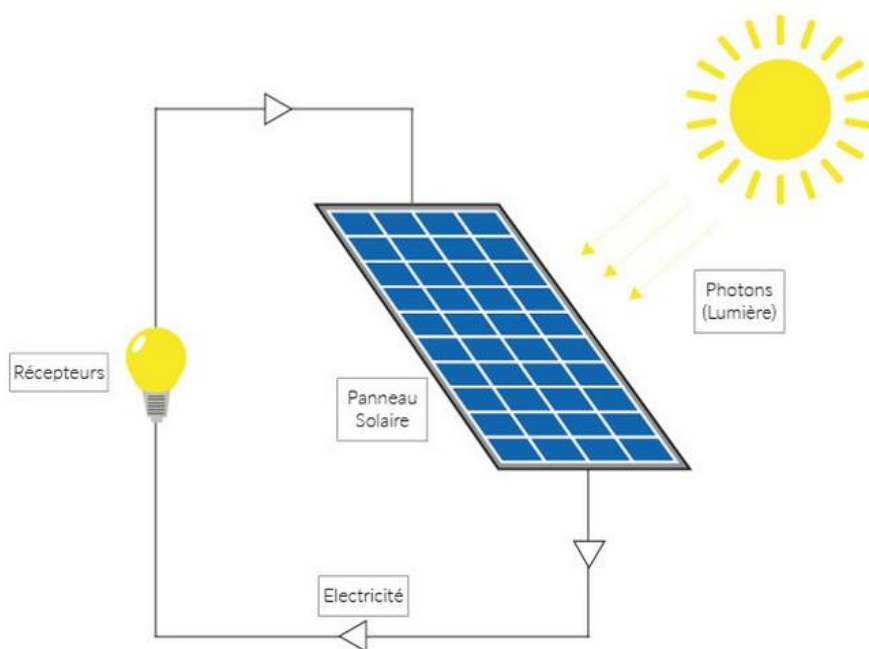


Figure 3 : Fonctionnement d'un panneau solaire¹

1

Il y a plusieurs facteurs qui vont influencer sur la quantité d'énergie que le panneau solaire va produire. Les facteurs tels que l'ensoleillement, la température, le type de contrôleurs utilisés affectent grandement la production du panneau solaire [14]. Dans la mesure du possible, le contrôleur est le seul facteur commandable qui nous permet d'avoir un impact directement sur la production du panneau solaire. Il est possible de maximiser la puissance de sortie du panneau solaire de plusieurs façons. La manière la plus utilisée est d'optimiser la puissance de sortie avec un convertisseur DC/DC. Le rapport cyclique va permettre de modifier l'impédance de sortie vue par le panneau solaire. Lorsque l'ensoleillement et la température changent, le point de puissance maximale change aussi. Le contrôleur doit détecter ce changement et poursuivre en continu la recherche du maximum de puissance, ce contrôleur est appelé MPPT [1].

La modélisation de la cellule Photovoltaïque

Le circuit équivalent d'une cellule solaire est composé d'un générateur de courant idéal en parallèle avec une diode polarisée en inverse, le tout connecté à une charge. Le courant généré est directement proportionnel à l'intensité lumineuse. Cela montre l'importance de reproduire

¹ <https://www.pose-renovation.fr/comment-fonctionne-un-panneau-solaire-photovoltaïque/>

fidèlement le spectre solaire lors des tests des cellules solaires, et explique pourquoi un simulateur solaire est un équipement indispensable dans ce contexte.

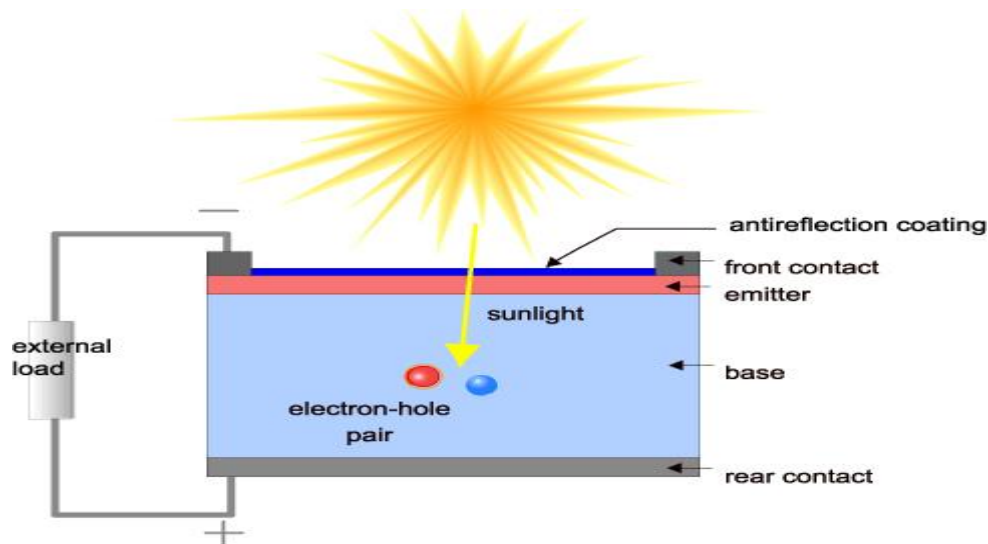


Figure 4 : Schéma d'une jonction d'un panneau solaire [15]

Bien que la quantité de courant produite varie en fonction de l'intensité lumineuse, d'autres limitations inhérentes aux cellules solaires restreignent leur efficacité. Ces limitations sont représentées par les autres composants du circuit équivalent.

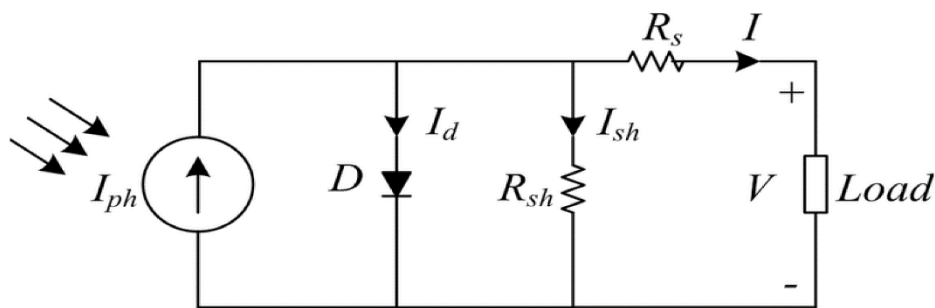


Figure 5 : modèle de la cellule photovoltaïque. [16]

Si la résistance de la charge dépasse celle de la diode, la diode absorbera une partie du courant, augmentant la différence de potentiel entre les bornes, mais réduisant le courant dirigé à travers la charge. À l'inverse, si la résistance de la diode est supérieure à celle de la charge, les électrons circuleront facilement à travers la charge, ce qui entraînera un courant plus élevé. Cependant, dans ce cas, la différence de potentiel entre les bornes sera relativement faible. Cela

illustre une contrainte fondamentale des cellules solaires : optimiser le courant implique souvent de compromettre la tension. Il existe un point d'équilibre, appelé point de puissance maximale, où la tension et le courant sont optimisés pour maximiser la puissance produite [16].

À partir du schéma de circuit équivalent de la cellule solaire (Figure.6), il est possible d'extraire des équations pour décrire et modéliser les panneaux photovoltaïques. Ces équations permettent également de définir certains des indicateurs clés utilisés pour caractériser les cellules solaires. Dans sa forme la plus simple, le courant traversant la charge (I) peut être décrit comme la différence entre le courant généré (I_{Gen}) et la somme du courant qui circule à travers la diode (I_{DIODE}) et du courant perdu à travers la résistance de shunt (I_{SH}).

Le courant de la jonction d'une cellule solaire :

$$I = I_{Gen} - I_{DIODE} - I_{SH} \quad (1)$$

Le courant de la diode (I_{DIODE}) est donné par l'équation idéale de la diode :

$$I_{Diode} = I_0 \left\{ \exp \left[\frac{qV}{nkT} \right] - 1 \right\} \quad (2)$$

où :

I_0 est le courant de saturation en polarisation inverse,

n est le facteur d'idéalité de la diode,

q est la charge de l'électron ($1,6 \times 10^{-19}$ C),

k est la constante de Boltzmann ($1,38 \times 10^{-23}$ J/K),

T est la température absolue (en Kelvin),

V est la tension aux bornes de la diode.

En intégrant ces paramètres, l'équation complète décrivant le courant dans une cellule solaire devient :

$$I = I_{GEN} - I_0 \left\{ \exp \left[\frac{qV}{nkT} \right] - 1 \right\} - I_{Sh} \quad (3)$$

Cette équation est fondamentale car elle explique la forme caractéristique de la courbe courant-tension (IV) des cellules solaires. Elle permet de modéliser et d'optimiser les performances des cellules en fonction de différents paramètres comme la température, l'intensité lumineuse ou les propriétés du matériau. Voici la courbe caractéristique d'une diode :

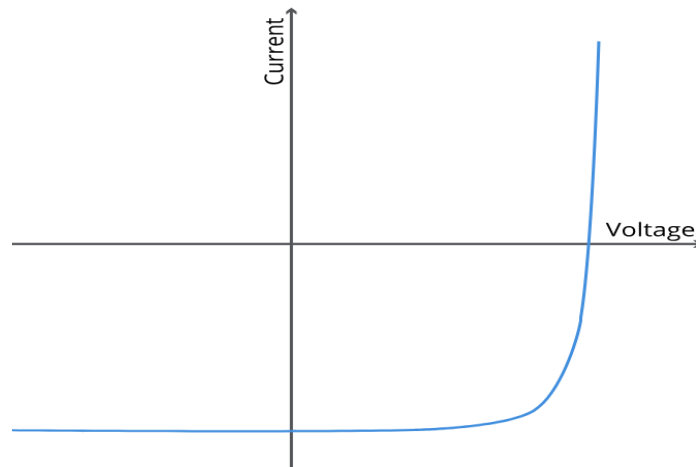


Figure 6 : modèle de la cellule photovoltaïque. ²

L'équation du courant de saturation de la diode est :

$$I_0 = I_0^{ref} \left(\frac{T}{T_{ref}} \right)^3 e^{\frac{E_g}{k} \left(\frac{1}{T} - \frac{1}{T_{ref}} \right)} \quad (4)$$

I_0^{ref} : Courant de saturation à une température de référence Tref.

E_g : Énergie de bande interdite du matériau (en eV).

Enfin, la relation entre le courant I et la tension V peut être modélisée par l'équation suivante en tenant compte de toutes les pertes et effets non idéaux [12]: $I_{gen} = (I_{sc} + K_i(T - 298)) \frac{G}{1000}$

$$I = I_{gen} - I_0 \left\{ e^{\frac{q(V+IR_s)}{nKT}} - 1 \right\} - \frac{V + IR_s}{R_{sh}} \quad (5)$$

R_s : Résistance série, ²qui modélise les pertes internes du panneau.

R_{sh} : Résistance shunt, qui représente les pertes dues à des chemins de fuite qui permettent au courant de circuler en dehors de la cellule photovoltaïque.

Cette équation décrit le courant à n'importe quelle tension en tenant compte des facteurs de perte liés à la résistance série et shunt, ainsi que des effets thermiques.

² <https://www.ossila.com/pages/iv-curves-measurement>

La courbe IV (courant-tension) d'une cellule solaire représente la relation entre le courant et la tension. Cette courbe est l'une des caractéristiques clés des cellules solaires et constitue également un indicateur important pour évaluer leur performance.

Dans la courbe IV d'une cellule solaire, le courant et la tension sont inversement proportionnels. Lorsque l'intensité lumineuse augmente, le courant de la cellule solaire augmente également, mais la tension diminue. Cela s'explique par le fait que la résistance interne de la cellule solaire diminue avec l'augmentation de l'intensité lumineuse, ce qui entraîne une augmentation du courant, mais une diminution de la tension [17].

La courbe IV d'une cellule solaire peut également être utilisée pour évaluer son efficacité.

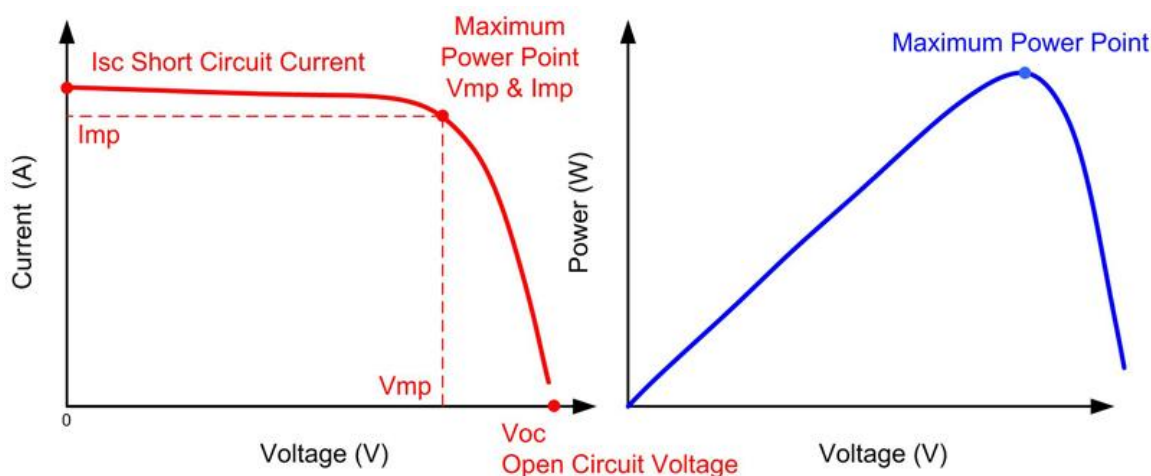


Figure 7 : Courbe I-V et P-V d'une cellule photovoltaïque.³

Le principe d'un MPPT est de trouver si le panneau opère à sa puissance optimale dans les conditions présentes et applique des changements au rapport cyclique du convertisseur DC/DC pour se rapprocher de ce point optimal. Il effectue ce processus en continu puisque les conditions environnementales sont en continuelles changements. Il est possible d'utiliser plusieurs variantes et techniques de recherches. Certaines de ces techniques requièrent différents signaux d'entrées. Certaines sont compliquées à implémenter et d'autre plus simples. Dans le cas ci-dessous, une approche avec les algorithmes génétiques sera explorée afin de

³ <https://www.solairepro.com/compareur-panneau-solaire.fr/tension-en-serie-et-parallele-comprendre-limpact-sur-vos-panneaux-solaires/>

démontrer leur utilité dans la recherche d'un maximum de puissance d'un panneau photovoltaïque.

Le panneau solaire présente trois points importants (figure 7) : le courant de court-circuit (I_{sc}), la tension à vide (V_{oc}) et la puissance optimale (P_{op}) délivrée par le panneau solaire à une charge optimale (R_{op}). Dans ce cas, le panneau solaire fonctionne à leur MPP. Les figures donnent les caractéristiques puissance-tension (P-V) d'un panneau solaire pour différentes valeurs de rayonnement solaire et de température. Le courant de court-circuit est fortement proportionnel au rayonnement solaire : l'augmentation du rayonnement entraîne une augmentation de courant et également de puissance de sortie maximale. D'autre part, la dépendance à la température est inverse : une augmentation de la température entraîne une réduction de la tension à vide (lorsqu'elle est suffisamment élevée) et donc également de la puissance de sortie maximale. Par conséquent, ces effets opposés des variations du rayonnement solaire et de la température sur la puissance de sortie maximale font qu'il est important de suivre efficacement le point de puissance maximale.

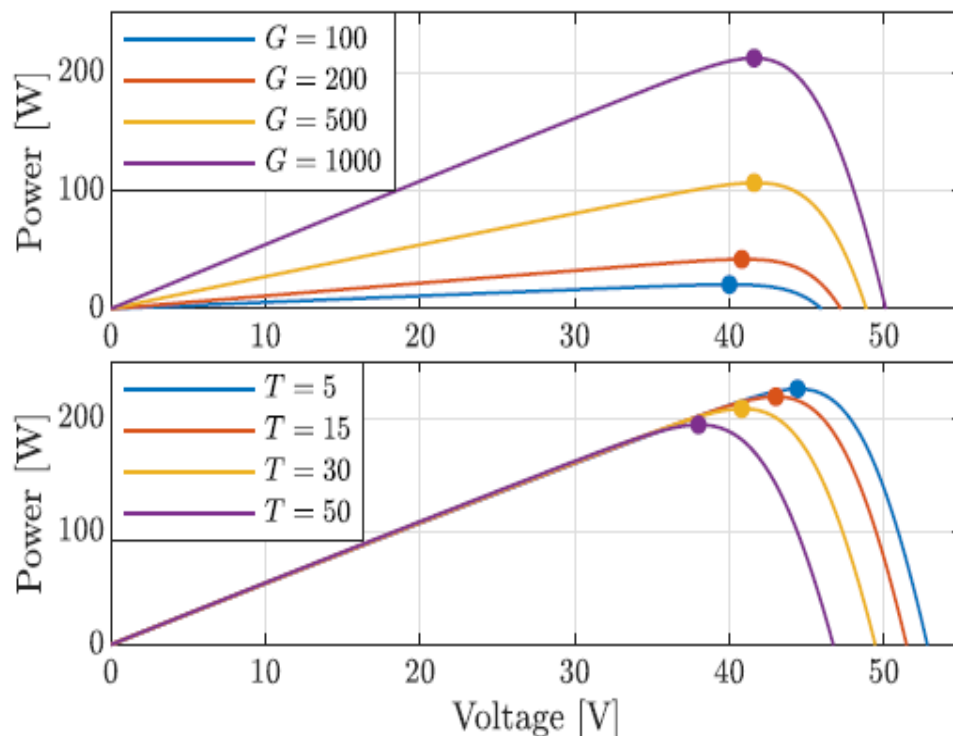


Figure 8 : Influence de l'irradiation lumineuse et la température sur le point de puissance maximale dans un panneau solaire [17]

2.3 Le Convertisseur DC/DC

Dans un système photovoltaïque, le convertisseur DC/DC abaisseur synchrone est utilisé pour ajuster la tension du panneau solaire à la tension requise par la charge ou la batterie. Le contrôleur MPPT ajuste dynamiquement le rapport cyclique D pour maximiser la puissance extraite du panneau solaire, en fonction des conditions d'ensoleillement et de température. [18]

Le convertisseur DC/DC abaisseur synchrone est particulièrement adapté aux systèmes photovoltaïques en raison de sa haute efficacité et de sa capacité à fonctionner avec des tensions d'entrée variables, ce qui est courant dans les applications solaires où la tension du panneau peut varier en fonction des conditions environnementales.

Le convertisseur DC/DC abaisseur synchrone, également appelé convertisseur Buck, est utilisé pour réduire la tension d'entrée à une tension de sortie inférieure. Dans un convertisseur Buck synchrone, un transistor et une diode sont remplacés par deux transistors, ce qui permet de réduire les pertes de conduction et d'améliorer l'efficacité globale du système.

Le fonctionnement du convertisseur DC/DC abaisseur synchrone peut être divisé en deux phases principales [19]:

2.3.1 Phase de conduction (ON) :

Le transistor $S1$ est activé (fermé), permettant au courant de circuler de la source d'entrée (panneau solaire) à travers l'inductance L et vers la charge. Pendant cette phase, l'énergie est stockée dans l'inductance L sous forme magnétique. Le transistor $S2$ est désactivé (ouvert) pendant cette phase.

2.3.2 Phase de blocage (OFF) :

Le transistor $S1$ est désactivé (ouvert), interrompant le courant provenant de la source d'entrée. Le transistor $S2$ est activé (fermé), permettant au courant de circuler à travers l'inductance L et la charge, en maintenant la continuité du courant. L'énergie stockée dans l'inductance L est transférée à la charge pendant cette phase.

Les équations de base du convertisseur sont les suivantes :

Pour le rapport cyclique :

$$D = \frac{T_{ON}}{T_{ON} - T_{Off}} \quad (6)$$

T_{ON} : Temps de conduction du transistor S1.

T_{Off} : Temps de blocage du transistor S1.

Dans le cas de la conduction discontinue, le courant de l'inductance s'annule et il faut considérer que l'inductance est déchargée au début du cycle. Cela ne change pas l'équation de tension et reste-la même. Pour dimensionner le reste des composantes, les équations suivantes sont utilisées. [10]

Pour trouver l'inductance :

$$L = \frac{V_{in} * D}{2 * \Delta I_L * F_{sw}} \quad (7)$$

ΔI_L = Courant ondulateur accepté

V_{in} = Tension d'entrée du système

L = inductance en Henry

D = rapport cyclique

F_{sw} = Fréquence de commutation du transistor

Pour trouver la capacité du condensateur :

$$C = \frac{I_{out} * D}{\Delta V_{RPL} * F_{sw}} \quad (8)$$

ΔV_{RPL} = Tension ondulateur accepté

I_{out} = courant de sortie du système

C = Capacitance en Farad

D = rapport cyclique

F_{sw} = Fréquence de commutation du transistor

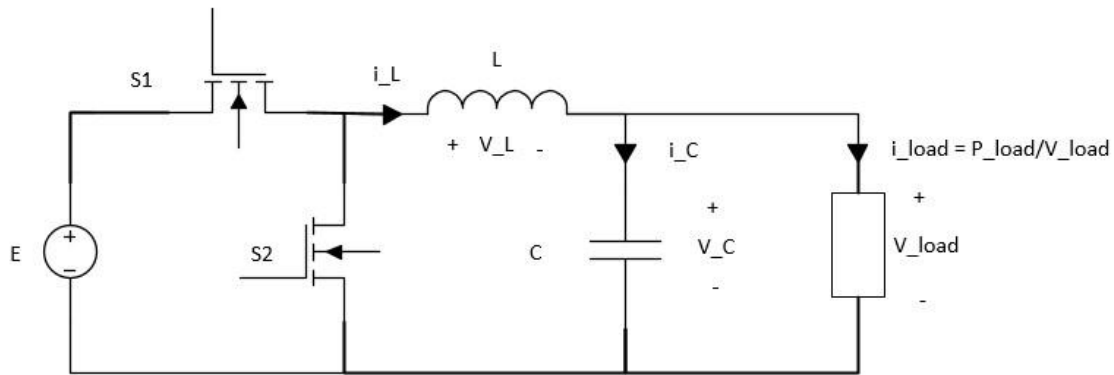


Figure 9 : Convertisseur de tension Buck [20]

2.4 Le Contrôleur MPPT

Les contrôleurs MPPT jouent un rôle essentiel dans l'optimisation du rendement des systèmes photovoltaïques. Leur fonction principale est de réguler la tension et le courant de sortie des panneaux solaires afin de maximiser la puissance extraite en fonction des conditions environnementales. Cela est nécessaire car l'ensoleillement et la température varient constamment, influençant la performance des cellules photovoltaïques (figures 10, 11 et 12).

Le principe du MPPT repose sur l'identification et la poursuite du point de puissance maximale de la courbe courant-tension (P-V) du panneau solaire. Ce point représente la combinaison optimale de tension et de courant permettant d'extraire le maximum d'énergie. Différentes techniques peuvent être employées pour assurer cette optimisation, chacune ayant ses propres avantages et inconvénients en termes de précision, de rapidité de convergence et de complexité de mise en œuvre.

Parmi les approches MPPT classiques utilisées, on trouve l'algorithme de Perturbation et Observation et l'algorithme de l'incrément de la conductance, qui ajustent dynamiquement le point de fonctionnement en fonction des variations de puissance. Cependant, ces méthodes peuvent souffrir de limitations telles que des oscillations autour du MPP ou une convergence lente en présence de changements rapides des conditions d'ensoleillement.

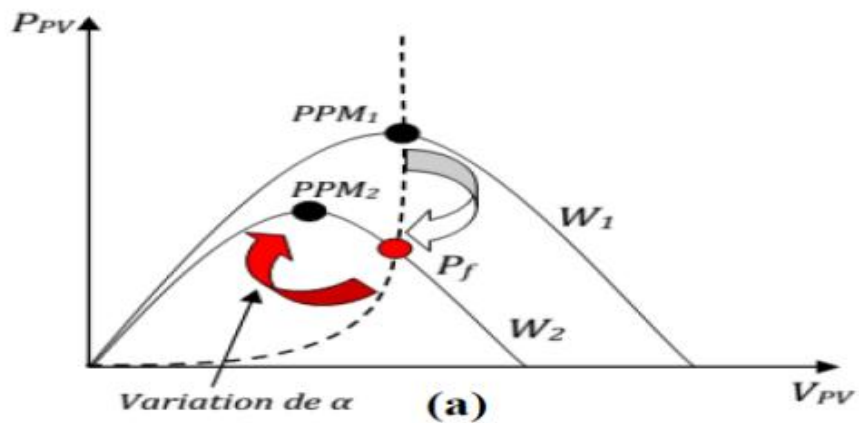


Figure 10 : Variation du point de puissance maximale dans le cas de variation de l'éclairement (la température et la charge constants). [21]

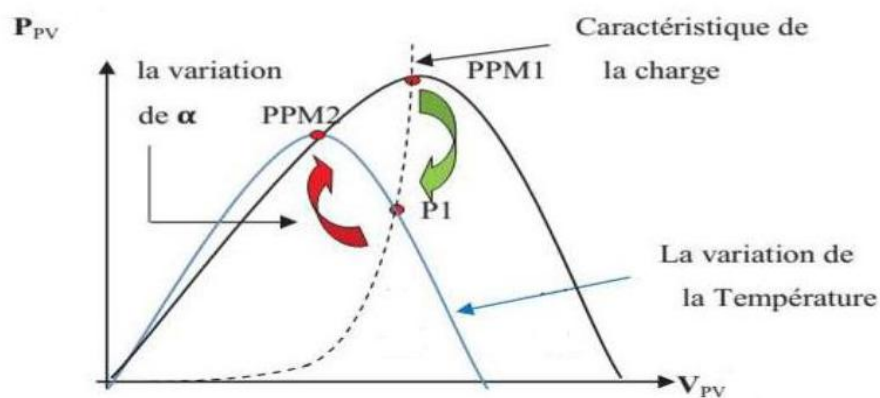


Figure 11 : Variation du point de puissance maximale dans le cas de variation de (la température (l'éclairement et la charge constants). [21]

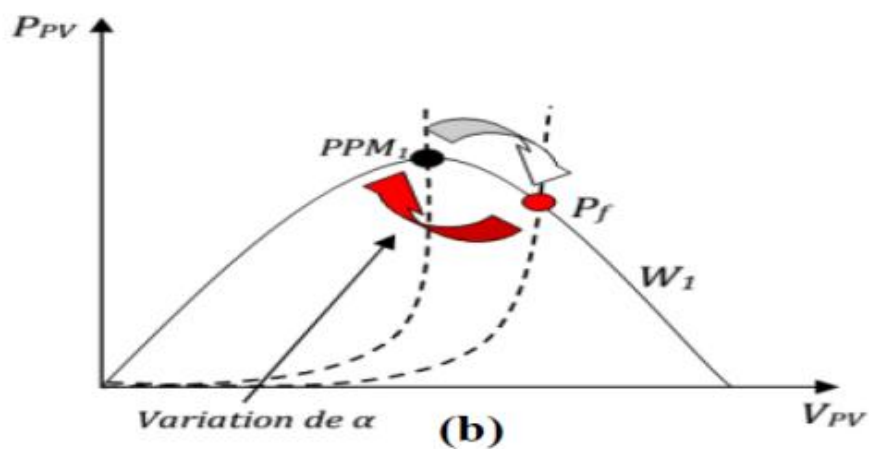


Figure 12 : Variation du point de puissance maximale dans le cas de variation de la charge (l'éclairement et la température constants). [21]

Dans le cadre de cette étude, une approche basée sur les algorithmes génétiques a été retenue pour optimiser la recherche du point de puissance maximale et améliorer les performances du contrôle MPPT. Cette technique s'inspire du processus d'évolution naturelle et permet de déterminer les paramètres optimaux du MPPT en fonction des conditions de fonctionnement. Contrairement aux méthodes conventionnelles, l'utilisation des algorithmes génétiques permet une exploration plus efficace de l'espace de recherche, réduisant ainsi le risque de convergence vers un maximum local et améliorant la robustesse du système.

L'implémentation du contrôleur MPPT est assurée par un microcontrôleur ESP32, qui exécute les calculs nécessaires à l'application de l'algorithme génétique. Ce dernier utilise une population initiale de solutions potentielles, évalue leur aptitude à optimiser la puissance, et applique des opérations de sélection, de croisement et de mutation pour améliorer progressivement les performances du système.

En intégrant cette approche innovante, le contrôleur MPPT développé dans ce projet vise à améliorer l'efficacité énergétique des systèmes photovoltaïques tout en garantissant une adaptation rapide aux fluctuations environnementales. L'étude, l'implémentation et l'évaluation expérimentale de cette méthode sera présentée dans les chapitres suivants afin de comparer ses performances avec celles des techniques classiques.

3. LES ALGORITHMES MPPT

3.1 Les algorithmes génétiques

Les algorithmes génétiques sont directement inspirés du modèle Darwinien de l'évolution [22,23]. Dans le cas de la nature, l'évolution a sélectionné les individus les mieux adaptés à leurs environnements pour la reproduction des descendants de la prochaine génération. Le processus de sélection naturelle permet aux individus avec des caractéristiques avantageuses de passer ces mêmes caractéristiques à leurs enfants. De ce fait, les enfants auront à leurs tours un avantage sur la compétition et devrait même être mieux équipé que leurs parents puisqu'ils devraient hériter des meilleures caractéristiques de leurs deux parents. Le simple fait de toujours prendre le meilleur des deux parents finirait sur plusieurs générations par avoir des individus tous identiques qui descendraient tous de la même ligne. Pourtant ce n'est pas ce qui est observé dans la nature. En effet, des mutations aléatoires sont introduites dans la population ce qui amène des nouveautés dans les caractéristiques disponibles [24]. De plus, la transmission des gènes ne se fait pas de manières optimales, certains gènes qui n'apportent rien aux capacités des individus peuvent se transmettre de manières aléatoires. Tout cela contribue à créer une population diversifiée qui s'adapte à différents écosystèmes sur plusieurs générations.

Ce système de sélections, mutations et reproductions par itération peut être reproduit afin de trouver le maximum de puissance dans les cellules photovoltaïque. En effet, on décompose souvent les algorithmes génétiques en cinq étapes pour l'implémentation dans l'optimisation de problème [25]. Les cinq étapes sont :

1. Production de la population initiale
2. Évaluation de la formule de sélection
3. Sélection des individus
4. Croisement des caractéristiques
5. Mutations

Les algorithmes génétiques sont itératifs, lorsque l'étape 5 est atteint, les paramètres obtenus sont évalués et retourne à l'étape 2 afin d'améliorer la précision (figure.13). Une boucle est donc enclenchée jusqu'à ce qu'il soit déterminé que le résultat est satisfaisant. Avant de parler de la population initiale de l'algorithme génétique, il faut aborder la fonction à optimiser. Si le but est d'optimiser la puissance produite par un ou plusieurs panneaux photovoltaïques, il faut utiliser la fonction définie au chapitre 2, équation 3 pour déterminer la puissance de sortie [14].

La fonction à optimiser est la puissance du système; $P=V \cdot I$

On utilise donc le courant, multiplié par la tension pour trouver :

$$P = \left(V_T * \ln \left[\left(1 - \frac{I}{I_{sc}} \right) \left(e^{\frac{V_{oc}}{V_t}} - 1 \right) + 1 \right] + R_s * I \right) * I \quad (9)$$

La fonction d'évaluation est donc obtenue. L'algorithme génétique va chercher le courant qui va maximiser la puissance du système. Il faut noter que la fonction a besoin de la tension en circuit ouvert et le courant de court-circuit pour l'optimisation. Il faudra donc prendre ces mesures directement sur les panneaux photovoltaïques lors du fonctionnement du système [26].

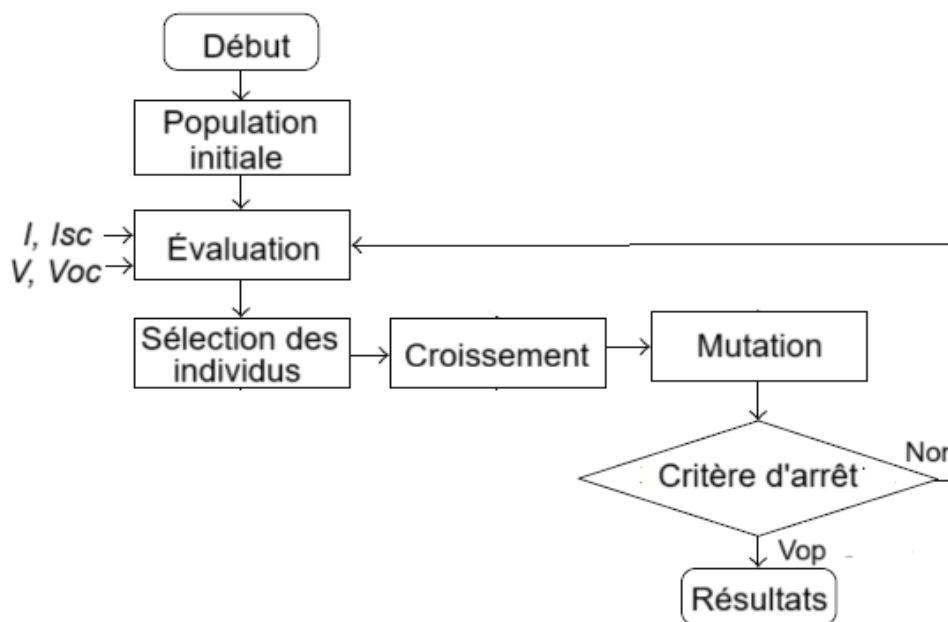


Figure 13 : Organigramme de la méthode MPPT à base d'algorithme génétique.

3.1.1 Étape 1 : Population initiale de départ

Maintenant que la fonction à optimiser est connue, il est possible de produire la population initiale. La population représente l'ensemble des jeux de paramètres à optimiser. Au départ, ils sont créés aléatoirement, certains paramètres peuvent avoir des intervalles déterminés afin de représenter la réalité et une quantité prédéterminée est créée selon les besoins de précision et la puissance de calcul qu'on a à notre disposition. Les jeux de paramètres sont appelés chromosomes. La population créée pour le MPPT va donc être une matrice de X individus par S . S représente le nombre de bits du système qui permet de paramétrer la précision de l'algorithme.

3.1.2 Étape 2 : Évaluation de la formule de sélection

Pour l'étape 2, les chromosomes sont créés et un score est attribué à chaque chromosome pour les comparer entre eux. La fonction d'évaluation est obtenue à partir du modèle du panneau photovoltaïque. Le score obtenu donne la probabilité d'un chromosome de se reproduire. Il existe plusieurs méthodes pour comparer les chromosomes entre eux tels que l'évaluation directe et la roulette.

3.1.3 Étape 3 : Sélection des individus

On prend les individus avec le meilleur score obtenus à l'étape 2 et on rejette les autres, encore une fois, le nombre d'individus retenus peut dépendre de plusieurs facteurs. Les individus sont mis en pair afin d'aller à la prochaine étape. Les paires sont sélectionnées selon l'implémentation voulue par l'algorithme, encore une fois, une sélection aléatoire est souvent utilisée.

3.1.4 Étape 4 : Croisement des caractéristiques

Le croisement est la section la plus importante du processus de l'algorithme génétique. Un point de croisement est choisi.

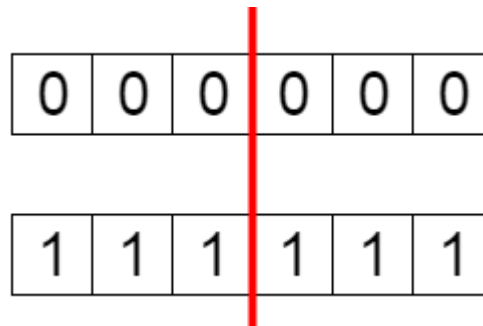


Figure 14 : représentation du système de croisement de gènes. [27]

Les gènes sont maintenant exprimés sous forme binaire. À partir du point de croisement, les gènes de la paire de chromosome sont ensuite échangés. Les chromosomes résultants du croisement sont appelés les descendants et on considère que les descendants sont une perturbation des parents. Ces descendants vont aller dans la nouvelle population pour la prochaine boucle de l'algorithme.

3.1.5 Étape 5 : Mutations

La dernière étape du processus de l'algorithme génétique est la mutation. La mutation est le procédé par lequel on modifie un bit aléatoirement dans les paramètres du descendant. Par

exemple, si le descendant d'un chromosome est [1, 0, 0, 1], après l'application de la mutation, il devient [1, 1, 0, 1]. La question est comment la mutation est appliquée. Un paramètre de mutation est construit pour ce faire. Le paramètre indique le pourcentage de l'ensemble de gènes qui est alloué à être modifié. Un autre paramètre qui est introduit par la mutation est l'amplitude de mutation. Ce paramètre détermine où dans le nombre binaire, la mutation peut avoir lieu. Plus la mutation a lieu proche du bit de poids fort MSB, plus le nombre entier qu'il représente ne va changer.

Après la mutation, les chromosomes sont reconvertis en nombre entier et envoyés à l'étape 2 pour évaluation. Si l'étape sur l'évaluation détermine que les paramètres sont suffisamment optimisés, la boucle s'arrête et un résultat est obtenu [27].

3.2 *L'algorithme Perturbation et observation*

Le suivi du point de puissance maximale est une technique essentielle dans les systèmes photovoltaïques pour maximiser l'efficacité de la conversion de l'énergie solaire en électricité. Parmi les diverses méthodes de MPPT, l'algorithme de perturbation et observation (P&O) est largement utilisé en raison de sa simplicité et de son efficacité et il est la référence pour comparer les algorithmes MPPT.

3.2.1 Principe de Fonctionnement

L'algorithme P&O repose sur l'observation de la puissance de sortie du panneau solaire puis la perturbation périodique du point de fonctionnement et après l'observation l'effet de la perturbation sur la puissance de sortie pour trouver le point de puissance maximale. Les principaux inconvénients de cet algorithme P&O sont un long temps de convergence, une grande oscillation autour du point de puissance maximale, et un problème de dérive associé aux changements rapides de l'éclairement solaire. [28].

3.2.2 Étapes de l'Algorithme

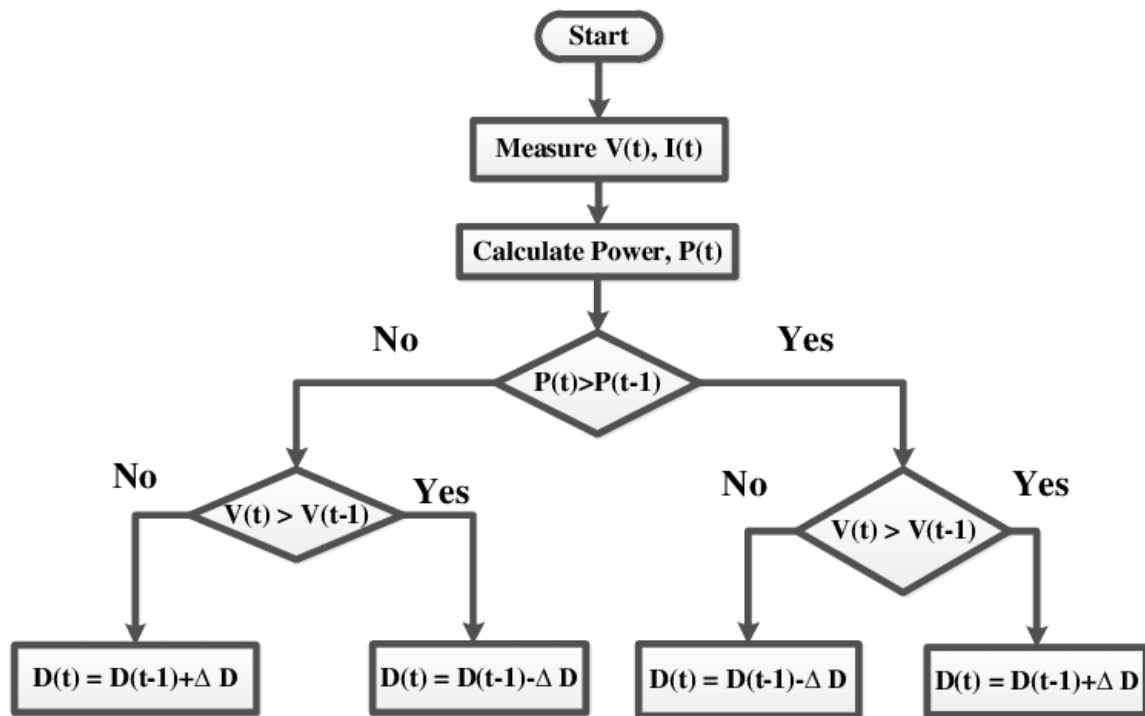


Figure 15 : Étape de l'algorithme P&O [29]

4. CONCEPTION DU BANC D'ESSAI ET IMPLÉMENTATION

La conception du banc d'essai est divisée en trois parties distinctes pour faciliter la démarche de réalisation et d'implémentation. La première partie consiste à implémenter l'interface homme machine (HMI) dans Matlab et puis à la deuxième partie, on développe et implémente les algorithmes MPPT (algorithme P&O et algorithme génétique). Ensuite, la troisième partie est consacré à la conception du convertisseur DC/DC. La dernière section de ce chapitre sera dédiée à la gestion et la communication via le microcontrôleur entre les différentes parties du projet. La figure.16 donne le schéma synoptique représentant le système du banc d'essai mise au point pour les tests expérimentaux des différents algorithmes MPPT.

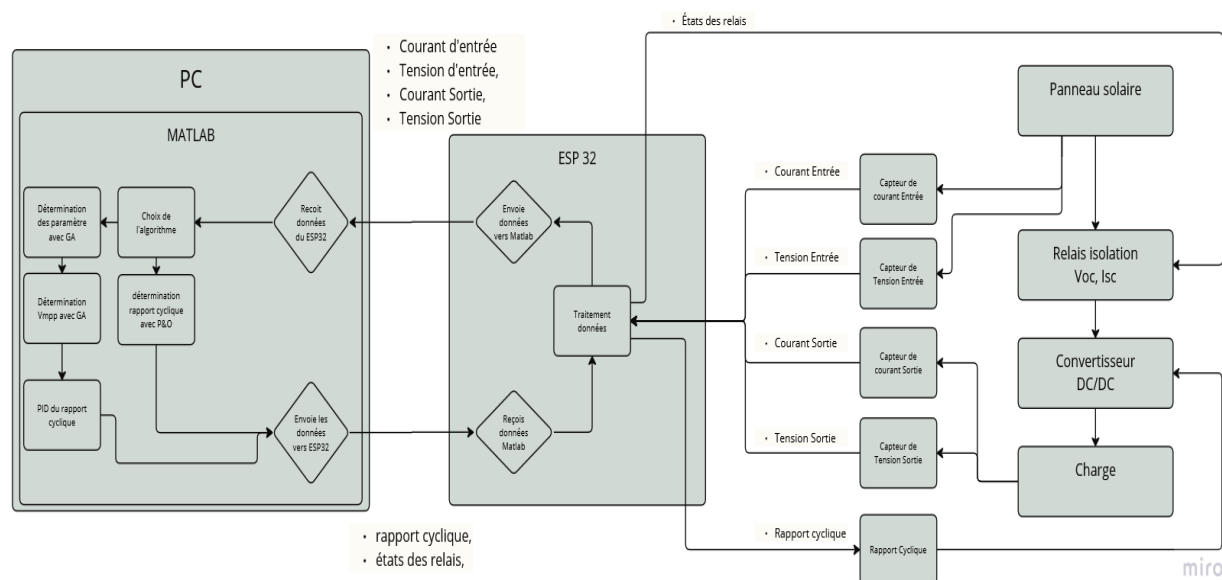


Figure 16 : Schéma synoptique du banc d'essai pour les tests des algorithmes MPPT.

4.1 Interface homme/machine avec MATLAB

MATLAB, abréviation de Matrix Laboratory, est un logiciel de calcul numérique et un environnement de programmation développé par MathWorks. MATLAB repose sur un langage de haut niveau conçu pour les calculs numériques, avec une structure axée sur les matrices, ce qui le rend particulièrement puissant pour les opérations linéaires. Il propose une interface conviviale composée d'un éditeur de scripts, d'une fenêtre de commande et d'outils de visualisation intégrés.

Il excelle dans le traitement et l'analyse de données, permettant de manipuler des jeux de données, de créer des graphiques en deux et trois dimensions, et de générer des visualisations

interactives. Simulink, un outil complémentaire de MATLAB, permet de modéliser et de simuler des systèmes dynamiques ce qui le rend encore plus flexible.

L'outil « APP designer » de Matlab a été utilisé pour développer l'interface HMI de contrôle pour les algorithmes MPPT (figure17). Cette interface permet de simplifier la programmation et d'intervenir rapidement sur les paramètres du système.

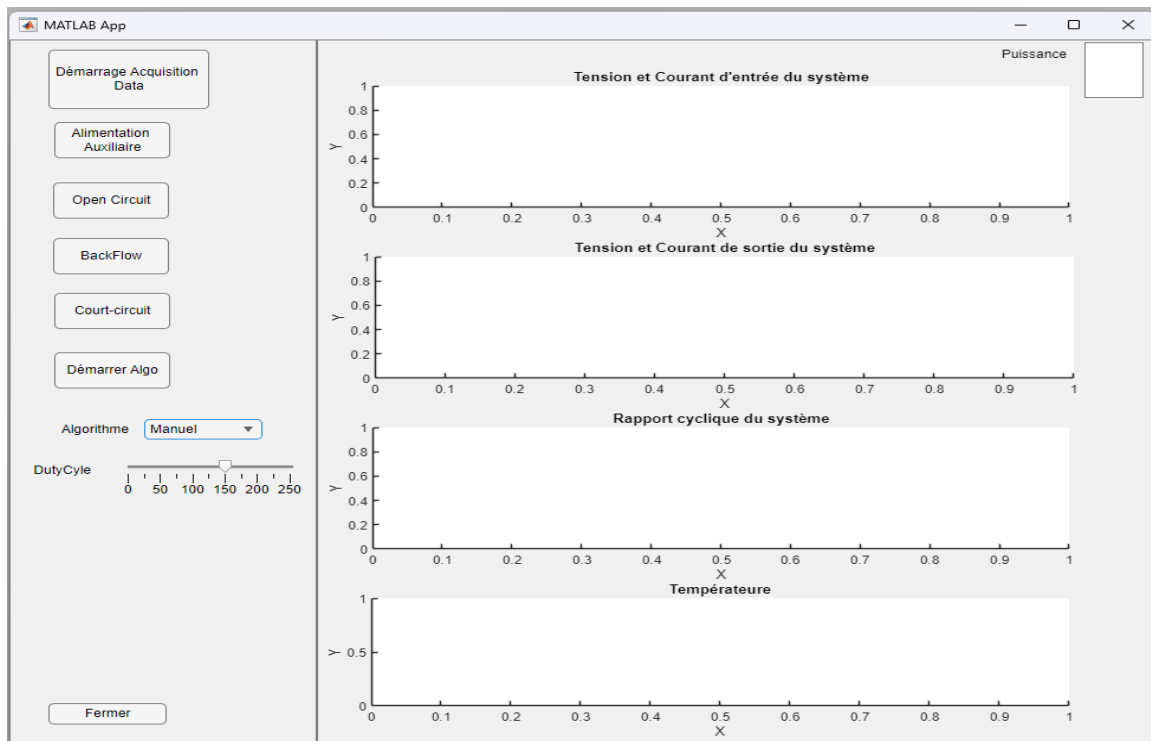


Figure 17 : Interface HMI "APP designer " de MATLAB.

Pour l'implémentation des Algorithmes MPPT, trois algorithmes distincts ont été programmés dans le système, soit la fonction de contrôle manuel, qui permet de paramétrer un rapport cyclique manuel, soit l'algorithme de perturbation et observation et finalement l'algorithme génétique.

4.2 Algorithme génétique

Pour ce qui est de l'algorithme génétique, l'implémentation prend plusieurs formes, en premier lieu, il faut prendre les données du système en circuit ouvert et en court-circuit. Cela permet ensuite de passer à la détermination des paramètres du panneau solaire tel que la résistance série (R_S) et la résistance en parallèle.

La prise de tension en circuit ouvert (V_{oc}) et la prise de courant en court-circuit (I_{sc}) sont des mesures essentielles pour caractériser un panneau photovoltaïque. Dans ce projet, ces valeurs sont obtenues à l'aide d'un microcontrôleur ESP32 et d'un traitement des données sous Matlab.

Pour mesurer V_{oc} , Matlab va lire la tension aux bornes du panneau solaire lorsque la routine de l'algorithme est enclenchée. Un convertisseur analogique-numérique (ADC) est utilisé pour capturer cette valeur, qui est ensuite envoyée à MATLAB via une communication série. MATLAB récupère les données et les stocke pour une analyse ultérieure.

De la même méthodologie que pour V_{oc} , la mesure de I_{sc} est réalisée par la lecture du capteur de courant lorsqu'un court-circuit est appliqué au système. Pour appliquer le court-circuit, un transistor en mode commutateur est utilisé. Les résultats sont envoyés vers MATLAB, où ils sont affichés et traités pour ajuster les paramètres du contrôleur MPPT.

Ces mesures permettent d'ajuster dynamiquement le fonctionnement du système MPPT et d'améliorer l'efficacité de la conversion d'énergie en fonction des variations des conditions environnementales.

4.2.1 Détermination des paramètres du panneau par les algorithmes génétiques

À la suite des prises de données du courant en court-circuit et de la tension en circuit ouvert, Il est possible d'utiliser l'algorithme pour déduire les paramètres de luminosité, de température, de résistance série et de résistance parallèle. Dans le contexte des systèmes photovoltaïques, les algorithmes génétiques sont utilisés pour déterminer les paramètres clés d'un panneau solaire, tels que l'irradiation solaire (G), la température (T), la résistance série (R_s) et la résistance shunt (R_{sh}) (figure.18). Ces paramètres sont essentiels pour modéliser avec précision le comportement électrique du panneau et optimiser son rendement. L'algorithme génétique opère en générant une population initiale de solutions potentielles, représentées sous forme de chromosomes, où chaque chromosome encode une combinaison possible des paramètres à optimiser. À travers des itérations successives, les algorithmes génétiques appliquent des opérations de sélection, croisement et mutation pour évoluer vers des solutions optimales. La fonction d'évaluation, basée sur des équations physiques décrivant le courant et la tension du panneau, permet de quantifier la performance de chaque solution. En comparant les résultats simulés aux données expérimentales, les algorithmes génétiques ajustent progressivement les paramètres pour minimiser l'erreur entre le modèle et la réalité. Cette approche permet une exploration efficace de l'espace des paramètres, même en présence de non-linéarités et de variations environnementales, offrant ainsi une méthode robuste pour l'extraction et l'optimisation

des caractéristiques des panneaux photovoltaïques. La démarche est expliquée en détail au chapitre 3. Pour ce qui est de la fonction de performance, la minimisation de l'erreur entre les données recueillies et le modèle obtenu à l'aide de l'algorithme génétique est utilisé.

La fonction de performance pour optimiser ces paramètres :

```
% Fonction objectif qui calcule l'erreur (différence de puissance)
function erreur = errorCalcul(params, Iscn, Vocn, Ki, Gn, Vt, Io, Ns, a, Pmes)
    Rs = params(1);
    Rsh = params(2);
    G = params(3);
    T = params(4);

    % Calcul de la température et de l'irradiance
    dT = T - 25; % Différence de température par rapport à la température nominale
    Ipvn = (Rs + Rsh) / Rsh * Iscn;
    Ipv = (Ipvn + Ki * dT) * G / Gn; % Courant ajusté en fonction de l'irradiance

    % Résolution de l'équation I-V pour plusieurs points de tension
    V = 0:1:Vocn; % Vecteur de tensions
    I = zeros(1, length(V)); % Vecteur de courants
    for j = 1:length(V)
        g = @(I) Ipv - Io * (exp((V(j) + I * Rs) / (Vt * Ns * a)) - 1) - (V(j) + I * Rs) / Rsh - I;
        I(j) = fsolve(g, 0); % Résolution de l'équation I-V
    end

    % Calcul de la puissance et de l'erreur
    P = (Ipv - Io * (exp((V + I * Rs) / Vt / Ns / a) - 1) - (V + I * Rs) / Rsh) .* V;
    Pmax_m = max(P); % Puissance maximale calculée
    erreur = abs(Pmax_m - Pmes); % Erreur (différence de puissance)
end
```

Figure 18 : Fonction de performance et paramètres du panneau solaire obtenus par les algorithmes génétiques.

4.2.2 Détermination de la tension et courant au maximum de puissance

Par la suite, il est possible d'utiliser les paramètres trouvés pour déterminer la tension qui donne le point maximum de puissance. L'algorithme génétique est encore utilisé pour déterminer les coordonnées du point de fonctionnement optimal (tension et courant) pour optimiser la puissance.

Reprenons le modèle du panneau photovoltaïque pour déterminer la fonction de performance utilisée pour l'algorithme génétique. Dans notre cas, la fonction d'aptitude est simplement la puissance de sortie du panneau photovoltaïque (PV), définie par :

$$P = V * I \quad (10)$$

où P est la puissance de sortie, V la tension et I le courant. L'objectif est de maximiser cette puissance pour trouver l'individu optimal.

L'équation (5) du chapitre 2 est utilisée pour trouver la puissance du système :

$$P = \left(I_{gen} - I_0 \left\{ e^{\frac{q(V+IR_s)}{nKT}} - 1 \right\} - \frac{V + IR_s}{R_{sh}} \right) * V \quad (11)$$

Pour ce qui est des étapes de l'algorithme génétique. Voici dans l'ordre les étapes implémentées dans MATLAB :

4.2.3 Création de la population

Pour la création de la population, le paramètre est modifiable à partir de l'écran de HMI du système est à la valeur par défaut de 100 membres. La valeur des membres de la population est contrainte entre la valeur 0 et le courant de court-circuit trouvée. Voici le code MATLAB de la population initiale de l'algorithme génétique:

```
% Initialisation de la population
population = I_min + (I_max - I_min) * rand(populationSize, 1);
```

4.2.4 Calcule des performances

Avec la fonction de performance déterminée plus haut, chaque membre de la population est testé et est attribuée un pointage. Voici le code MATLAB du calcul de la performance de l'algorithme génétique :

```
% Calcul de la fitness (valeurs de puissance pour chaque individu)
fitness = power_function(population);
```

4.2.5 Sélection des meilleurs individus

La sélection des individus s'effectue par un système élitisme. Le paramètre est modifiable à partir du HMI et la valeur par défaut est 20% des meilleurs résultats sont retenus. Voici le code Matlab de la sélection des meilleurs individus de l'algorithme génétique :

```

% Sélection des meilleurs individus (élitisme)
[~, sortedIdx] = sort(fitness, 'descend');
numElite = round(eliteFraction * populationSize);
newPopulation = population(sortedIdx(1:numElite));

```

4.2.6 Croisement des membres de la population

Le croisement des individus s'effectue par la moyenne des individus. Encore une fois, le paramètre est modifiable à partir du HMI et s'effectue sur toute la population qui a été retenu à la sélection. Voici le code MATLAB du croisement des individus de la population de l'algorithme génétique :

```

% Croisement
numOffsprings = populationSize - numElite;
for i = 1:numOffsprings
    parents = newPopulation(randperm(numElite, 2));
    newPopulation(end+1) = mean(parents); % Croisement simple
end

```

4.2.7 Mutation de la population

Finalement, la dernière étape de l'algorithme génétique est la mutation. C'est un paramètre qui est modifiable à partir du HMI Matlab. Le paramètre détermine la variation de chaque membre de la population. Voici le code MATLAB la mutation des individus de l'algorithme génétique

```

% Mutation
mutationMask = rand(size(newPopulation)) < mutationRate;
newPopulation(mutationMask) = I_min + (I_max - I_min) * rand(sum(mutationMask), 1);

```

4.2.8 Régulation de la tension VMPPT avec un contrôleur PID

Afin d'atteindre rapidement la valeur optimale de la tension VMPPT déterminée par l'algorithme génétique, un régulateur PID (Proportionnel-Intégral-Dérivé) est utilisé. Ce contrôleur ajuste dynamiquement le rapport cyclique du convertisseur DC-DC pour s'assurer que la tension d'entrée du panneau solaire se stabilise autour de VMPP. Le régulateur PID fonctionne en comparant la tension actuelle du panneau avec la valeur cible VMPPT. L'erreur entre ces deux valeurs est ensuite traitée à travers trois actions distinctes :

Terme proportionnel (P) : Réagit instantanément aux écarts en appliquant une correction proportionnelle à l'erreur.

Terme intégral (I) : Corrige les erreurs accumulées au fil du temps pour éliminer les écarts statiques.

Terme dérivé (D) : Anticipe les variations futures en se basant sur la vitesse de changement de l'erreur.

Pour déterminer les paramètres du PID, une identification du système a été faite grâce à l'outil PIDTuner dans MATLAB (figure 19).

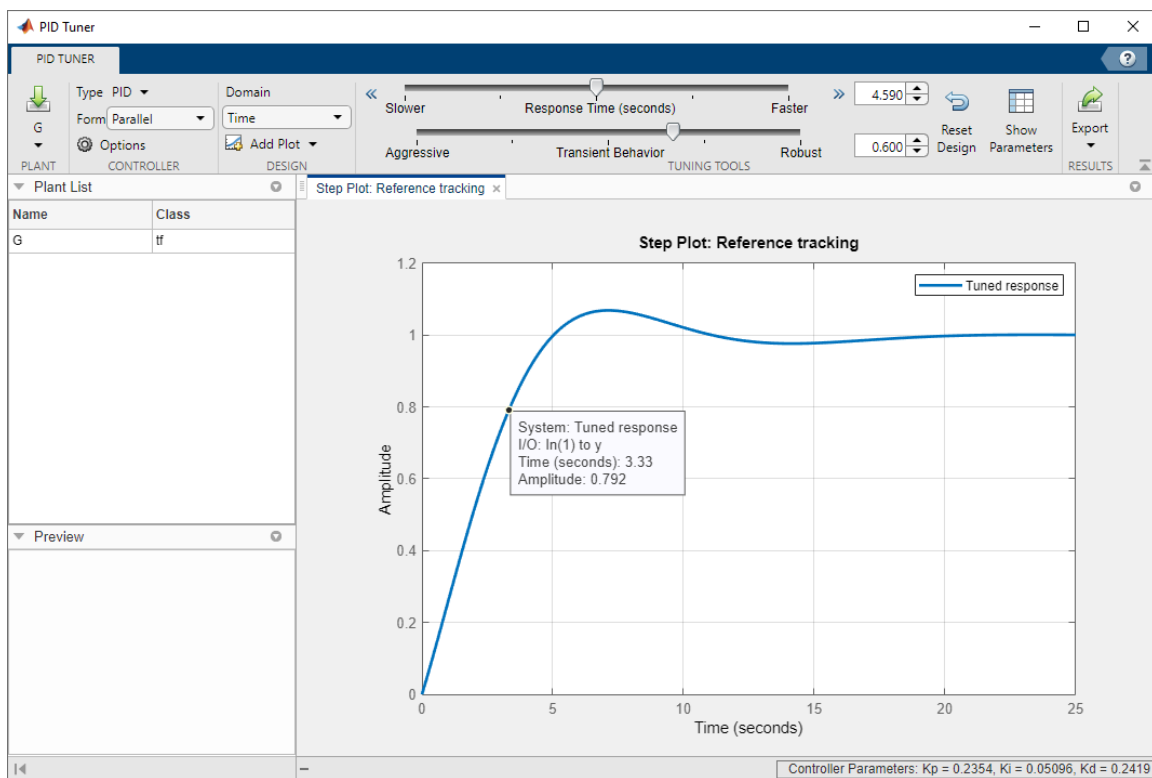


Figure 19 : Identification du système avec l'outil de PIDTuner de Matlab.

Pour effectuer l'identification du système, une réponse à un échelon a été utilisée. Le modèle de la fonction de transfert a un pôle utilisé est :

$$G(s) = \frac{K_p}{1 + T_p s} * e^{-T_d s} \quad (12)$$

Les paramètres du système obtenu sont :

$K_p = 0.0824$, $T_p = 0.84$, $T_d = 0.6$.

Avec la fonction de transfert, il est toujours possible de déterminer les paramètres du contrôleur avec PIDTuner. Il est donc possible de déterminer que les paramètres du contrôleur PID sont :

$K_p = 1$, $K_i = 0.017$, $K_d = 0$.

MATLAB calcule en temps réel la sortie du régulateur PID et ajuste le rapport cyclique du convertisseur DC-DC en conséquence. MATLAB est utilisé pour ajuster et optimiser les paramètres PID afin de garantir une réponse rapide et stable du système.

Grâce à cette approche, le contrôleur PID permet d'améliorer la précision et la stabilité de la régulation de tension, maximisant ainsi l'extraction d'énergie du panneau solaire tout en réduisant les fluctuations liées aux variations d'ensoleillement et de température.

4.3 La carte de commande

La carte de commande utilisée pour l'acquisition du courant et de la tension est à base du microcontrôleur ESP32 (figure 20). L'ESP32 est basé sur une architecture de microcontrôleur à double cœur Tensilica Xtensa LX6, cadencé jusqu'à 240 MHz. Le microcontrôleur prend en charge le protocole WiFi avec les normes 802.11 b/g/n/e/i (2,4 GHz) et 802.11 a/n/ac (5 GHz). Il prend aussi en charge le protocole Bluetooth v4.2 et Bluetooth Low Energy (BLE). Dispose de deux cœurs processeurs 32 bits, ce qui permet d'exécuter plusieurs tâches simultanément. La carte dispose d'une mémoire Flash d'une capacité de 16 Mo et une mémoire RAM de 4Mo.

Pour ce qui est des interfaces, la carte possède plusieurs GPIO (General Purpose Input/Output). Les principaux sont :

UART, SPI, I2C : Interfaces sérielles standard pour la communication avec d'autres périphériques.

ADC (Analog-to-Digital Converter) : Convertisseur analogique-numérique pour la lecture de signaux analogiques.

DAC (Digital-to-Analog Converter) : Convertisseur numérique-analogique pour la génération de signaux analogiques.

De plus, la carte ESP32 dispose d'une vaste gamme de bibliothèques logicielles pour différents périphériques et protocoles, avec un support pour des environnements de développement tels qu'Arduino IDE, ESP-IDF (ESP32 IoT Development Framework), MicroPython, etc.

Ces caractéristiques font de l'ESP32 un choix populaire pour une grande variété d'applications IoT. L'avantage principal de sélectionner l'ESP32 comme microcontrôleur est qu'il comprend des convertisseurs analogiques digitaux pour ces entrées et sorties. Ce qui permet de réduire l'interfaçage en utilisant directement une sortie du microcontrôleur pour effectuer la modulation de fréquence du convertisseur DC/DC. Il est possible de connecter directement la carte de prototypage au module de convertisseur DC/DC sans passer par une autre carte pour générer une fréquence modulable. De plus, le ESP32 possède 32 entrées et sorties interchangeables, ce qui permet de ne pas avoir à utiliser une carte d'extension d'entrées et sorties pour le système de MPPT.

ESP32 Wroom DevKit Full Pinout

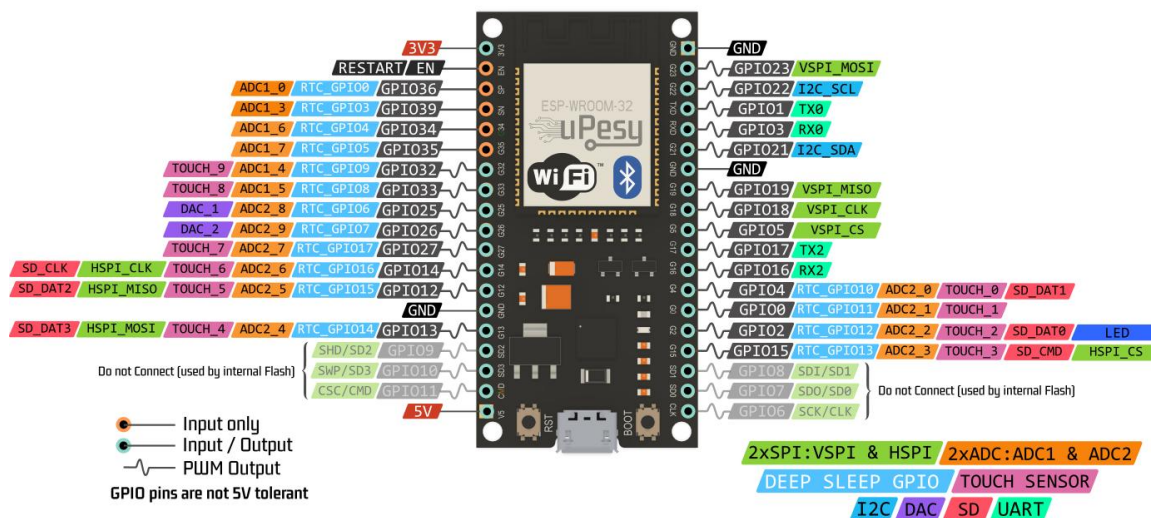


Figure 20 : Diagramme des Entrées/Sorties du ESP32.⁴

⁴ <https://www.espressif.com/>

4.3.1 Le modèle d'interface ESP32 et Matlab

La communication série entre MATLAB et un microcontrôleur ESP32, facilitée par l'utilisation du format JSON, offre une méthode efficace pour l'échange de données structurées. MATLAB, en tant qu'environnement de calcul et de simulation, dispose de bibliothèques permettant de configurer une communication série avec des dispositifs externes via des interfaces comme le port série (COM ou USB). Dans ce contexte, le format JSON est couramment utilisé pour structurer les données échangées, en raison de sa légèreté, de sa lisibilité et de sa facilité d'intégration avec des langages de programmation comme C, utilisé dans l'ESP32. Ce dernier, grâce à sa capacité de traitement et de gestion des communications série, peut recevoir des données JSON de MATLAB, les analyser et y répondre de manière appropriée. L'utilisation du JSON permet une abstraction des données, facilitant ainsi leur manipulation et leur interprétation tant côté MATLAB que côté ESP32, tout en assurant la compatibilité entre différents systèmes de traitement. En outre, l'implémentation de ce protocole de communication permet de maintenir une structure flexible et évolutive pour le développement de systèmes embarqués et d'applications IoT. Voici un exemple de la programmation qui effectue la communication à partir du ESP32 par le port série. Voici la programmation du ESP32 pour transférer des messages JSON sur le port série :

```
#include <ArduinoJson.h>

void setup() {
  Serial.begin(115200); // Initialisation de la communication série à 115200 bauds
}

void loop() {
  // Création d'un objet JSON
  StaticJsonDocument<200> doc;
  doc["temperature"] = 24.5;
  doc["humidity"] = 60;

  // Sérialisation de l'objet JSON en chaîne de caractères
  String jsonString;
```

```

serializeJson(doc, jsonString);

// Envoi de la chaîne JSON via la communication série
Serial.println(jsonString);

delay(1000); // Attente de 1 seconde avant le prochain envoi
}

```

4.3.2 Communication analogique du ESP32

La communication entre un ESP32 et des capteurs de courant et de tension, utilisant une plage de 0-3,3 V, repose sur la gestion précise des niveaux de signaux analogiques. L'ESP32, avec ses capacités de lecture analogique via ses broches ADC (convertisseur analogique-numérique), permet de mesurer des tensions allant de 0 à 3,3 V, ce qui est adapté aux signaux produits par de nombreux capteurs de courant et de tension, notamment ceux qui utilisent des sorties proportionnelles dans cette plage. Les capteurs tels que les shunt résistors pour la mesure du courant ou les capteurs de tension analogique génèrent des signaux de tension proportionnels à l'intensité du courant ou à la valeur de la tension mesurée, souvent dans la plage de 0-3,3 V. Ces signaux peuvent être directement intégrés à l'ESP32 grâce à ses entrées analogiques, après une éventuelle mise à l'échelle du signal. Les données collectées par l'ESP32 peuvent ensuite être traitées, analysées et envoyées.

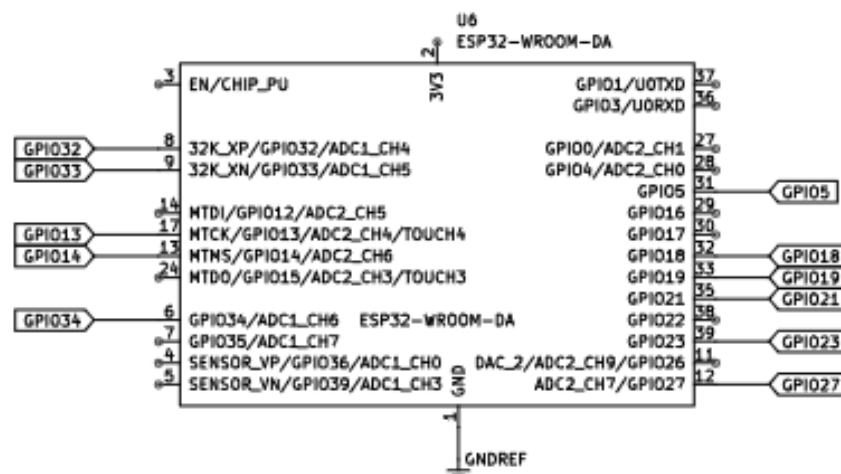


Figure 21 : Représentation du branchement des GPIO du ESP32.

4.4 Le montage du panneau solaire et de la source lumineuse

La source photovoltaïque utilisée constitue un arrangement de 36 cellules solaires branchées en série pour monter un panneau solaire de 40W tel que montré à la Figure 22. Cette figure montre aussi la source de lumière qui alimente les panneaux. Il s'agit de quatre lampes halogènes fournissant en moyenne 500W selon les spécifications techniques du produit.

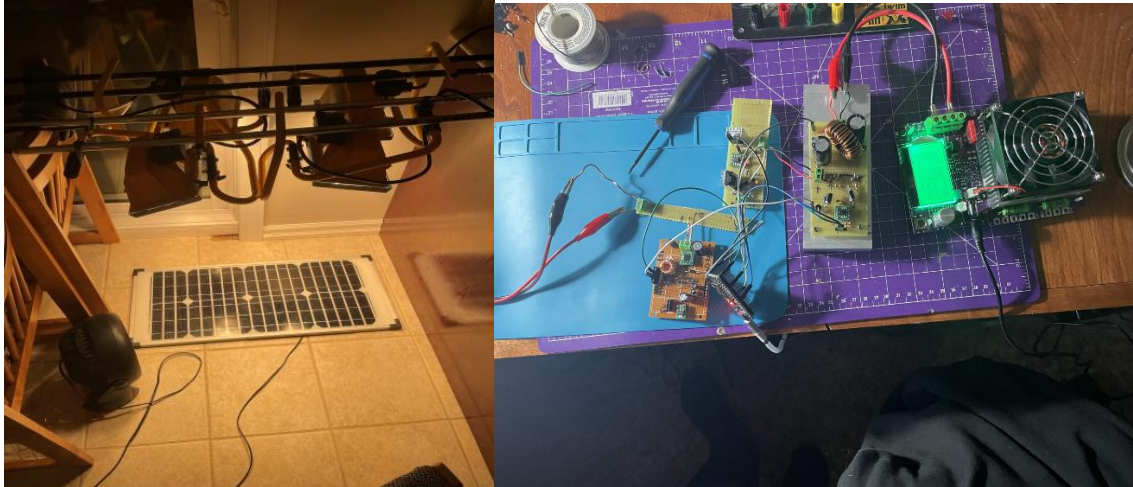


Figure 22 : Banc des tests expérimentaux.

4.5 Le montage du convertisseur DC/DC

La méthode de conception adoptée vise à optimiser l'efficacité et la simplicité. À cette fin, une topologie de convertisseur DC/DC abaisseur synchrone a été choisie pour son rendement élevé. Étant donné que l'un des objectifs est de créer un banc d'essai, la topologie du convertisseur DC/DC abaisseur synchrone a été retenue, car dans des conditions de laboratoire, il sera toujours possible d'utiliser une charge résistive pour transférer l'énergie des panneaux solaires, sans contrainte de tension de batterie à respecter. Les principaux paramètres d'un convertisseur DC/DC abaisseur synchrone ont été discutés au chapitre 2 et les données utilisées pour le dimensionnement du convertisseur ont été prises avec les paramètres du panneau solaire discuté au chapitre 4.

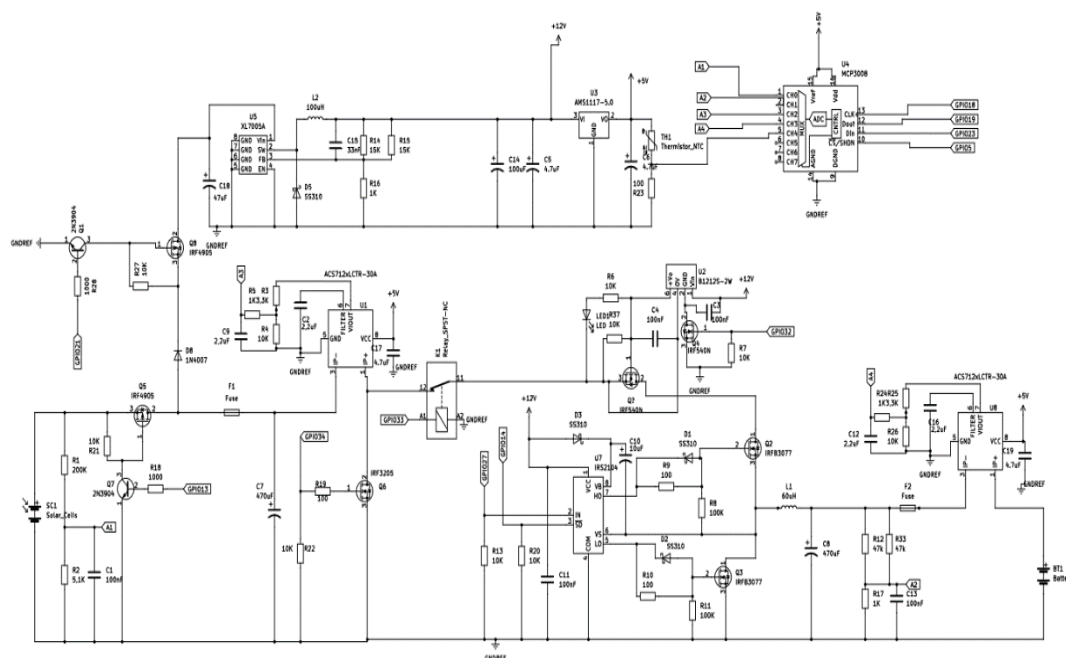


Figure 23 : Schéma électrique du convertisseur DC/DC et de la carte commande.

4.5.1 Source lumineuse pour les tests expérimentaux

La source lumineuse utilisée pour le système est quatre lampes halogènes de 500W. Voici une représentation de la lumière utilisée :



Figure 24 : Source lumineuse pour les tests expérimentaux (lampe halogène de 500w).⁵

4.5.2 Capteur de courant

⁵<https://www.homedepot.ca/produit/globe-electric-lampe-de-travail-portable-500-watt-finition-en-jaune-et-noir/1000821522>

Le capteur de courant ACS711 utilisé est un capteur à effet Hall qui permet de mesurer le courant en utilisant un signal analogique proportionnel à l'intensité du courant traversant un conducteur (figure.25). Il fournit une sortie analogique, généralement dans la plage de 0 à 3,3 V, qui varie en fonction du courant mesuré, avec une tension de référence au centre (1.65V) lorsqu'il n'y a pas de courant. Le capteur ACS711 est particulièrement adapté pour être utilisé avec des microcontrôleurs comme l'ESP32, car il génère des signaux dans la plage de tension compatible avec les entrées analogiques de l'ESP32, qui peuvent mesurer des tensions de 0 à 3,3 V via ses broches ADC. Pour un courant de 0 A, la sortie du capteur sera typiquement proche de la tension de référence (1,65V), et pour des courants positifs ou négatifs, cette valeur de référence augmentera ou diminuera en fonction de l'intensité du courant. L'ESP32 peut alors lire cette tension analogique, la convertir en une valeur numérique, et en déduire la valeur du courant en utilisant une simple formule mathématique tenant compte des caractéristiques spécifiques du capteur (comme la sensibilité en mV par ampère).

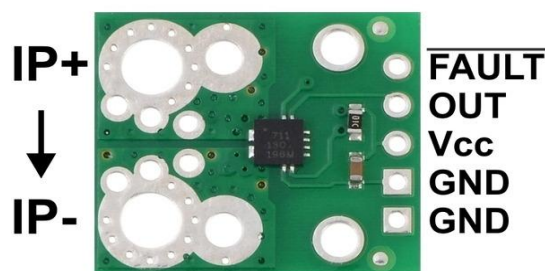


Figure 25 : Capteur de courant ACS711 ⁶

4.5.3 Capteur de tension

Pour les capteurs de tension, un système de pont diviseur de tension est utilisé. Le pont diviseur de tension est le moyen plus simple d'abaisser une tension continue. Il est constitué de deux résistances de base en série. Le pont diviseur de tension est constitué de deux résistances, R1 et R2, connectées en série. Une tension d'entrée U est appliquée aux bornes de cette association de résistances, tandis que la tension de sortie est mesurée aux bornes de la

⁶ <https://www.pololu.com/product/2197>

résistance R_2 . Les deux résistances sont également reliées à la masse, formant ainsi un circuit simple et efficace pour diviser la tension d'entrée en une tension de sortie proportionnelle.

$$V_{out} = V \frac{R_2}{R_1 + R_2} \quad (13)$$

4.5.4 Charge du système

L'utilisation d'une charge électronique pour varier la résistance dans un circuit permet de simuler et de contrôler des conditions de charge variées avec une grande précision. Contrairement à des résistances fixes ou variables classiques, une charge électronique offre la possibilité de modifier dynamiquement la résistance en réponse à des paramètres externes, comme la tension ou le courant. Cela permet de tester des sources d'alimentation, des circuits de régulation, ou des capteurs sous différentes conditions de charge tout en mesurant précisément les variations du courant et de la tension. Dans notre système de mesure de courant et de tension utilisant le microcontrôleur ESP32, la charge électronique est utilisée pour ajuster la résistance du circuit, ce qui modifie le courant circulant dans le circuit. Cette variation de résistance peut être contrôlée par une interface numérique, offrant ainsi une flexibilité accrue dans le développement et le test de dispositifs électroniques. Lorsqu'elle est utilisée avec des capteurs comme l'ACS711 pour mesurer le courant, la charge électronique permet de simuler différents scénarios de charge tout en mesurant l'impact sur le système de manière précise. Cette approche est particulièrement utile dans les bancs d'essai et les simulations de conditions variables.



Figure 26 : Charge électronique utilisée dans le banc d'essai. ⁷⁷

⁷⁷ <https://www.aliexpress.com/item/32779830400.html>

5. ESSAIS EXPÉRIMENTAUX ET COMPARAISON DES ALGORITHMES MPPT

Ce chapitre présente les résultats des essais expérimentaux réalisés pour comparer les performances des algorithmes de suivi du point de puissance maximale basés sur la méthode de Perturbation et Observation et l'algorithme génétique. L'objectif principal de ces tests est d'évaluer l'efficacité et la robustesse de ces deux approches dans des conditions reflétant des scénarios d'exploitation de systèmes photovoltaïques.

Les essais ont été menés sous deux configurations distinctes : une première configuration simulant une illumination complète du panneau solaire (quatre lumières halogènes de 500W), et une seconde configuration reproduisant des conditions d'ombrage partiel (deux lumières halogènes de 500W). Ces deux scénarios permettent de mettre en lumière les différences de comportement des algorithmes face à des variations d'ensoleillement, un défi courant dans les applications photovoltaïques.

Les résultats obtenus dans ce chapitre illustrent les performances des deux algorithmes MPPT en termes de courant, tension, puissance et rapport cyclique. Ces paramètres sont essentiels pour comprendre comment chaque algorithme ajuste le point de fonctionnement du système photovoltaïque pour maximiser l'extraction d'énergie. Les résultats obtenus dans des conditions d'éclairement total permettent de comparer les performances optimales des deux méthodes, tandis que les tests en conditions d'ombrage mettent en évidence leur capacité à s'adapter à des environnements plus complexes et variables.

5.1 Comparaison des algorithmes génétique et P&O dans les conditions normales (éclairement total)

La figure 27 illustre la comparaison entre les courants et les tensions d'entrée mesurés au niveau du panneau solaire. Les courbes représentées en rouge correspondent aux résultats obtenus avec le contrôleur MPPT utilisant l'algorithme de Perturbation et Observation, tandis que les courbes en bleu reflètent les performances du contrôleur MPPT piloté par l'algorithme génétique. Cette comparaison permet d'analyser les différences de comportement entre les deux approches en termes de régulation du courant et de la tension, essentiels pour l'optimisation de l'extraction de puissance dans un système photovoltaïque. Il est à noter que les essais de l'algorithme P&O, le système était apporté près du point MPP avant le démarrage de l'acquisition.

5.1.1 Courant et tension mesuré au panneau solaire dans des conditions normales

Pour l'algorithme génétique, le graphique de courant montre que le régulateur PID converge dans les 10 premières secondes de l'essai, stabilisant le courant à une moyenne de 0,52 A avec des oscillations entre 0,5 A et 0,55 A. En revanche, pour l'algorithme de Perturbation et Observation, le graphique de courant indique que l'algorithme a rencontré des difficultés à atteindre le point de puissance maximale à 28 secondes, avec des oscillations plus prononcées, variant entre 0,45 A et 0,7 A.

Concernant la tension, le graphique associé à l'algorithme P&O présente des oscillations significatives, allant de 18 V à 22 V. En comparaison, l'algorithme génétique montre des oscillations de tension plus faibles, se situant entre 18 V et 19,5 V, témoignant d'une meilleure stabilité dans la régulation de la tension.

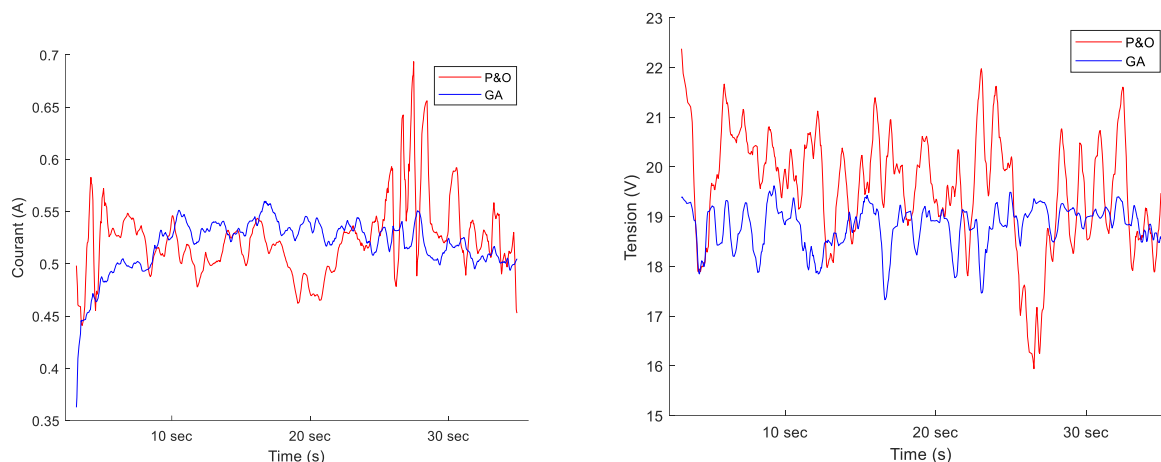


Figure 27 : Le courant (gauche) et la Tension (droite) mesurés au panneau solaire lors d'essai obtenus avec les algorithmes P&O et génétique.

5.1.2 Puissance au panneau solaire dans des conditions d'illumination

La comparaison des puissances d'entrée révèle que l'algorithme de Perturbation et Observation a extrait en moyenne 10,1 W du panneau solaire, tandis que l'algorithme génétique a extrait en moyenne 9,8 W. Cependant, l'algorithme P&O présente des oscillations de puissance plus importantes, variant entre 8 W et 14 W, alors que pour l'algorithme génétique, les oscillations de puissance sont plus faibles, se situant entre 9 W et 10,7 W.

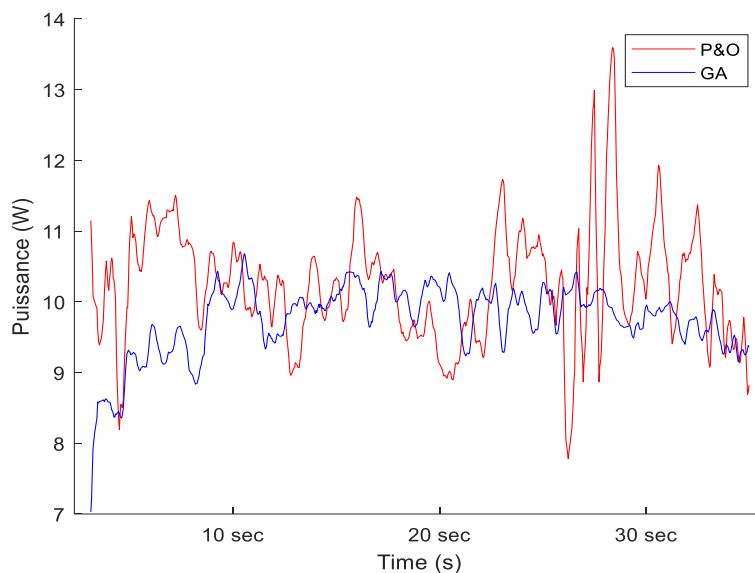


Figure 28 : Comparaison de la puissance d'entrée entre l'algorithme P&O et génétique.

5.1.3 Courant et tension de la charge dans des conditions normales

Les courbes de courant et de tension de la charge suivent une tendance similaire à celles observées pour le courant et la tension photovoltaïques. Cependant, des pertes systémiques réduisent légèrement les valeurs mesurées. Ainsi, le courant moyen en charge pour l'algorithme de Perturbation et Observation est de 0,68 A, tandis que pour l'algorithme génétique, il atteint en moyenne 0,71 A. Les oscillations au niveau de la charge persistent, avec des variations comprises entre 0,6 A et 0,75 A pour le P&O, alors que pour l'algorithme génétique, le courant oscille entre 0,69 A et 0,75 A.

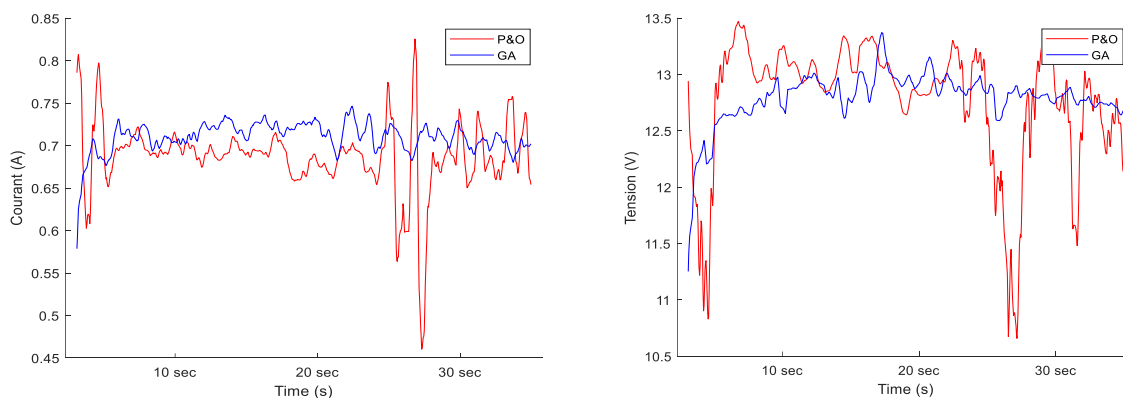


Figure 29 : Graphique du courant à la charge (gauche) et de la tension a la charge (droite) pour les algorithmes P&O et génétique.

5.1.4 Puissance de la charge et rapport cyclique dans des conditions normales

Le figure 30 de la puissance de charge met en évidence une corrélation générale entre les performances des deux algorithmes en termes de puissance de sortie. En ce qui concerne la puissance délivrée à la charge, la valeur moyenne mesurée est de 9,1 W pour l'algorithme de Perturbation et Observation, légèrement inférieure aux 9,34 W générée par l'algorithme génétique. Des oscillations de puissance sont observées pour l'algorithme P&O, variant entre 8,31 W et 9,74 W. Une perturbation notable est également détectée entre 40 et 50 secondes, affectant temporairement l'algorithme P&O. Cependant, celui-ci parvient à retrouver le point de puissance maximale par la suite, confirmant sa robustesse face à des variations transitoires.

Pour l'algorithme génétique, la variation du rapport cyclique peut être interprétée en deux phases distinctes. Premièrement, une augmentation de 173 à 192 unités correspond à l'action du régulateur PID pour atteindre la consigne de tension V_{mpp} . Ensuite, le rapport cyclique diminue afin de maintenir la tension à la valeur optimale déterminée par l'algorithme génétique. Cette adaptation peut être attribuée aux variations de luminosité des lampes halogènes et à l'augmentation de la température, nécessitant des ajustements du rapport cyclique pour maintenir une extraction optimale de l'énergie.

En ce qui concerne l'algorithme P&O, le graphe du rapport cyclique illustre les oscillations caractéristiques de la recherche du MPP. Une perturbation notable est également observée entre 40 et 50 secondes, suivie d'une stabilisation du rapport cyclique autour d'une valeur approximative de 220 unités. Cette stabilisation indique que l'algorithme a réussi à retrouver une zone proche du point de puissance maximale après la perturbation, démontrant ainsi sa capacité à s'ajuster et à converger vers des conditions optimales.

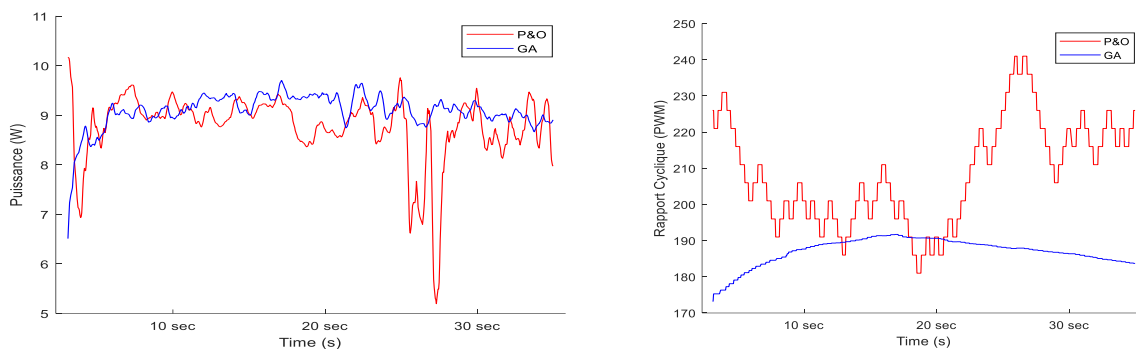


Figure 30 : Puissance de charge (gauche) et rapport cyclique (droite) du système pour chaque l'algorithme P&O et génétique.

5.2 Comparaison des algorithmes GA et P&O dans des conditions ombragées (éclairage partiel)

Les graphiques ci-dessous illustrent une comparaison entre les courants et les tensions d'entrée mesurés directement au niveau du panneau solaire. L'algorithme de Perturbation et Observation est représenté par la courbe rouge, tandis que l'algorithme génétique est indiqué par la courbe bleue. La condition d'ombrage est simulée en utilisant seulement deux lampes halogènes de 500W au lieu de quatre comme lors des essais dans les conditions normales.

5.2.1 Courant et tension au panneau solaire dans des conditions d'ombragées

Le courant maximal atteint par l'algorithme P&O est de 0,27 A, tandis que celui de l'algorithme génétique est de 0,265 A. Les oscillations du courant sont plus marquées avec l'algorithme P&O, où il fluctue entre 0,18 A et 0,27 A, alors qu'avec l'algorithme génétique, il varie dans une plage plus restreinte, entre 0,23 A et 0,26 A.

Concernant la tension d'entrée, on observe des tendances similaires à celles du courant. L'algorithme génétique maintient une tension proche de 19 V, avec de légères variations comprises entre 18,76 V et 19,35 V. En revanche, avec l'algorithme P&O, les variations sont plus importantes, oscillant entre 16,75 V et 19,65 V. Ainsi, la stabilité de la tension d'entrée est nettement meilleure avec l'algorithme génétique.

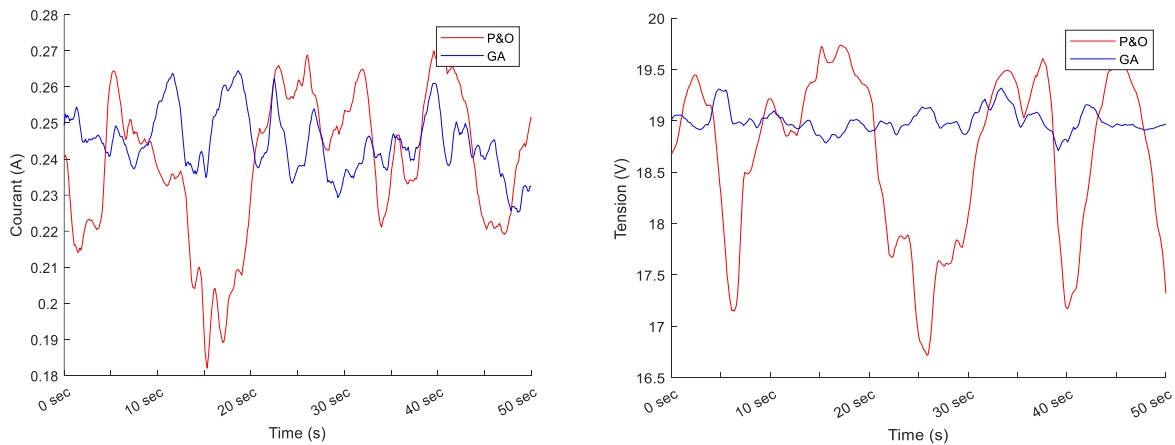


Figure 31 : Graphique du courant (gauche) et de la Tension (droite) mesuré au panneau solaire lors d'essai fait avec les algorithmes P&O et GA dans des conditions d'ombrages

5.2.2 Puissance du panneau solaire dans des conditions ombragées

La puissance d'entrée, qui résulte de la combinaison du courant et de la tension, montre qu'avec deux lampes en moins, la puissance extraite du panneau solaire est en moyenne de 4,65

W pour l'algorithme génétique et de 4,48 W pour l'algorithme P&O. Cela représente une réduction d'un peu plus de la moitié par rapport aux valeurs obtenues dans des conditions d'éclairage avec quatre lampes halogènes, où la puissance était de 10,1 W pour l'algorithme génétique et de 9,8 W pour l'algorithme P&O.

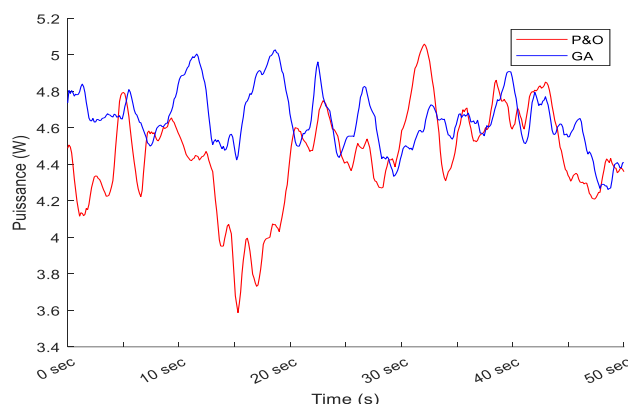


Figure 32 : Comparaison de la puissance d'entrée entre l'algorithme P&O et GA pour des conditions d'illumination ombragée

5.2.3 Courant et tension à charge dans des conditions ombragées

Pour le courant et la tension à la charge, l'algorithme P&O atteint un maximum de 0,49 A, avec des oscillations comprises entre 0,41 A et 0,49 A, tandis que l'algorithme génétique atteint un maximum de 0,48 A, oscillant entre 0,42 A et 0,48 A. En ce qui concerne la tension, l'algorithme génétique atteint un maximum de 8,3 V, avec des variations plus faibles, oscillant entre 8,1 V et 8,3 V. En revanche, l'algorithme P&O atteint un maximum de 8,5 V, mais avec des oscillations plus importantes, variant entre 7 V et 8,7 V.

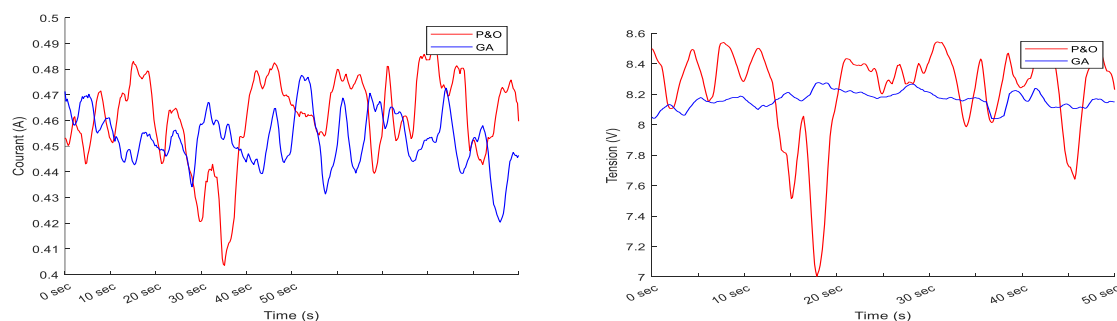


Figure 33 : Graphique du courant à la charge (gauche) et de la tension à la charge (droite) pour les algorithmes P&O et génétique avec des conditions ombragées.

5.2.4 Puissance à la charge et rapport cyclique dans des conditions ombragées

Comme lors de l'essai en pleine luminosité, les graphes montrent que la puissance se maintient en moyenne à 3,7 W pour l'algorithme génétique et à 3,8 W pour l'algorithme P&O. Cette diminution était attendue par rapport aux tests en pleine luminosité, où les puissances atteignaient respectivement 9,34 W et 9,31 W. Cependant, on observe que grâce au régulateur PID implémenté pour l'algorithme génétique, la puissance reste plus stable, avec des variations moins prononcées que celles de l'algorithme P&O. Malgré cela, l'algorithme P&O a permis d'extraire en moyenne 0,1 W de plus que le GA, bien que cette différence soit minime.

Une analyse suggère que le paramètre du pas de PWM de l'algorithme P&O pourrait être optimisé pour réduire l'amplitude de ses oscillations, améliorant ainsi la stabilité et l'efficacité du système. En comparaison, l'algorithme génétique démontre une excellente capacité à s'adapter aux conditions d'ombrage, ajustant dynamiquement le point de fonctionnement pour maximiser la puissance extraite.

Concernant le rapport cyclique, on observe les oscillations typiques de l'algorithme P&O, qui varie continuellement pour rechercher le point de puissance maximale. Dans cet essai, le PWM oscille entre 100 et 135. En revanche, pour l'algorithme génétique, la PWM se stabilise autour de 110, car c'est le PID qui prend le relais pour atteindre et maintenir le point de puissance maximale de manière plus stable.

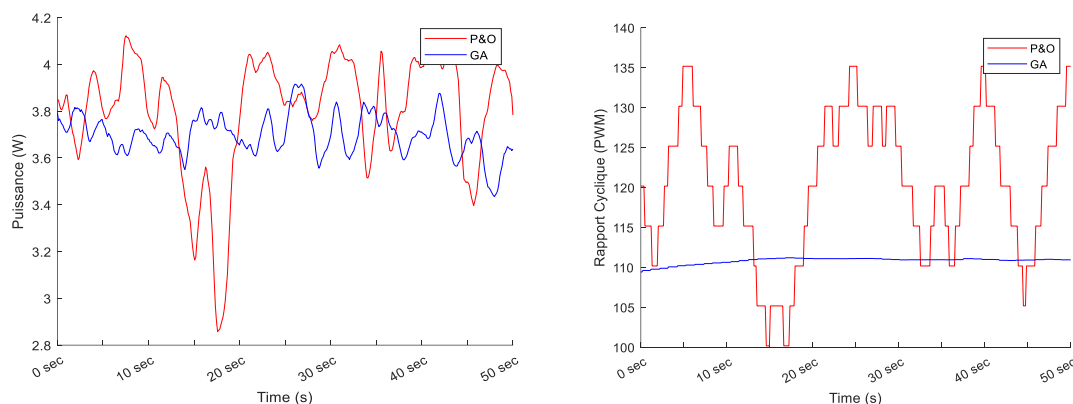


Figure 34 : Puissance de charge (gauche) et rapport cyclique (droite) du système pour chaque l'algorithme P&O et génétique dans des conditions ombragées.

À travers ces comparaisons, ce chapitre démontre les avantages et les limites de chaque algorithme, en mettant l'accent sur leur capacité à maintenir une extraction d'énergie optimale dans des conditions changeantes. Les conclusions tirées de ces essais expérimentaux fourniront des connaissances précieuses pour l'optimisation future des systèmes MPPT, en particulier dans des environnements où les conditions d'éclairage sont fluctuantes et imprévisibles.

CONCLUSION

Ce mémoire a exploré l'optimisation de l'extraction de puissance dans un système photovoltaïque à l'aide d'une technique de suivi du point de puissance maximale basée sur un algorithme génétique. L'objectif principal était de comparer les performances de cet algorithme avec une méthode classique, à savoir l'algorithme de Perturbation et Observation (P&O), tout en concevant et en validant un banc de test fonctionnel pour l'expérimentation. Les résultats obtenus démontrent que la réalisation du banc de test a été un succès, permettant une évaluation précise et comparative des deux algorithmes dans des conditions variées d'illumination et ce avec la prise de donnée en temps réel.

Le banc de test a été conçu en intégrant un panneau photovoltaïque, un convertisseur DC/DC, un microcontrôleur ESP32 et une interface MATLAB pour l'implémentation des algorithmes. Les essais expérimentaux ont été menés sous deux configurations distinctes : une illumination complète et une illumination partielle (ombrage). Ces conditions ont permis de simuler des variations réalistes de luminosité, similaires à celles rencontrées dans des environnements réels. La réussite de la mise en place de ce banc de test a été confirmée par la capacité à mesurer avec précision les paramètres électriques (tension, courant, puissance) et à ajuster dynamiquement le rapport cyclique du convertisseur pour maximiser l'extraction d'énergie à partir de MATLAB.

Le mémoire a été structuré en cinq chapitres principaux, chacun contribuant à la compréhension et à la réalisation du projet. Le chapitre 1 a introduit le contexte de l'énergie solaire et les enjeux liés à l'optimisation de l'extraction d'énergie, tout en présentant les objectifs du projet. Le chapitre 2 a détaillé les composants du système photovoltaïque, incluant la modélisation d'un panneau photovoltaïque, le convertisseur DC/DC et le contrôleur MPPT. Le chapitre 3 a approfondi les algorithmes MPPT, en discutant de l'algorithme génétique et son application dans le contexte du suivi du point de puissance maximale. Le chapitre 4 a décrit la conception et l'implémentation du système, incluant l'interface Matlab, le microcontrôleur ESP32 et le montage expérimental. Enfin, le chapitre 5 a présenté les résultats expérimentaux obtenus, comparant les performances des algorithmes génétiques et P&O dans différentes conditions d'utilisation.

Les résultats expérimentaux ont montré que l'algorithme génétique a obtenu des performances supérieures à celles de l'algorithme P&O. Bien que la différence de performance soit relativement faible, l'algorithme génétique a démontré une meilleure stabilité, avec moins d'oscillations autour du point de puissance maximale. Cette stabilité est attribuée à l'utilisation

d'un contrôleur PID pour réguler la tension, ce qui a permis de réduire les fluctuations et d'améliorer la convergence vers le point de puissance maximale. Ce travail ouvre la voie à des recherches futures visant à développer de nouvelles méthodes MPPT basées sur les algorithmes intelligents et métaheuristique pour l'extraction d'énergie encore plus efficace dans les systèmes photovoltaïques. Le banc d'essai développé permettra aussi de tester les nouvelles topologies de convertisseur DC/DC et les nouvelles méthodes MPPT dans les conditions réelles de fonctionnement.

Le mémoire apporte une perspective innovante dans sa conception, notamment par la méthode d'implémentation de l'algorithme génétique (GA) directement sur une plateforme MATLAB. Ce choix technologique permet de tirer parti de la puissance de calcul suffisante. De plus, l'utilisation d'un microcontrôleur ESP32 dans la conception permet la faible consommation énergétique tout en conservant une flexibilité de développement. L'intégration des méthodes de communication, offre une interface de supervision et de contrôle efficace. L'approche Hardware-in-the-Loop (HIL) confère au système un haut niveau de paramétrisation en temps réel, en permettant une interaction directe entre le modèle algorithmique et les composants physiques, ce qui favorise une évaluation précise des performances en conditions quasi-réelles.

En termes de perspectives, l'utilisation de composants standards, simples et largement disponibles confère au banc d'essai un avantage significatif en matière de coût et d'accessibilité, sans compromettre ses performances. Sa facilité d'implémentation constitue un atout majeur, rendant cette plateforme aisément adaptable aux besoins spécifiques des parcs énergétiques. Elle peut être préalablement dimensionnée selon les contraintes du site cible et livrée sous une forme personnalisée, ce qui en fait une solution flexible et scalable, parfaitement adaptée aux dynamiques évolutives du secteur énergétique.

Le banc d'essai offre également un potentiel considérable pour l'expérimentation et l'optimisation d'une variété d'algorithmes MPPT, au-delà de l'algorithme génétique actuellement implémenté. Grâce à son architecture modulaire et à la capacité de reprogrammation de l'ESP32, il est possible d'intégrer des approches de techniques avancées basées sur l'intelligence artificielle, comme les réseaux de neurones ou les systèmes flous. Cette flexibilité ouvre la voie à une analyse comparative rigoureuse des performances de chaque méthode dans des conditions environnementales variables simulées par le système HIL. Ainsi, le banc devient un outil évolutif de recherche et de développement, apte à soutenir l'amélioration continue des stratégies de suivi du point de puissance maximale, en adéquation avec les

exigences croissantes en matière de rendement énergétique et d'adaptabilité des systèmes photovoltaïques.

En somme, le banc d'essai développé se distingue par son originalité, sa flexibilité et sa capacité à intégrer des algorithmes avancés de suivi du point de puissance maximale dans un environnement réaliste grâce à l'approche Hardware-in-the-Loop. L'utilisation de composants accessibles comme l'ESP32, couplée à une architecture évolutive, en fait une plateforme d'expérimentation robuste et adaptable, à la fois pour la validation d'algorithmes existants et le développement de nouvelles stratégies MPPT. Sa facilité de déploiement et son potentiel de personnalisation en fonction des besoins spécifiques du secteur énergétique renforcent sa pertinence comme outil de recherche appliquée et de prototypage rapide. Ce banc d'essai ouvre ainsi la voie à une exploration continue et à l'amélioration des performances des systèmes photovoltaïques de demain.

ANNEXE A – PROGRAMME ESP32

Annexe A: Programme ESP32

```

#include "WiFi.h"
#include "PubSubClient.h"
#include "Wire.h"
#include <Adafruit_MCP3008.h>
#include <ArduinoJson.h>

//const char *ssid = "Bourni"; // Entrez votre SSID WiFi
//const char *password = "allo1111"; // Entrez votre mot de passe WiFi
/// WiFi
const char *ssid = "Volta ouh"; // Entrez votre SSID WiFi
const char *password = "Allo1111"; // Entrez votre mot de passe WiFi

/// MQTT Broker

//const char *mqtt_broker = "172.20.10.6";
//const char *mqtt_broker = "192.168.0.145";

//publish
StaticJsonDocument<300> JSONData;

const char *CourantINTopic = "ESP32/MPPT/CourantIN";
const char *CourantOUTtopic = "ESP32/MPPT/CourantOUT";
const char *TensionINTopic = "ESP32/MPPT/TensionIN";
const char *TensionOUTtopic = "ESP32/MPPT/TensionOUT";
const char *DutyCycleVAlueTopic = "ESP32/MPPT/DutyCyleValue";

//subscribe

const char *JSONDATACTRTopic = "ESP32/MPPT/JSONDATACTR";

const char *mqtt_username = "AigDHBQpDzYqLgItKQYvFjo";
const char *mqtt_password = "C5UA2b7l+JudOXg9RkCAJau1";
const int mqtt_port = 1883;

/// variable server et MQTT
WiFiClient espClient;
PubSubClient client(espClient);

///assignation constante Pin de controle
const int backFlowPIN = 12;

```

```

const int OCPIN = 13;
const int PinSub = 21;
const int CCProtPin = 22;
const int CCPin = 5;

///assignation constante Pin IR2184
const int PWMPIN = 27;
const int PINS_D = 14;

///Constante IR2184
const int freq = 50000;
const int ledChannel = 0;
const int resolution = 8;
float dutyCycle = 50;

/// Assignation Pin analogique
const int TensionINPIN = 34;
const int CourantINPIN = 35;
const int TensionOUTPIN = 33;
const int CourantOutPin = 32;

///Variable analogique
double TensionIN = 0;
double TensionOut = 0;

double CourantInINter;
double CourantIn = 0;
double courantOUT = 0;

///moyennage des donnée d'entrée
const int numlecture = 50;
double TensionTAB[numlecture];
double TensionOUTTAB[numlecture];
double CourantOUTTAB[numlecture];
double CourantINTAB[numlecture];
double CalibTab[numlecture];
int lectureIndex = 0;           // the index of the current reading

int count = 10;
unsigned long previousMillis = 0;
unsigned long interval = 1500;

///calibration
double calibrationResultCourantIN = 0;

```

```

double calibrationResultCourantOUT =0;
unsigned long previousMillis2 = 0;
unsigned long interval2 = 5000;

///initiation WIFI
void initWifi()
{
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.println("Connecting to WiFi..");
    }
    Serial.println("Connected to the Wi-Fi network");
}

///initiation MQTT
void initMQTT()
{
    client.setServer(mqtt_broker, mqtt_port);

    while (!client.connected()) {
        String client_id = "esp32-client-";
        client_id += String(WiFi.macAddress());
        Serial.printf("The client %s connects to the public MQTT broker",
client_id.c_str());
        if (client.connect(client_id.c_str(), mqtt_username, mqtt_password)) {
            client.subscribe(JSONDATACTRTopic);
            Serial.println("Public EMQX MQTT broker connected");
        } else {
            Serial.print("failed with state ");
            Serial.print(client.state());
            delay(2000);
        }
    }
}

void PublishMQTT()
{

```

```

    if(!client.connected()){
        client.disconnect();
        String client_id = "esp32-client-";
        client_id += String(WiFi.macAddress());
        Serial.printf("The client %s connects to the public MQTT brokern",
client_id.c_str());
        client.connect(client_id.c_str(), mqtt_username, mqtt_password);
        Serial.println("Public EMQX MQTT broker connected");
    }

```

```

JSONData["CourantIN"] = CourantIn;
JSONData["CourantOUT"] = courantOUT;
JSONData["TensionIN"] = String(TensionIN,2);
JSONData["TensionOut"] = String(TensionOut,2);
JSONData["dutyCycle"] = dutyCycle;

```

```

char out[256];
int b = serializeJson(JSONData,out);
boolean dataout= client.publish("ESP32/MPPT/DATAJSON", out);
}

```

```

float mapfloat(long x, long in_min, long in_max, long out_min, long out_max)
{
    return (float)(x - in_min) * (out_max - out_min) / (float)(in_max - in_min) +
out_min;
}

```

```

double MoyenneAnalogue(double TableauValeur[],int DIMtableau, int indexAnalogue,
double ValeurLu)
{
    unsigned int index;
    double total;

    total = 0;
    for(index=0; index < DIMtableau;index++)
    {
        total+=TableauValeur[index];
    }

    total = total - TableauValeur[indexAnalogue];
    TableauValeur[indexAnalogue] = ValeurLu;
}

```

```

    total = total + ValeurLu;

return total/DIMtableau;
}

double ReadVoltage(byte pin){
    double reading = analogRead(pin); // Reference voltage is 3v3 so maximum reading
is 3v3 = 4095 in range 0 to 4095
    if(reading < 1 || reading > 4095) return 0;
    // return -0.000000000009824 * pow(reading,3) + 0.000000016557283 *
pow(reading,2) + 0.000854596860691 * reading + 0.0654403483454;
    return -0.000000000000016 * pow(reading,4) + 0.000000000118171 * pow(reading,3)-
0.000000301211691 * pow(reading,2)+ 0.001109019271794 * reading +
0.034143524634089;
}

double calibrationACS712(int pin)
{
    int NBcalibration = 1000;
    double sum =0;
    double moyenne = 0;
    for (int i = 0; i < NBcalibration; i++) {
        sum += ReadVoltage(pin);
        delay(1); // Petite pause pour stabiliser les lectures
    }
    moyenne = sum/NBcalibration;
    return moyenne;
}

void callback(char* topic, byte* message, unsigned int length) {
    //Serial.print("Message arrived on topic: ");
    //Serial.print(topic);
    //Serial.print(". Message: ");
    String messageTemp;

    for (int i = 0; i < length; i++) {
        //Serial.print((char)message[i]);
        messageTemp += (char)message[i];
    }
}

```

```

StaticJsonDocument <256> doc;
deserializeJson(doc,message);

bool OCPINState = doc["OCPINState"];
bool BackFlowState = doc["BackFlowState"];
int PinSubState = doc["PinSubstate"];
bool CCProtState = doc["CCProtState"];
bool IR2184State = doc["IR2184State"];
bool CCPINState = doc["CCPINState"];
dutyCycle = doc["DutyCycleCTR"];
//Serial.print("valeur PIN : ");Serial.println(PinSubState);

if (OCPINState != digitalRead(OCPIN)){
    digitalWrite(OCPIN,!digitalRead(OCPIN));
    Serial.println("OCPIN toggle");
}
if (BackFlowState != digitalRead(backFlowPIN)){
    digitalWrite(backFlowPIN,!digitalRead(backFlowPIN));
    Serial.println("backFlowPIN toggle");
}
if (PinSubState != digitalRead(PinSub)){
    digitalWrite(PinSub,!digitalRead(PinSub));
    Serial.println("PinSub toggle");
}
if(CCPINState != digitalRead(CCPin)){
    digitalWrite(CCPin,!digitalRead(CCPin));
    Serial.println("CCPINState toggle");
}

if (CCProtState != digitalRead(CCProtPin)){
    digitalWrite(CCProtPin,!digitalRead(CCProtPin));
    Serial.println("CCProtPin toggle");
}
if (IR2184State != digitalRead(PINSD)){
    if(IR2184State ==1)
    {
        calibrationResultCourantIN = 1.65 - calibrationACS712(CourantINPIN);
        calibrationResultCourantOUT = 1.65 - calibrationACS712(CourantOutPin);
        Serial.print("Résultat calibration IN : ");Serial.println(calibrationResultCourantIN);
        Serial.print("Résultat calibration OUT : ");Serial.println(calibrationResultCourantOUT);
    }
}

```

```

        if(IR2184State ==0)
        {
            calibrationResultCourantIN = 0;
        }

        digitalWrite(PINSD,!digitalRead(PINSD));
        Serial.println("IR2184 toggle");
    }
}

void setup() {
    //Mise de la vitesse de transmission à 115200;
    Serial.begin(115200);
    // Connecting to a Wi-Fi network
    initWifi();
    //initiation Pin PWM
    ledcAttach(PWMPIN,freq, resolution);

    // Initiation Pin IR2184
    pinMode(PINSD, OUTPUT);
    digitalWrite(PINSD, LOW);

    //setup des pin de controle
    pinMode(backFlowPIN, OUTPUT);
    pinMode(OCPIN, OUTPUT);
    pinMode(PinSub, OUTPUT);
    pinMode(CCProtPin, OUTPUT);
    pinMode(CCPin,OUTPUT);
    digitalWrite(backFlowPIN,LOW);
    digitalWrite(OCPIN,LOW);
    digitalWrite(PinSub,LOW);
    digitalWrite(CCProtPin,LOW);
    digitalWrite(CCPin,LOW);

    //connexion au broker MQTT
    initMQTT();
    client.setCallback(callback);

    //adc.begin();
    Serial.println("Setup du esp32");
    delay(10);
}

```

```

}

void loop() {
    //Serial.println(ESP.getFreeHeap());
    client.loop();
    //Lecture Tension d'entrée
    //int TensionRAW = adc.readADC(TensionINPIN);

    double TensionRAW = ReadVoltage(TensionINPIN);
    double TensionMOY = MoyenneAnalogue(TensionTAB, sizeof(TensionTAB) /
sizeof(TensionTAB[0]), lectureIndex, TensionRAW); // filtre sur les 10 derniere data
    double TensionMAP = TensionMOY; //TensionMOY/4095*3.3; //mapping sur le 3.3volt
    TensionIN = (TensionMAP*(220000+5100)/5100);
    delay(10);

    //Lecture Tension Sortie
    double TensionSortie = ReadVoltage(TensionOUTPIN);
    double TensionOUTMOY = MoyenneAnalogue(TensionOUTTAB, sizeof(TensionOUTTAB) /
sizeof(TensionOUTTAB[0]), lectureIndex, TensionSortie);
    double TensionOUTMAP = TensionOUTMOY;
    TensionOut = (TensionOUTMAP*(23500+1000)/1000);
    delay(10);
    //Lecture Courant d'entrée
    double CourantRaw = ReadVoltage(CourantINPIN) + calibrationResultCourantIN;
    double CourantInMoy = MoyenneAnalogue(CourantINTAB, sizeof(CourantINTAB) /
sizeof(CourantINTAB[0]), lectureIndex, CourantRaw);
    //CourantInINter = CourantInMoy*(2177+6700+4960)/(2177+6700) + 0.015;
    //CourantIn = (CourantInINter-2.5)/0.185;
    CourantIn = 36.7*CourantInMoy/3.3-18.3;
    //4960 2177 6700 13837 8877
    // -0.34
    delay(10);
    //Lecture Courant de Sortie
    double courantOutRaw = ReadVoltage(CourantOutPin) + calibrationResultCourantOUT;
    double courantOUTMOY = MoyenneAnalogue(CourantOUTTAB, sizeof(CourantOUTTAB) /
sizeof(CourantOUTTAB[0]), lectureIndex, courantOutRaw);
    //double courantOUTter = courantOUTMOY * (4878+8960)/8960 + 0.03;
    //courantOUT = (courantOUTter-2.5)/0.185;
    courantOUT = 36.7*courantOUTMOY/3.3-18.3;
    //PWM pour le rapport cyclique du DC/DC
    ledcWrite(PWMPIN, dutyCycle);

    lectureIndex = lectureIndex + 1;
    if(lectureIndex >= numlecture) {
        lectureIndex = 0;
    }
}

```

```

}

//xTaskCreate(myCoolTask, "myCoolTask", 4096, NULL, 2, NULL);

PublishMQTT();

unsigned long currentMillis2 = millis();

if (currentMillis2 - previousMillis2 >= interval2) {
    previousMillis2 = currentMillis2;

    Serial.print("rapport : ");Serial.println(dutyCycle);
    Serial.print("TensionEntree : "); Serial.println(TensionIN);
    Serial.print("CourantIn : "); Serial.println(CourantIn);
    //Serial.print("CourantSortieRAW : "); Serial.println(courantOutRaw);
    //Serial.print("CourantsortieInter : "); Serial.println(courantOUTter);
    Serial.print("COURANT SORTIE : "); Serial.println(courantOUT);

}

unsigned long currentMillis = millis();
// if WiFi is down, try reconnecting every CHECK_WIFI_TIME seconds
if ((WiFi.status() != WL_CONNECTED) && (currentMillis - previousMillis
>=interval)) {
    Serial.print(millis());
    Serial.println("Reconnecting to WiFi...");
    WiFi.disconnect();
    WiFi.reconnect();
    previousMillis = currentMillis;
}

}

```

BIBLIOGRAPHIE

1. UR REHMAN, H., YAN, X., ABDELBAKY, M. A., ULLAH JAN, M. & IQBAL, S. 2021. An advanced virtual synchronous generator control technique for frequency regulation of grid-connected PV system. *International Journal of Electrical Power and Energy Systems*, 125.
2. **Mingxuan Mao**, Lichuang Cui, Qianjin Zhang, Ke Guo, Lin Zhou, Han Huang, Classification and summarization of solar photovoltaic MPPT techniques: A review based on traditional and intelligent control strategies, *Energy Reports*, Volume 6, 2020, Pages 1312-1327.
3. R. Boukenoui, H. Salhi, R. Bradai, A. Mellit, A new intelligent MPPT method for stand-alone photovoltaic systems operating under fast transient variations of shading patterns, *Solar Energy*, Volume 124, 2016, Pages 124-142
4. Elyaqouti, M., et al., Implementation in arduino of MPPT using variable stepsize P&O algorithm in PV installations. 2017. 8: p. 434-443
5. Podder, Amit & Roy, Naruttam & Pota, Hemanshu. (2019). MPPT Methods for Solar PV Systems: A Critical Review Based on Tracking Nature. *IET Renewable Power Generation*. 13. 10.1049/iet-rpg.2018.5946.
6. Baimel, D.; Tapuchi, S.; Levron, Y.; Belikov, J. Improved Fractional Open Circuit Voltage MPPT Methods for PV Systems. *Electronics* **2019**, 8, 321.
<https://doi.org/10.3390/electronics8030321>
7. Mingxuan Mao, Lin Zhou, Zengrui Yang, Qianjin Zhang, Chen Zheng, Bao Xie, Yihao Wan, A hybrid intelligent GMPPT algorithm for partial shading PV system, *Control Engineering Practice*, Volume 83, 2019, Pages 108-115
8. **Faranda, Roberto** & Leva, S. & Maugeri, V.. (2008). MPPT techniques for PV Systems: Energetic and cost comparison. 1 - 6. 10.1109/PES.2008.4596156.
9. S. Sumathi, L.A.K.a.P.S., *Solar PV and Wind Energy Conversion Systems*. Green Energy and Technology, ed. Springer. 2015. 807

10. JAMIL, M., RIZWAN, M. & KOTHARI, D. P. 2017. *Grid integration of solar photovoltaic systems*.
11. NANDA, S. & SUDHAKAR, P. Design of integrated single stage photovoltaic solar inverter(stand-alone). IOP Conference Series: Materials Science and Engineering, 2020.
12. Chitransh, Apar & Kumar, Sachin. (2021). The Different Type of MPPT Techniques for Photovoltaic System. Indian Journal of Engineering and Materials Sciences. 1. 4. 10.35940/ijee.A1809.111221.
13. HEGARTY, A., WESTBROOK, G., GLYNN, D., MURRAY, D., OMERDIC, E. & TOAL, D. A Low-cost remote solar energy monitoring system for a buoyed iot ocean observation platform. IEEE 5th World Forum on Internet of Things, WF-IoT 2019 - Conference Proceedings, 2019. 386-391
14. Hadji, S.; Gaubert, J.-P.; Krim, F. Real-Time Genetic Algorithms-Based MPPT: Study and Comparison (Theoretical an Experimental) with Conventional Methods. *Energies* **2018**, *11*, 459
15. Emeter, Moses & Elumole, Oluwatoyin. (2019). Optimization of solar photovoltaic efficiency using angular inclination: Effect of dust deposition on PV Module. IOP Conference Series: Earth and Environmental Science. 331. 012061.
16. Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press
17. Zecevic, Zarko & Rolevski, Maja. (2020). Neural Network Approach to MPPT Control and Irradiance Estimation. Applied Sciences. 10. 5051.
18. MAO, M., CUI, L., ZHANG, Q., GUO, K., ZHOU, L. & HUANG, H. 2020. Classification and summarization of solar photovoltaic MPPT techniques: A review based on traditional and intelligent control strategies. *Energy Reports*, 6, 1312-1327.
19. SIMÕES, M. G. & FARRET, F. A. 2016. *Modeling power electronics and interfacing energy conversion system*

20. Akbar, Moh & Efendi, Moh & Nugraha, Syechu. (2022). SEPIC Converter for Lead Acid Battery Charger Using Fuzzy Logic type-2 Controller
21. Korich, Belkacem. (2022). Contribution à l'amélioration des performances d'un système d'énergie renouvelable.
22. SAAD, I., TANGOUR, F. & BORNE, P. 2009. Application des algorithmes génétiques aux problèmes d'optimisation. *Revue de l'Electricité et de l'Electronique*, pp. 33-38.
23. KUMARI, P. A. & GEETHANJALI, P. 2017. Adaptive Genetic Algorithm Based Multi-Objective Optimization for Photovoltaic Cell Design Parameter Extraction. *Energy Procedia*, 117, 432-441.
24. MIRZA, A. F., MANSOOR, M., LING, Q., KHAN, M. I. & ALDOSSARY, O. M. 2020. Advanced variable step size incremental conductance mppt for a standalone PV system utilizing a ga-tuned pid controller. *Energies*, 13
25. T. Tafticht, M. Tchakala, Md. J. Rahman, "GMPPT approach for photovoltaic systems under partial shading conditions using a genetic algorithm", International Journal of Power Electronics and Drive Systems (IJPEDS), Vol. 13, No. 2, June 2022, ISSN: 2088-8694, DOI: 10.11591/ijpeds.v13.i2.ppab-cd
26. SAMEEULLAH, M. & SWARUP, A. 2016. MPPT schemes for PV system under normal and partial shading condition: A review. *International Journal of Renewable Energy Development*, 5, 79-94
27. Danoy, Grégoire. (2008). A Multi-Agent Approach for Hybrid and Dynamic Coevolutionary Genetic Algorithms: Organizational Model and Real-World Problems Applications
28. ALILOUCH, H. 2020. *Conception et analyse des performances d'un contrôleur flou pour un système de réglage de tension automatique*. Mémoire de maîtrise, Université du Québec en Abitibi-Témiscamingue

29. Kamath, H. & Goswami, A. & Aditya, Kumar & Aithal, R & Singh, P.. (2011). RBF and BPNN combi model based KALMAN filter application for maximum power point tracker of PV cell. Lecture Notes in Engineering and Computer Science.