

وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR- ANNABA UNIVERSITY
UNIVERSITE BADJI MOKHTAR ANNABA



جامعة باجي مختار - عنابة

Année : 2019

Faculté: Sciences de l'Ingénierat
Département: Electronique

MEMOIRE

Présenté en vue de l'obtention du diplôme de : MASTER

Intitulé :
Méthodes d'optimisation pour l'auto-réglage de
contrôleur PID d'ordre fractionnaire

Domaine : Sciences et Technologie

Filière : Electronique

Spécialité: Automatique et Système

Par :
ATTOUI Fateh

DEVANT Le JURY

Président : Pr. R.Lakel	Grade	Professeur UBM Annaba
Directeur de mémoire: Dr. T. Ait Izem	Grade	M.A.B UBM Annaba
Examineurs: Dr. S Benmoussa	Grade	M.C.A UBM Annaba
Dr. K. Beddoud	Grade	M.C.A UBM Annaba

Résumé

L'optimisation par Essaims particulaires (PSO), et l'algorithme génétique (GA) sont des algorithmes évolutionnaires qui utilisent des individus pour développer une solution optimale adaptée à une multitude de problèmes. Nous utiliserons ces méthodes dans ce mémoire pour déterminer les paramètres optimaux du correcteur PID et PID fractionnaire, à savoir λ , μ , K_p , K_i et K_d , qui minimisent une fonction objective définie en terme de l'erreur de régulation. Une comparaison entre les performances obtenus par la correction PID et $PI^\lambda D^\mu$ fractionnaire nous démontre la supériorité de ce dernier. Une validation des algorithmes (GA/PSO) pour la détermination des paramètres du PID/ $PI^\lambda D^\mu$, ainsi que les performances du contrôleur fractionnaire sont illustrés par deux applications : un asservissement de niveau et un moteur a courant continu.

Mot clé : Optimisation par l'Essaims Particulaires (PSO), Algorithme génétique (GA), $PI^\lambda D^\mu$ fractionnaire, régulateur

Abstarct:

Particle Swarm Optimization (PSO), and the Genetic Algorithm (GA) are evolutionary algorithms that use individuals to develop an optimal solution for a multitude of problems. We will use these methods in this memory to determine the optimal parameters of the fractional PID and PID corrector, namely λ , μ , K_p , K_i and K_d , which minimize an objective function defined in terms of the regulation error. A comparison between the performances obtained by the PID correction and Fractional $PI^\lambda D^\mu$ shows us the superiority of the latter. Validation of the algorithms (GA / PSO) for the determination of the parameters of the PID- $PI^\lambda D^\mu$, as well as the performances of the fractional controller are illustrated by two applications: a servo-level control and a DC motor.

Keyword: Optimization by Particle Swarms (PSO), Genetic Algorithm (GA), Fractional $PI^\lambda D^\mu$, Regulator

الملخص :

يعد تحسين سرب الجسيمات (PSO) ، والخوارزمية الجينية (GA) خوارزميات تطويرية تستخدم الأفراد لتطوير حل مثالي للعديد من المشكلات. سنستخدم هذه الطرق في هذه الذاكرة لتحديد المعلمات المثلى لمصحح PID و PID الكسري ، أي إيجاد المعاملات λ و μ و K_p و K_i و K_d ، مما يقلل من وظيفة موضوعية محددة من حيث خطأ التنظيم. توضح المقارنة بين العروض التي حصلنا عليها كلا من المصحح PID و $PI^{\lambda}D^{\mu}$ الكسري على تفوق الأخير. يوضح التحقق من صحة الخوارزميات (GA / PSO) لتحديد المعاملات $PI^{\lambda}D^{\mu}$ / PID ، بالإضافة إلى أداء وحدة التحكم الكسري من خلال تطبيقين: تحكم على مستوى الماكينة ومحرك DC

الكلمة الرئيسية: التحسين بواسطة أسراب الجسيمات (PSO) ، الخوارزمية الجينية (GA) ، منظم الكسور $PI^{\lambda}D^{\mu}$ ،

Dédicaces

*A mes parents,
A tout ma famille,
A tous mes amis,*

Remerciement

*Le plus grand merci revient tout d'abord à « Dieu » qui, lui seul, nous a guidé dans le bon chemin durant notre vie. Nous remercions nos parents qui nous ont amenés à cet endroit avancée. Nous tenons aussi à remercier tous nos sœurs, Nous offrons tous les remerciements et l'appréciation et la gratitude à notre encadreur **Mr. TAREK ,AIT IZEM** qui nous a donné toute l'aide et les conseils et tous les explication et le temps nécessaire pour accomplir ce travail avec succès sans oublié Mr NADIR FARGUENI*

*Nous tenons aussi à remercier tous nos amis
et collègues de travail*

Liste des tableaux

CHAPITRE III :

Tableau (III. 1) Performances obtenues pour l'asservissement de niveau par le critère ISE .	28
Tableau (III. 2) Performances obtenues pour l'asservissement de niveau par le critère IAE .	29
Tableau (III. 3) Performances obtenues pour l'asservissement de niveau part le critère ISE.(PSO).....	30
Tableau (III. 4) Performances obtenues pour l'asservissement de niveau part le critère IAE.	31
Tableau (III. 5) Valeurs numériques d'un moteur à courant continu	33
Tableau (III. 6) Performances obtenues pour La vitesse de moteur part le critère ISE – GA.	36
Tableau (III. 7) Performances obtenues pour La vitesse de moteur part le critère IAE GA..	37
Tableau (III. 8) Performances obtenues pour moteur part le critère ISE -PSO	38
Tableau (III. 9) Performances obtenues pour vitesse de moteur part le critère ISE-PSO-	39

Liste des figures

CHAPITRE I :

Figure (I.1) Schéma fonctionnel d'une boucle de régulation	4
Figure (I.2) Structure parallèle d'un correcteur PID	5
Figure (I.3) Structure série du correcteur PID	5
Figure (I.4) Tracé de Bode de l'intégrateur fractionnaire $1/s^{0.62}$ et son approximée.....	11
Figure (I.5) Structure du $PI^\lambda D^\mu$ d'ordre fractionnaire.....	12
Figure (I.6) Types de correcteurs en fonction de λ et μ	13
Figure (I.7) Variation de λ du correcteur $PI^\lambda D^\mu$ Fractionnaire.....	14
Figure (I.8) Variation de μ du correcteur $PI^\lambda D^\mu$ Fractionnaire	14
Figure (I.9) Tracé de Bode de la fonction de transfert du $PI^{1.7}D^{1.7}$ et son approximation	15

CHAPITRE II :

Figure (III. 1) Asservissement de niveau.....	26
Figure (III. 2) Réponse indicielle du système.....	27
Figure (III. 3) Réponse indicielle de l'asservissement de niveau avec correction PID et $PI^\lambda D^\mu$ (critère ISE).....	28
Figure (III. 4) Réponse indicielle de l'asservissement de niveau avec correction PID et $PI^\lambda D^\mu$ par critère IAE-GA	29
Figure (III. 5) Réponses indicielles de système $G1(s)$ après correction de $PI^\lambda D^\mu$ et PID classique. Critère ISE-PSO-	30
Figure (III. 6) Réponses indicielles de système $G1$ après correction de $PI^\lambda D^\mu$ et PID classique. Critère IAE-PSO	31
Figure (III. 7) les réponses de correcteur fractionnaire avec les résultats de référence.....	32
Figure (III. 8) Le moteur à courant continu.....	33
Figure (III. 9) Schéma bloc du Modèle de moteur électrique en vitesse	34
Figure (III. 10) Réponse indicielle du système.....	34
Figure (III. 11) Réponse indicielle de vitesse du moteur avec correction PID et $PI^\lambda D^\mu$	35
Figure (III. 12) Réponses indicielles de système vitesse du moteur après correction de $PI^\lambda D^\mu$ et PID classique. GA-IAE	36

Figure (III. 13) Réponses indicielles de système vitesse de moteur après correction de $PI\lambda D\mu$ et PID classique. PSO-ISE	38
Figure (III. 14) Réponses indicielles de vitesse de moteur PSO-IAE.....	39

Sommaire

CHAPITRE I :

Introduction	4
I.1. Le Régulateur PID	4
I.2. Structure d'un PID:	5
I.2.a. Structure parallèle:	5
I.2.b. La structure série	5
I.3. Réglage des paramètres d'un PID:	6
I.4. Le Calcul Fractionnaire.....	6
I.4.1. Introduction au calcul fractionnaire :	6
I.4.1.a. Intégrale de Riemann-Liouville :	7
I.4.1.b. Dérivée de Riemann-Liouville :	7
I.4.1.c. Quelques propriétés de l'opérateur fractionnaire :	7
I.4.2. Approximation analogique des opérateurs fractionnaires	8
I.4.2.a. Méthode d'Oustaloup	8
I.4.2.b. Méthode de Charef.....	9
I.5. Le Correcteur $PI\lambda D^\mu$ Fractionnaire	11
I.5.1. Structure du correcteur $PI^\lambda D^\mu$ fractionnaire	12
I.5.2. Approximation d'un $PI^\lambda D^\mu$ fractionnaire par une fonction rationnelle.....	14
Conclusion	16

CHAPITRE II :

Introduction	17
II.1. Optimisation par Essaim Particulaire(PSO):	17
II.2. Principe de fonctionnement de PSO:	17
II.3. Mécanisme de confinement	19
II.4. Optimisation par Algorithme génétique.....	21

II.4.1. Présentation del’algorithmme génétique (GA).....	21
II.4.2. Les Opérateurs :.....	22
II.4.2.a. L’opérateur de sélection :	22
II.4.2.b. L’opérateur de Croisement :	22
II.4.2.c.l’opérateur de mutation[5] :	23
II.5. Ajustement des Paramètres d’un régulateur PID par optimisation.....	4
II.5.1. Critère de performances	25
II.5.1.a.L’intégrale du carré de l’erreur (ISE):	25
II.5.1.b. L’intégrale de l’erreur absolue (IAE)	25
Conclusion.....	25

CHAPITRE III :

Introduction	26
III.1.Asservissement de niveau.....	26
III.1.1.Optimisation par GA	27
III.1.1.a. Critère ISE :	28
III.1.1.b. Critère IAE	28
III.2. Optimisation par PSO.....	29
III.2.a. Critère ISE	30
III.2.2.b. Critère IAE.....	31
III.3. Commande d’un moteur à courant continu.....	33
III.3.1. Régulation en vitesse	35
III.3.2. Réglage par GA:.....	35
III.3.2.a. Critère ISE	35
III.3.2.b. Critère IAE	36
III.3.3. Réglage par PSO	37
III.3.3.a. Critère ISE	37

Introduction générale

Le correcteur PID est la technique la plus utilisée dans l'asservissement des processus industriels pour des décennies. La conception et le réglage du correcteur PID a été un sujet de recherche depuis le jour où Ziegler et Nichols 1942 ont présenté leur méthode de réglage. Ces dernières décennies ont vu l'apparition de plusieurs autres approches de réglage des paramètres du PID se basant sur des méthodes d'optimisations, l'objectif étant d'obtenir une réponse adéquate du procédé et de la régulation. Bien qu'il y a plusieurs techniques existantes pour le réglage des paramètres du correcteur PID, un travail intensif de recherche continu encore pour le rehaussement de la qualité et l'amélioration des performances de la commande. En fait, et au début des années 1990, le régulateur CRONE a été proposé par Oustaloup [1]. Ce correcteur profite des propriétés avantageuses des systèmes d'ordre fractionnaire qui ont vu le jour grâce au calcul fractionnaire, ce régulateur permettait d'assurer la robustesse de la commande dans une bande de fréquences donnée. Depuis cette initiative, la commande d'ordre fractionnaire captiva l'intérêt de beaucoup de chercheurs. En 1999, Podlubny [2] qui a proposé le régulateur $PI^\lambda D^\mu$, qui une généralisation du correcteur PID classique, comprenant une intégration fractionnaire d'ordre λ et une dérivation fractionnaire d'ordre μ , cela qui va permettre de élargie le champ d'application du calcul fractionnaire à la théorie de la commande, ce qui a orienté plusieurs chercheurs à un nouveau axe de recherche qui est le réglage du correcteur PID d'ordre fractionnaire [3-4].

En effet, Plusieurs approches récentes ont été documentées dans la littérature pour déterminer les paramètres PID classique ont été généralement satisfaisantes dans la commande des systèmes linéaires.

Notre intérêt dans ce mémoire porte sur l'adaptation des méthodes d'optimisations intelligentes pour le réglage du contrôleur $PI^\lambda D^\mu$ fractionnaire ainsi que le PID classique, avec comme objectif d'avoir une réponse optimale du système contrôlé. Les méthodes d'optimisation que nous utiliserons sont les algorithmes génétiques (GA) [5] [6] et l'optimisation par essaim particulaire (PSO) [8]. A partir d'une population initiale (individus), et en se référant à un critère de cout défini en fonction de l'erreur, ces algorithmes nous permettrons de développer une solution optimale de notre problème, synonymes d'une erreur minimale de régulation, et donc de paramètres optimaux du contrôleur.

Ce mémoire sera organisé en trois chapitres structurés comme suit :

Après l'introduction générale, un premier chapitre contiendra une introduction aux éléments de base du calcul d'ordre fractionnaire avec quelques notions essentielles sur celui-ci, nous citerons les méthodes d'approximation rationnelles existantes dans la littérature. Par la suite, ces approximations seront utilisées pour la mise en œuvre du nouveau correcteur $PI^{\lambda}D^{\mu}$ fractionnaire, qui fera l'objectif d'étude, notamment, pour la détermination de ses paramètres optimaux.

Le deuxième chapitre portera sur les méthodes d'optimisations. Où nous introduirons deux approches : GA et le PSO. Le principe de déroulement de ces algorithmes sera expliqué, et seront utilisé par la suite pour le calcul des paramètres du correcteur PID et $PI^{\lambda}D^{\mu}$ fractionnaire.

Le troisième chapitre sera dédié à l'application des algorithmes d'optimisation présentés, pour le calcul optimal des paramètres d'un correcteur dont la tâche est d'asservir un système donné, dans ce mémoire, un asservissement de niveau et un moteur à courant continu, feront l'objet de contrôle. Une discussion des résultats obtenus révélera les approches les plus performantes en termes des caractéristiques temporelles du système.

Nous finirons ce mémoire par une conclusion et quelques perspectives.

Chapitre I

Correcteur PID et $PI^\lambda D^\mu$ fractionnaire

Introduction

Le contrôleur PID est la forme la plus commune de rétroaction. C'était l'outil standard quand le contrôle des processus a émergé dans les années 40. Il est basé sur un mécanisme de rétroaction, largement répandu dans les systèmes de commande industriel ; grâce notamment à sa simple structure et sa stratégie de commande.

Bien qu'il existe plusieurs techniques pour le réglage des paramètres du correcteur PID, un travail de recherche continu et intensif est encore en cours pour le rehaussement de la qualité et l'amélioration des performances de la commande. En revanche, une nouvelle naissance d'un régulateur plus généralisé que la version classique du PID a été introduite, en se basant sur le calcul fractionnaire, pour permettre d'avoir deux degrés de liberté de plus. A savoir, les ordres d'intégration et de dérivée fractionnaires i.e. non entiers. Ces paramètres peuvent être utilisés pour satisfaire des performances additionnelles dans la conception des systèmes asservis.

I.1. Le Régulateur PID

Le calcul du contrôleur PID (algorithme) implique trois paramètres : le proportionnel, les deux valeurs intégrales et dérivées respectivement dénoté I et D. Ces valeurs peuvent être interprétées en termes d'erreur ; P dépend de l'erreur actuelle, I sur l'accumulation des erreurs passées, et D est une prévision de futures erreurs.

La somme pondérée de ces trois actions est employée pour ajuster le processus et En accordant les trois paramètres dans l'algorithme du contrôleur ainsi que des conditions spécifiques, le Contrôleur peut fournir une action de commande qui répond à notre attente.

Le régulateur, dont la fonction de transfert est désignée par $C(s)$ est situé en amont du système à régler $H(s)$ et l'entrée du régulateur comprend forcément la consigne $r(s)$. Ainsi la mesure $y(s)$ de la grandeur réglée est comparé à chaque instant par la référence (consigne) $r(s)$ qui nous donne un paramètre important appelée écart ou erreur $e(s) = r(s) - y(s)$, comme démontré dans la figure (I.1)

Le rôle de ce régulateur est de maintenir le signal d'erreur $e(s)$ aussi proche de zéro que Possible dont le but il fournit au système à régler la commande $u(s)$ telle que cette grandeur est Appliquée au système à régler $H(s)$ et qui provoque donc une modification de la grandeur réglée $y(t)$. Cependant la puissance se varier le réglage par l'intermédiaire d'un actionneur (organe de réglage) [8].

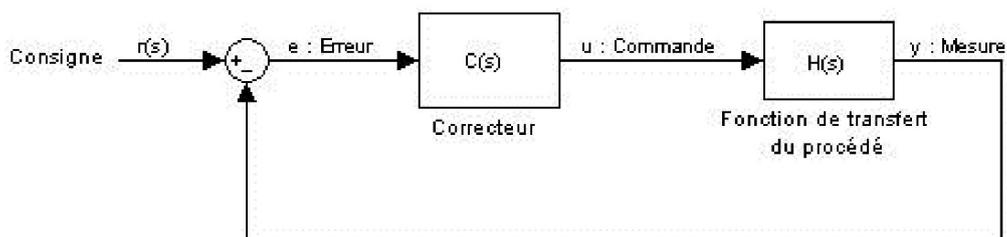


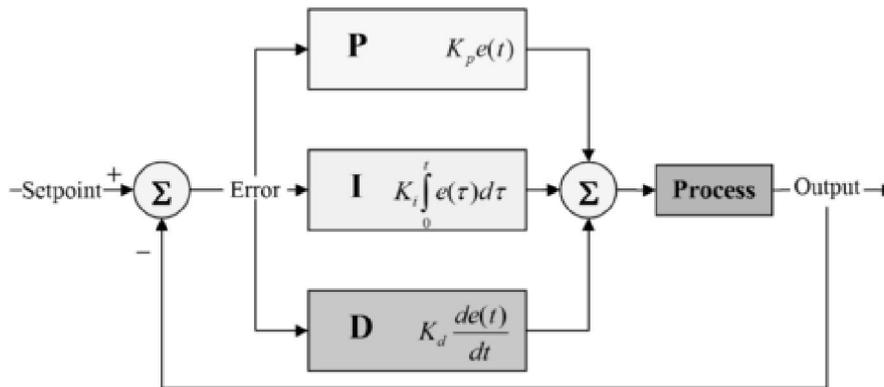
Figure (I.1) Schéma fonctionnel d'une boucle de régulation

I.2. Structure d'un PID:

Les actions élémentaires d'un régulateur peuvent être associées de plusieurs façons. On parle de la structure d'algorithme du régulateur. Les valeurs des paramètres PID ne donnent pas le même comportement du procédé selon que la structure est parallèle ou mixte [9][10]. C'est pourquoi il est primordial que l'automaticien connaisse les structures existantes. Donc il y a plusieurs architectures possibles pour combiner les trois effets (série, parallèle ou mixte). Les deux premières formes étant les plus utilisés.

I.2.a. Structure parallèle:

Cette structure est illustrée par la figure (I.2).



La fonct

I.2.b.La

La struct

$$C_{serie}(s) = k(1 + 1/(s\tau_i))(1 + s\tau_d) \tag{I.2}$$

Ou bien

$$C_{serie}(s) = \frac{(1+\tau_1s)(1+\tau_2s)}{\tau_i(1+\tau_Ns)} \tag{I.3}$$

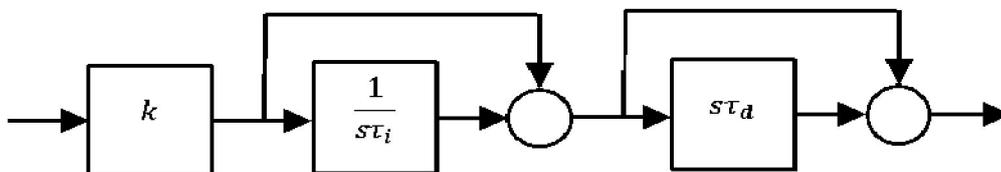


Figure (I.3) Structure série du correcteur PID

I.3. Réglage des paramètres d'un PID:

La conception et le réglage du correcteur PID a été un sujet de recherche depuis le jour où Ziegler et Nichols 1942 ont présenté leur méthode de réglage. L'utilisation des méthodes d'optimisations est devenue de plus en plus utilisée, permettant ainsi la détermination des coefficients K_p , K_d et K_i afin d'obtenir une réponse adéquate du procédé et de la régulation.

Nous nous intéressons dans ce mémoire à un nouveau type de correcteur, notée $PI^\lambda D^\mu$ d'ordre fractionnaire, en se basant sur des techniques d'optimisations pour le réglage et la détermination de ses cinq paramètres K_p , K_i , K_d , ainsi que les ordres de d'intégration et de dérivation fractionnaires λ et μ .

I.4. Le Calcul Fractionnaire

Le calcul fractionnaire est le champ de l'analyse mathématique, l'investigation et l'application des intégrales et des dérivées d'ordre non-entiers. Le calcul fractionnaire peut être considéré comme un sujet ancien et encore nouveau. Ces dernières années l'intérêt considérable pour le calcul fractionnaire a été stimulé par les applications de ce calcul dans les différents domaines de la physique et de l'ingénierie [1][3]. Notamment en contrôle, où son utilité se manifeste dans l'introduction du nouveau correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire.

L'objectif de ce chapitre est de présenter les bases théoriques des opérateurs d'ordre fractionnaire, tout en rappelant les définitions et les principales propriétés de ces opérateurs, ainsi que la présentation du correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire.

I.4.1. Introduction au calcul fractionnaire :

Le calcul fractionnaire est une généralisation de l'intégration et de la différentiation à l'opérateur fondamental d'ordre non entier ${}_{t_0}D_t^m$ où t_0 et t sont des limites de l'opération. L'opérateur intégro-différentiel continu est défini par [1][2] :

$${}_{t_0}D_t^m = \begin{cases} \frac{d^m}{dt^m} & R(m) > 0 \\ 1 & R(m) = 0 \\ \int_{t_0}^t (d\tau)^{-m} & R(m) < 0 \end{cases} \quad (I.4)$$

Où $m \in \mathbb{C}$ est l'ordre de l'opération, est l'anneau des nombres complexes et $R(m)$ symbolise la partie réelle d'un nombre complexe

Il existe plusieurs définitions mathématiques pour l'intégration et la dérivation d'ordre fractionnaire. Ces définitions ne mènent pas toujours à des résultats identiques mais sont équivalentes pour un large panel de fonctions [2]. Nous citerons la définition de Riemann-Liouville [2] qui est la définition la plus utilisée.

I.4.1.a. Intégrale de Riemann-Liouville :

Soient C et R les anneaux des nombres complexes et réels respectivement $R(m)$ symbolise la partie réelle d'un nombre complexe. Soient $\alpha \in C$ avec $R(\alpha) > 0, t_0 \in R$ et f une fonction localement intégrable définie sur $[t_0, +\infty[$. L'intégrale de Riemann-Liouville d'ordre m de f de borne inférieure t_0 est définie par :

$${}^{RL}I_{t_0}^m f(t) \equiv \frac{1}{\Gamma(m)} \int_{t_0}^t (t-\tau)^{m-1} f(\tau) d\tau \quad (I.5)$$

Avec $t > t_0$ et $\Gamma(m)$ est la fonction gamma d'Euler définie par :

$$\Gamma(x) = \int_0^{\infty} y^{x-1} e^{-y} dy \quad (I.6)$$

I.4.1.b. Dérivée de Riemann-Liouville :

Soient C et R les anneaux des nombres complexes et réels respectivement, $R(.)$ symbolise la partie réelle d'un nombre complexe. Soient $\alpha \in C$ avec $R(\alpha) > 0, t_0 \in R$ et f une fonction localement intégrable définie sur $[t_0, +\infty[$. La dérivée de Riemann-Liouville d'ordre m de f de borne inférieure t_0 est définie par :

$${}^{RL}D_{t_0}^m f(t) \equiv \frac{1}{\Gamma(n-m)} \frac{d^n}{dt^n} \int_{t_0}^t (t-\tau)^{n-m-1} f(\tau) d\tau \quad (I.7)$$

Où le nombre entier n est tel que $(n-1) < m < n$.

I.4.1.c. Quelques propriétés de l'opérateur fractionnaire :

Les principales propriétés des opérateurs d'ordre fractionnaire sont les suivantes:

1. Si $f(t)$ est une fonction analytique de t , alors, sa dérivée d'ordre fractionnaire $D^m f(t)$ est une fonction analytique de t et m .
2. Pour $n=m$, où n est un entier, l'opération $D^m f(t)$ donne le même résultat que la différentiation classique d'ordre entier n .
3. Pour $m=0$ l'opération $D^m f(t)$ est l'opérateur identité: $D^0 f(t) = f(t)$.
4. La différentiation et l'intégration d'ordres fractionnaire sont des opérations linéaires $D^m a f(t) + D^m b g(t) = a D^m f(t) + b D^m g(t)$.
5. La loi additive (propriété du semi-groupe) $D^m D^n f(t) = D^{m+n} f(t)$ est valable, sous certaines contraintes sur la fonction $f(t)$.
6. la dérivée non entière de $f(t)$ nécessite la connaissance de $f(t)$ sur l'intervalle $[t_0, t]$ alors que dans le cas entier, seule la connaissance "locale" autour de t est nécessaire. Cette propriété permet d'interpréter les systèmes non entiers comme des systèmes à mémoire longue et les systèmes entiers comme des systèmes à mémoire courte.

I.4.2. Approximation analogique des opérateurs fractionnaires

Il est très important de trouver des approximations d'ordre entier pour des fonctions de transfert d'ordre fractionnaire. Autrement dit, lorsque les simulations doivent être exécutées ou les correcteurs doivent être implémentés, les fonctions de transfert d'ordre fractionnaire sont remplacées par des fonctions de transfert d'ordre entier, avec un comportement assez identique à celles désirées, mais beaucoup plus facile à manipuler. Notons que les approximations disponibles dans le domaine fréquentiel « s » sont appelées : Approximations analogiques.

Il existe plusieurs méthodes pour trouver de telles approximations .Dans la référence [11][4], on peut trouver un très bon résumé des méthodes d'approximation analogique des systèmes d'ordre fractionnaire existantes dans la littérature, dont nous pouvons citer :

- Méthode de Matsuda ;
- Méthode d'Oustaloup ;
- Méthode de Charef ;
- Autres méthodes ((Expansion Fractionnaire Continue, Roy,Wang, Jones...).

Le choix d'une méthode dépend de la structure du système que l'on cherche à simuler. Nous choisirons, bien évidemment, la mieux adaptée au problème. Dans ce mémoire, nous présenterons les méthodes d'Oustaloup et de Charef. qui sont les plus utilisées.

I.4.2.a. Méthode d'Oustaloup

La méthode d'Oustaloup [1], repose sur l'approximation en temps continu de l'opérateur d'ordre fractionnaire $G(s) = s^\alpha, \alpha \in \mathfrak{R}$, par une fonction rationnelle en utilisant une distribution récursive de zéros et pôles d'ordre entier, répartis dans une bande de fréquence limitée. Ainsi, l'approximation de l'opérateur s^α dans une bande de fréquence est donnée par :

$$G(s) = s^\alpha \approx \hat{G}(s) = C_0 \prod_{i=1}^N \left(\frac{1 + \frac{s}{z_i}}{1 + \frac{s}{p_i}} \right) \quad (I.8)$$

Avec :

$$si \alpha > 0 : \begin{cases} z_1 = w_b \sqrt{\eta} \\ p_i = z_i \lambda, & i = 1, 2, \dots, N \\ z_{i+1} = p_i \eta, & i = 1, 2, \dots, N-1 \\ w_h = p_N \sqrt{\eta} \end{cases} \quad si \alpha < 0 : \begin{cases} p_1 = w_b \sqrt{\eta} \\ z_i = p_i \lambda, & i = 1, 2, \dots, N \\ p_{i+1} = z_i \eta, & i = 1, 2, \dots, N-1 \\ w_h = z_N \sqrt{\eta} \end{cases} \quad (I.9)$$

$$N = \text{Partie entière de } \left\{ \frac{\log \{w_h / w_b\}}{\log \{\alpha \eta\}} \right\} \quad (I.10)$$

$$\alpha = \left(\frac{w_h}{w_b} \right)^{|\lambda|/N}, \quad \eta = \left(\frac{w_h}{w_b} \right)^{(1-|\alpha|)/N} \quad (I.11)$$

Le modèle rationnel est obtenu ensuite en remplaçant chaque opérateur d'ordre fractionnaire du modèle original par son approximation rationnelle. La sortie du modèle est donc la convolution du modèle rationnel obtenu avec le signal d'entrée.

I.4.2.b. Méthode de Charef

- Approximation d'un intégrateur fractionnaire [11]

La fonction de transfert rationnelle de l'opérateur intégral d'ordre fractionnaire

$H(s) = 1/s^m$ est calculée dans une bande de fréquence donnée $[w_b, w_h]$. La méthode d'approximation de Charef consiste à approximer la pente de 20dB/dec sur le tracé de Bode de l'opérateur fractionnaire par un nombre de lignes sous forme de Zig-Zag, produite par une alternance de pente 20 dB/dec et 0 dB/dec correspondant à une alternance de pôles et de zéros sur l'axe réel négatif du plan s tel que : $p_0 < z_0 < p_1 < z_1 < \dots < z_{n-1} < p_n$. D'où l'approximation suivante :

$$H_I(s) = \frac{K_I}{\left(1 + \frac{s}{w_c}\right)^m} \cong K_I \frac{\prod_{i=0}^{n-1} \left(1 + \frac{s}{z_i}\right)}{\prod_{i=0}^n \left(1 + \frac{s}{p_i}\right)} \quad (I.12)$$

Avec $K_I = (1/w_c^m)$, et w_c la fréquence de coupure donnée par $w_c = w_b \sqrt{10^{\left(\frac{\varepsilon}{10m}\right)} - 1}$. ε est l'erreur d'approximation maximale permise dans la bande de fréquence donnée $[w_b, w_h]$. Les p_i et z_i sont les pôles et les zéros de l'approximation.

On note aussi que :

$$n = \text{Partie entière de} \left[\frac{\log\left(\frac{w_{\max}}{p_0}\right)}{\log(ab)} \right] + 1 \quad (I.13)$$

$$p_i = (ab)^i p_0 \text{ pour } i = 0, 1, \dots, n \quad z_i = (ab)^i a p_0 \text{ pour } i = 0, 1, \dots, n-1 \quad (I.14)$$

$$a = 10^{\left(\frac{y}{10(1-m)}\right)}, b = 10^{\left(\frac{y}{10m}\right)} \quad (I.15)$$

$w_{max} = 100.w_h$, p_0 et z_0 sont le premier pôle et le premier zéro, respectivement donnés par:

$$p_0 = w_c \sqrt{b}, \quad z_0 = ap_0 \quad (I.16)$$

- Approximation d'un dérivateur fractionnaire

La fonction de transfert de l'opérateur intégral d'ordre fractionnaire est représentée dans le domaine fréquentiel par la fonction irrationnelle $G_D(s) = s^m$. L'approximation de Charef est donnée donc par la relation suivante :

$$G_D(s) = K_D \left(1 + \frac{s}{w_c} \right)^m \cong K_D \frac{\prod_{i=0}^n \left(1 + \frac{s}{z_i} \right)}{\prod_{i=0}^n \left(1 + \frac{s}{p_i} \right)} \quad (I.17)$$

Avec :

$$n = \text{Partie entière de} \left[\frac{\log\left(\frac{w_{max}}{z_0}\right)}{\log(ab)} \right] + 1 \quad (I.18)$$

L'arrangement des pôles-zéros est établi selon les deux progressions géométriques suivantes :

$$z_i = (ab)^i z_0 \text{ pour } i = 0, 1, \dots, n \quad p_i = (ab)^i a z_0 \text{ pour } i = 0, 1, \dots, n \quad (I.19)$$

$$\text{avec } : z_0 = w_c \sqrt{b} \text{ et } p_0 = a.z_0$$

Exemple illustratif:

Considérons l'intégrateur d'ordre 0.62, on veut approximer sur la bande $[10^{-2}, 10^2]$ et pour ($\gamma=1\text{dB}$, $\epsilon=10^{-5}$). L'approximation par la méthode de Charef est donnée par l'équation suivante :

$$H(s) = \frac{1}{s^{0.62}} \cong 838.2128 G_I(s) \frac{\prod_{i=0}^{20} \left(1 + \frac{s}{4.253 * 10^{-5} (2.6573)^i} \right)}{\prod_{i=0}^{21} \left(1 + \frac{s}{2.3504 * 10^{-5} (2.6573)^i} \right)} \quad (I.20)$$

La figure (I.1) représente le tracé de Bode de la fonction de transfert de l'intégrateur d'ordre 0.62 et son approximée. Cette figure montre que les tracés de Bode de la fonction $\frac{1}{s^{0.62}}$ et son approximé sont superposés sur la bande d'approximation, on remarque que l'amplitude

est une droite de pente $-20 \times 0.62 = 12.4 \text{ db/dec}$ et que la phase est une constante de $-0.62 \times \pi/2 = -55.8 \text{ deg}$.

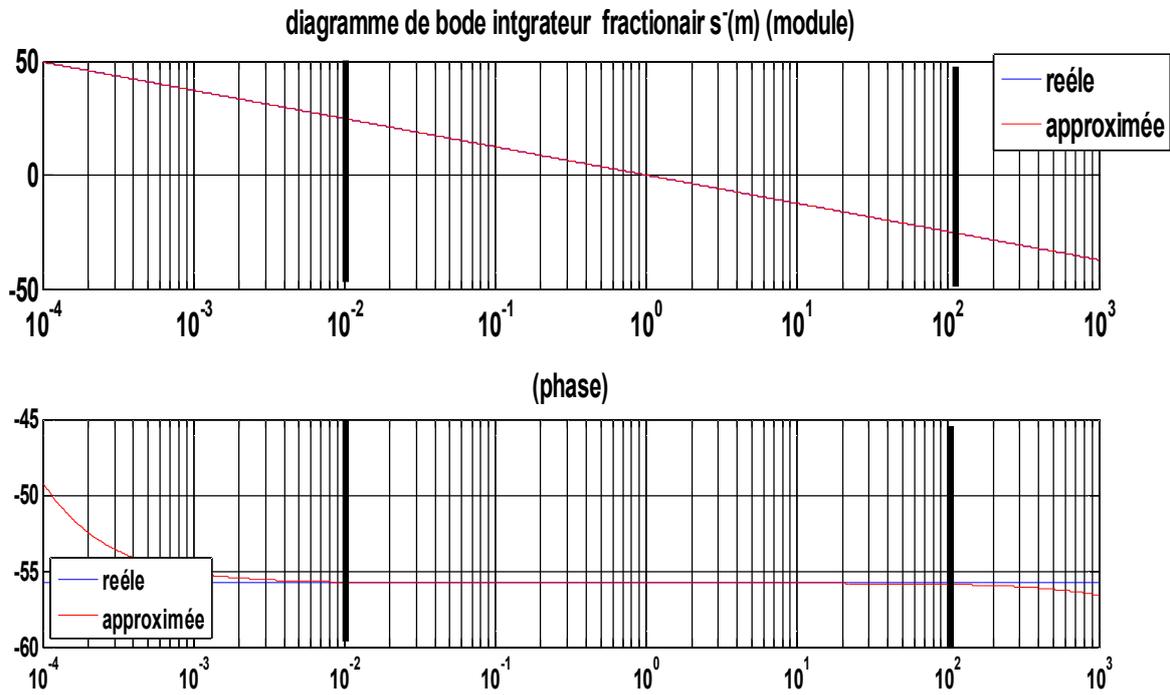


Figure (I.4) Tracé de Bode de l'intégrateur fractionnaire $1/s^{0.6}$ et son approximée.

I.5. Le Correcteur $PI^\lambda D^\mu$ Fractionnaire

En automatique, ce n'est qu'au début des années 1990 que le régulateur CRONE (Commande Robuste d'Ordre Non Entier) a été proposé par Oustaloup [12]. En profitant des propriétés avantageuses des systèmes d'ordre fractionnaire, ce régulateur permettait d'assurer la **robustesse** de la commande dans une bande de fréquences donnée. Depuis cette initiative, La commande d'ordre fractionnaire captiva l'intérêt de beaucoup de chercheurs.

En 1999, Podlubny [13] a proposé le régulateur $PI^\lambda D^\mu$, une généralisation du correcteur PID classique, comprenant une intégration fractionnaire d'ordre λ et une dérivation fractionnaire d'ordre μ , élargissant ainsi le champ d'application du calcul fractionnaire à la théorie de la commande ce qui a orienté plusieurs chercheurs à un nouveau axe de recherche qui est le réglage du correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire [1-12-13-14-15].

I.5.1. Structure du correcteur $PI^\lambda D^\mu$ fractionnaire

La version fractionnaire du correcteur PID avec sa forme la plus commune est donnée par l'équation suivante :

$$u_c(t) = k_p \left(e(t) + \frac{1}{T_I} D^{-\lambda} (e(t)) + T_D D^\mu (e(t)) \right) \quad (I.21)$$

Où K_p est la constante proportionnel, T_I est la constante d'intégration, T_D est la constante de différentiation, λ est l'ordre fractionnaire de l'action d'intégration et μ est l'ordre fractionnaire de l'action de différentiation.

Une expression équivalente, où les paramètres apparaissent de manière linéaire, est souvent plus appréciée pour les calculs analytiques est donnée par la forme parallèle :

$$G_c(s) = K_p + \frac{K_I}{s^\lambda} + K_D s^\mu \quad (I.22)$$

La figure (I.5) présente la structure parallèle interne du $PI^\lambda D^\mu$ d'ordre fractionnaire, définie par des connexions en parallèle entre les parties proportionnelles, intégrale d'ordre fractionnaire et dérivée d'ordre fractionnaire.

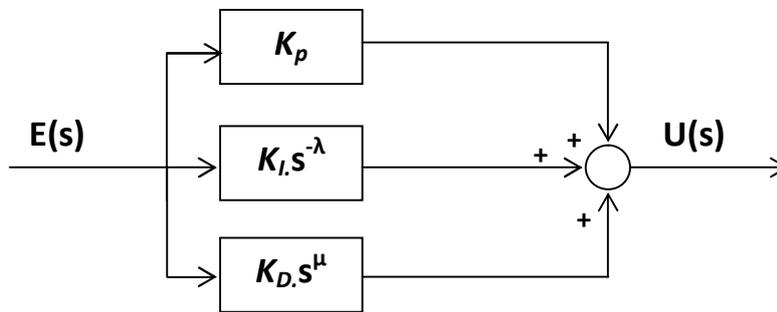


Figure (I.5) Structure du $PI^\lambda D^\mu$ d'ordre fractionnaire

Comme indiqué dans la figure (I.6), le correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire généralise le correcteur PID classique et l'étend du point à un plan. Cette expansion pourrait fournir beaucoup plus de flexibilité dans la conception de commande PID. Clairement, en choisissant $(\lambda, \mu) = (1, 1)$, un correcteur PID classique peut être récupéré et en utilisant $(\lambda, \mu) = (1, 0)$ et $(\lambda, \mu) = (0, 1)$, nous obtenons des correcteurs PI et PD classiques, respectivement. En d'autres termes, tous ces types de correcteurs PID classiques sont des cas spéciaux du correcteur $PI^\lambda D^\mu$ fractionnaire donné par l'équation (I.22).

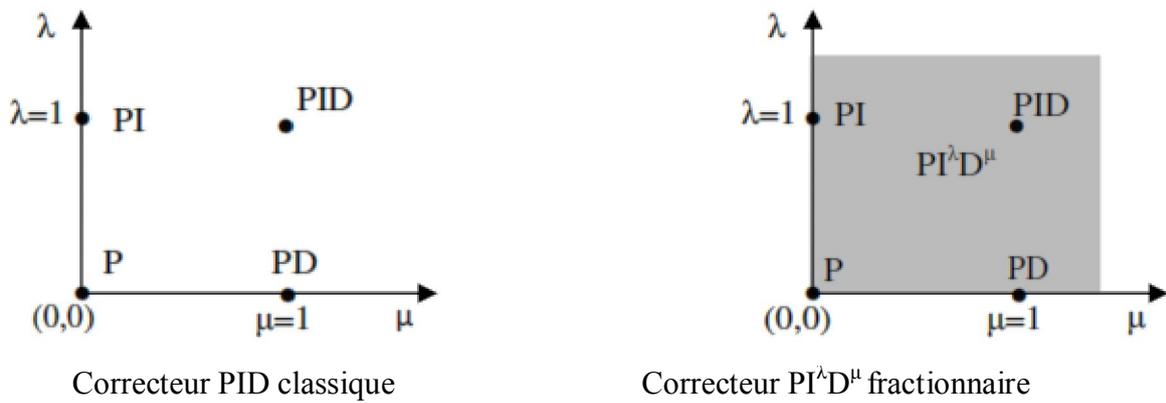


Figure (I.6) Types de correcteurs en fonction de λ et μ

Les figures (I.7) et (I.8) représentent les réponses fréquentielles d'un correcteur $PI^\lambda D^\mu$ fractionnaire avec variation des paramètres λ et μ . Nous pouvons constater que l'utilisation des paramètres intégrale et dérivée d'ordre fractionnaire permet d'ajuster les réponses fréquentielles du correcteur PID classique d'une manière beaucoup plus significative que les coefficients, et le changement des réponses en fréquence est également plus prévisible.

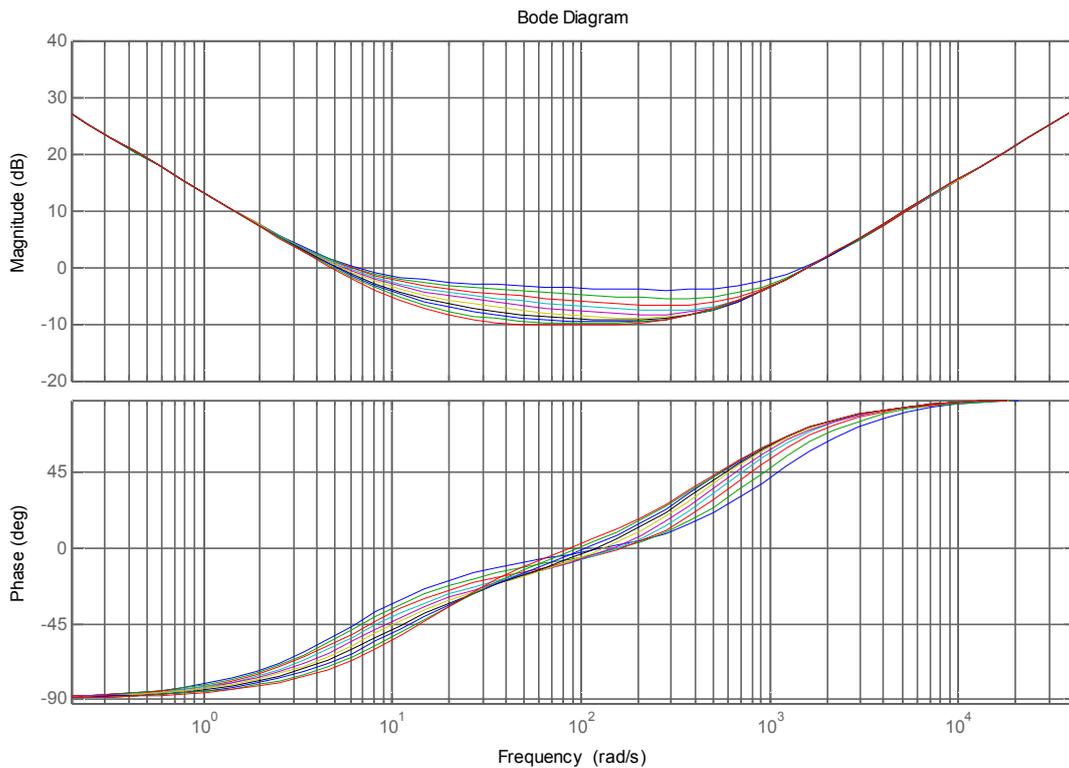


Figure (I.7) Variation de λ du correcteur $PI^\lambda D^\mu$ Fractionnaire

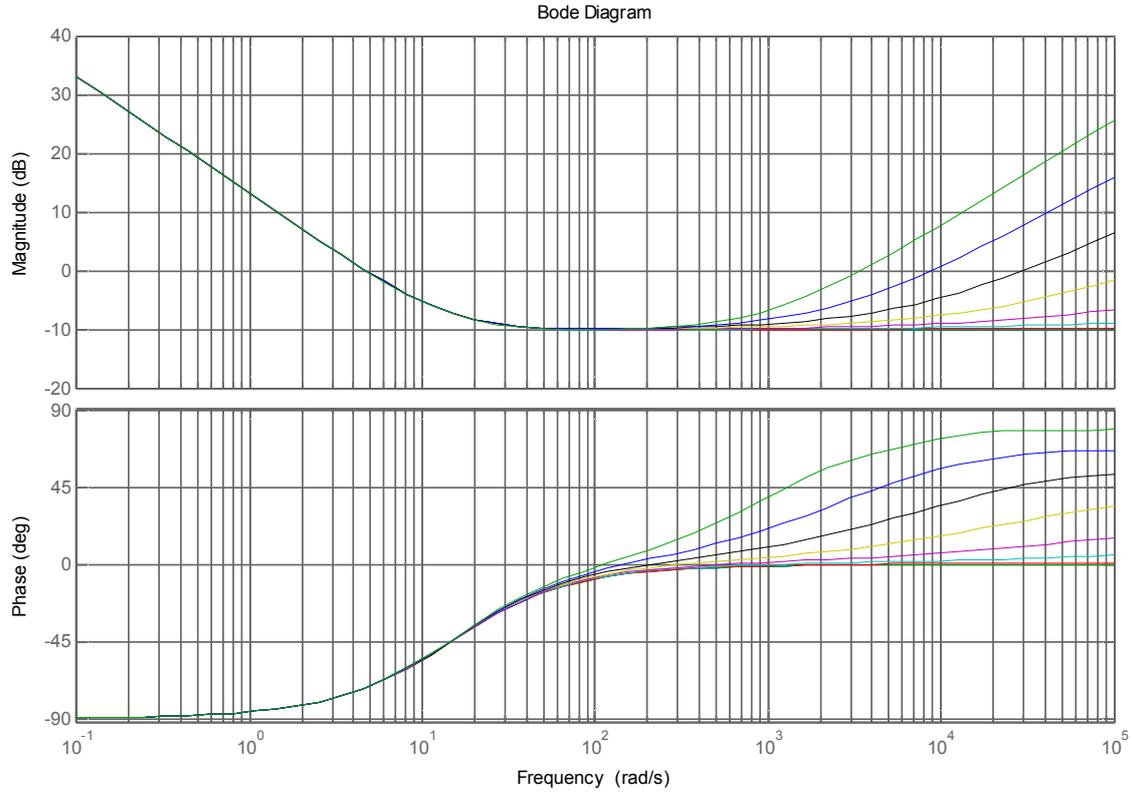


Figure (I.8) Variation de μ du correcteur $PI^\lambda D^\mu$ Fractionnaire

I.5.2. Approximation d'un $PI^\lambda D^\mu$ fractionnaire par une fonction rationnelle

On peut obtenir l'approximation de la fonction de transfert du correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire à l'aide d'une méthode d'approximation rationnelle des opérateurs fractionnaires (intégrateur, dérivée). Dans le cas de l'approximation de Charef, et pour une variations des paramètres λ et μ dans l'intervalle $[0,1]$, l'approximation de la fonction $G_c(s)$ de l'équation (I.22) dans une bande de fréquence donnée d'intérêt pratique $[w_b, w_h]$ est calculée comme suit :

$$G_c(s) = K_P + K_I \left(K_{I_{apx}} \frac{\prod_{i=0}^{n_I-1} \left(1 + \frac{s}{z_{I_i}} \right)}{\prod_{i=0}^{n_I} \left(1 + \frac{s}{p_{I_i}} \right)} \right) + K_D \left(K_{D_{apx}} \frac{\prod_{i=0}^{n_D} \left(1 + \frac{s}{z_{D_i}} \right)}{\prod_{i=0}^{n_D} \left(1 + \frac{s}{p_{D_i}} \right)} \right) \quad (I.22)$$

3.2.a. Exemple Illustratif :

Considérons le correcteur $PI^\lambda D^\mu$ défini par sa fonction de transfert suivante :

$$C(s) = 2.4 + \left(\frac{0.18}{s^{1.7}} \right) + 6(s^{1.7})$$

Et qui s'écrit donc :
$$C(s) = 2.4 + \frac{0.18}{s} \left(\frac{1}{s^{0.7}} \right) + 6s (s^{0.7})$$

En utilisant la formule (2.18), nous obtenons l'approximation rationnelle du PI^λD^μ en approximant les opérateurs $\frac{1}{s^{0.7}}$ et $s^{0.7}$ sur la même bande fréquentielle $[w_b, w_h] = [0.1 \text{ rad/sec}, 10 \text{ rad/sec}]$. Les paramètres d'approximation utilisés sont $\epsilon = 10^{-5}$ et $y=1$. Les pôles et zéros de l'approximation peuvent être calculés à partir de l'équation (I-19)

D'où la fonction de transfert rationnelle suivante :

$$C(s) = 2.4 + \frac{0.18}{s} \left\{ 415.9188 \frac{\prod_{i=0}^{14} \left[1 + \left(\frac{s}{4.6060 \times 10^{-4} (2.9936)^i} \right) \right]}{\prod_{i=0}^{15} \left[1 + \left(\frac{s}{2.1379 \times 10^{-4} (2.9936)^i} \right) \right]} \right\} + 6s \left\{ 0.0024 \frac{\prod_{i=0}^{15} \left[1 + \left(\frac{s}{2.1379 \times 10^{-4} (2.9936)^i} \right) \right]}{\prod_{i=0}^{15} \left[1 + \left(\frac{s}{4.6060 \times 10^{-4} (2.9936)^i} \right) \right]} \right\}$$

La figure (I.9) représente le tracé de Bode, amplitude et phase, de la fonction de transfert du PI^{1.7}D^{1.7} et son approximation rationnelle, on remarque la bonne approximation sur la bande $[0.1, 10 \text{ rad/s}]$.

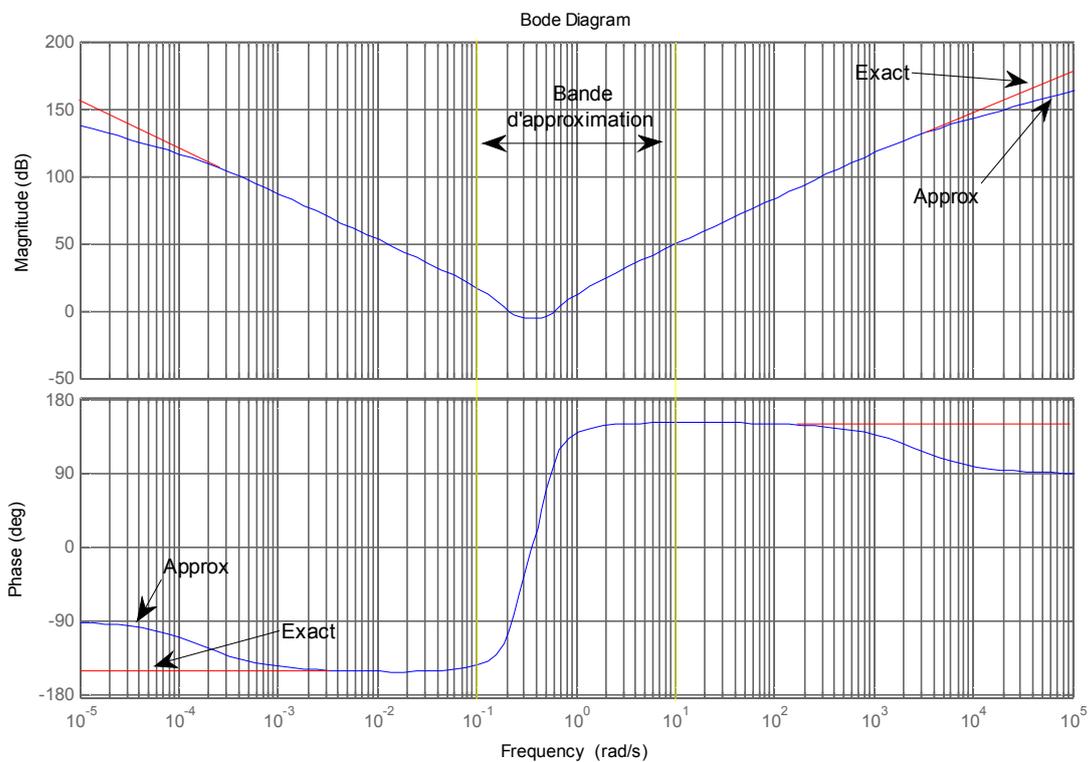


Figure (I.9) Tracé de Bode de la fonction de transfert du PI^{1.7}D^{1.7} et son approximation

Conclusion

Dans ce chapitre, nous avons expliqué la notion du calcul fractionnaire, dans lequel les ordres de dérivation et d'intégration peuvent être d'ordre non entier. On s'inspirant de ce concept, le correcteur $PI^{\lambda}D^{\mu}$ fractionnaire a vu le jour, celui-ci étant un correcteur PID classique avec deux paramètres complémentaires qui sont l'ordre d'intégration et de dérivation fractionnaire. Différentes approximations peuvent être utilisées pour le calcul d'une fonction de transfert rationnelle du correcteur fractionnaire, permettant ainsi son utilisation.

Cependant, le domaine de détermination des paramètres du correcteur $PI^{\lambda}D^{\mu}$ fractionnaire reste un sujet d'actualité, dont le but est de trouver des techniques plus efficaces et optimale dans la détermination de ces paramètres. Donc, dans le cadre de cet mémoire, nous avons exploré l'utilisation des techniques d'optimisations pour déterminer les cinq paramètres du correcteur $PI^{\lambda}D^{\mu}$ fractionnaire, et qui seront présentée et détaillée dans le chapitre suivant. Deux méthodes intelligentes d'optimisation basée au premier lieu sur l'optimisation par essaim Particulaire (PSO: Particle Swarm Optimization),et en second lieu sur l'algorithme génétique (GA : Genetic algorithm) seront étudiés.

Chapitre II

Les Méthodes d'optimisations des
paramètres $PI^\lambda D^\mu$ fractionnaire

Introduction

L'algorithme d'optimisation fait partie des moyens que nous proposons aux professionnels qui souhaitent mettre en place une stratégie efficace pour l'amélioration de leurs résultats. Ce type de procédé est particulièrement recommandé pour la conception de solutions sur mesure destinées à des projets délicats dont les problématiques sont aussi nombreuses que complexes. L'algorithme d'optimisation est adapté à tous les secteurs d'activité.

Plusieurs méthodes d'optimisations ont été proposées dans la littérature, et ont prouvé leur efficacité dans différents domaines d'application. Nous nous intéressons dans ce chapitre à deux de ces méthodes, l'algorithme génétique (GA), et l'optimisation par essaim particulaire (PSO).

II.1. Optimisation par Essaim Particulaire(PSO):

Cet algorithme a été proposé par Russel Eberhart (ingénieur en électricité) et James Kennedy (socio-psychologue) en 1995 [16], l'origine de cette méthode vient des observations faites lors des simulations informatiques de vols groupés d'oiseaux et de bancs de poissons de REYNOLD [17], HEPPNER et GRENANDER [18]. Les individus observés sont à la recherche de sources de nourriture qui sont dispersés de façon aléatoire dans un espace de recherche, et dès lors qu'un individu localise une source de nourriture, les autres individus vont alors chercher à le reproduire.

Ce comportement social basé sur l'analyse de l'environnement et du voisinage constitue alors une méthode de recherche d'optimum par l'observation des tendances des individus voisins. Chaque individu cherche à optimiser ses chances en suivant une tendance qu'il modère par ses propres vécus.

II.2. Principe de fonctionnement de PSO:

L'optimisation par essaim particulaire repose sur un ensemble d'individus originellement disposés de façon aléatoire et homogène, que nous appellerons dès lors des **particules**, qui se déplacent dans l'hyper-espace de recherche et constituent, chacune, une solution potentielle. Chaque particule dispose d'une mémoire concernant sa meilleure solution visitée ainsi que la capacité de communiquer avec les particules constituant son entourage. À partir de ces informations, la particule va suivre une tendance faite, d'une part, de sa volonté à retourner vers sa solution optimale ; est d'autre part, de son mimétisme par rapport aux solutions trouvées dans son voisinage.

La figure (II.1) montre l'agissement des particules d'un essaim. À partir d'optimums locaux et empiriques, l'ensemble des particules va, normalement, converger vers la solution **optimale globale** du problème traité.

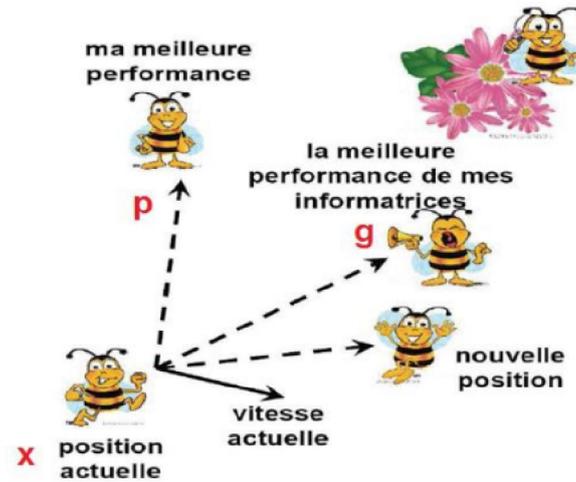


Figure (II.1) schéma de principe du comportement des particules d'un essaim

Le déplacement d'une particule est influencé par trois composantes [19]:

- Une composante d'inertie: permet de définir la capacité d'exploration de chaque particule en vue d'améliorer la convergence de la méthode et la particule tend à suivre sa direction courante de déplacement.
- Une composante cognitive : la particule tend à se fier à sa propre expérience et, ainsi, à se diriger vers le meilleur site par lequel elle est déjà passée.
- Une composante sociale : la particule tend à se fier à l'expérience de ses congénères et, ainsi, à se diriger vers le meilleur site déjà atteint collectivement par l'essaim

Dans un espace de recherche de dimension D la particule i de l'essaim est modélisée par son vecteur position $\vec{x}_{ij} = (x_{i1} \ x_{i2} \ \dots \ x_{iD})^T$ et par son vecteur vitesse $\vec{v}_{ij} = (v_{i1} \ v_{i2} \ \dots \ v_{iD})^T$.

La qualité de sa position est déterminée par la valeur de la fonction objective (cout) en ce point. Cette particule garde en mémoire la meilleure position par laquelle elle est déjà passée, que l'on note $\vec{p}_{ij} = (p_{i1} \ p_{i2} \ \dots \ p_{iD})^T$. La meilleure position atteinte par ses particules voisines est notée $\vec{g}_{ij} = (g_{i1} \ g_{i2} \ \dots \ g_{iD})^T$.

Le vecteur vitesse est calculé à partir de l'équation (II.1) suivante [19]:

$$\vec{v}_{ij} = w \cdot \vec{v}_{ij}(k-1) + c_1 r_1 (p_{ij}(k-1) - x_{ij}(k-1)) + c_2 r_2 (g_{ij}(k-1) - x_{ij}(k-1)) \quad (II.1)$$

La position à l'itération k de la particule i est alors définie par l'équation (II.2)

$$x_{ij}(k) = x_{ij}(k-1) + v_{ij}(k) \quad (II.2)$$

avec $i=1, 2, 3, \dots, N_p$, $J=1, 2, 3, \dots, N_d$, $k=1, 2, 3, \dots, k_{max}$, et sachant que :

N_p : nombre de particules de l'essaim

N_d : nombre de variables du problème (c-à-d la dimension d'une particule)

k_{max} : nombre maximal d'itération.

$v_{ij}(k)$: vitesse de la $j^{ème}$ composante de la $i^{ème}$ particule de l'essaim, à la $k^{ème}$ itération. p_{ij} : $j^{ème}$ Composante de la meilleure position occupée par la $i^{ème}$ particule de l'essaim enregistrée dans les itérations précédentes (local best)

g_{ij} : $j^{ème}$ Composante de la meilleure position occupée par la $i^{ème}$ particule globale de l'essaim (global best)

$x_{ij}(k)$: $j^{ème}$ Coordonnée de la position actuelle de particule i à la $k^{ème}$

ω : constante appelée coefficient d'inertie, c_1 et c_2 : sont deux constantes appelées coefficients d'accélération, r_1 et r_2 : sont deux nombres aléatoires tirés uniformément dans $[0,1]$ à chaque itération et pour chaque dimension

$\omega.v_{ij}(k-1)$: Correspond à la composante physique du déplacement. Le paramètre ω contrôle l'influence de la direction de déplacement sur le déplacement futur.

$c_1 r_1 (p_{ij}^{best} - x_{ij}(k-1))$ Correspond à la composante cognitive du déplacement où c_1 contrôle le comportement cognitif de la particule

$c_2 r_2 (g_{ij}^{best} - x_{ij}(k-1))$ Correspond à la composante sociale du déplacement ; ou contrôle l'aptitude sociale de la particule

II.3. Mécanisme de confinement

Il arrive parfois que la position courante et la vitesse courante d'une particule, avoir cette dernière une tendance à sortir de son espace de recherche lors de son déplacement. Lorsque c'est le cas, l'algorithme fait intervenir un mécanisme de confinement, dit confinement d'intervalle, afin de gérer le déplacement de la particule, qui leurs amènent dans un nouveau point qui appartenant à l'espace de recherche. De manière générale, ce mécanisme consiste à ramener la particule dans le point admissible qui est le plus proche [20][21].

$$\begin{cases} \text{si } x_{ij} > x_{jmax} & \text{alors } x_{ij} = x_{jmax} \\ \text{si } x_{ij} < x_{jmin} & \text{alors } x_{ij} = x_{jmin} \end{cases} v_{ij}=0 \quad (\text{II.3})$$

Où x_{jmax} et x_{jmin} sont les valeurs limites du paramètre x_{ij}

Dans une autre part, le coefficient d'inertie ω joue un rôle important dans la procédure de recherche car elle garantit un équilibre entre la recherche locale et la recherche globale, en 2002 M. Clerc et J. Kennedy [22] ont présenté une autre approche de calcul qui introduit un coefficient de construction χ qui permet de contrôler la vitesse au cours de itération et donc au pallier de convergence et de nouveaux paramètres relatifs aux composantes cognitive et sociale ϕ_1 et ϕ_2 ont ainsi été introduit selon l'équation (3.1)

$$\chi = \frac{2.k}{|2-\phi-\sqrt{\phi^2-4.\phi}|} \text{ avec } 0 \leq k \leq 1 \quad (\text{II.4})$$

$$\phi = \phi_1 + \phi_2 \geq 4 ; \text{ et } \phi_1 = \phi_2 = 2.05 ; \quad (\text{II.5})$$

Et les paramètres de l'équation (II.1) deviennent

$$\begin{cases} \omega = \chi \\ c_1 = \chi \cdot \phi_1 \\ c_2 = \chi \cdot \phi_2 \end{cases} \quad (\text{II.6})$$

A partir de l'équation (II.4) on peut éliminer la vitesse v_{max}

L'algorithme PSO se déroule suivant les étapes suivantes [20]:

1. initialiser une population de particules et de vitesses, uniformément réparties dans l'espace de recherche D , et fixer le critère d'arrêt.
2. évaluer la fonction objective, pour chaque particule.
3. mettre à jour la meilleure position P_{best} pour chaque particule et la meilleure position globale g_{best} dans la population
4. mettre à jour la position et la vitesse utilisant les équations (II.1) et (II.2)
5. si une particule, déplacée par l'algorithme vers la limite ou à l'extérieur de l'espace de recherche, le mécanisme de confinement des particules exprimé par les équations (II.3) intervient et ramène la particule à l'intérieur de l'espace de recherche.
6. vérifier le critère d'arrêt. Si le critère d'arrêt n'est pas satisfait, aller vers l'étape 2; sinon, le programme se termine, et la solution optimale est produite.

La figure (II.2) montre l'organigramme qui résume ces étapes

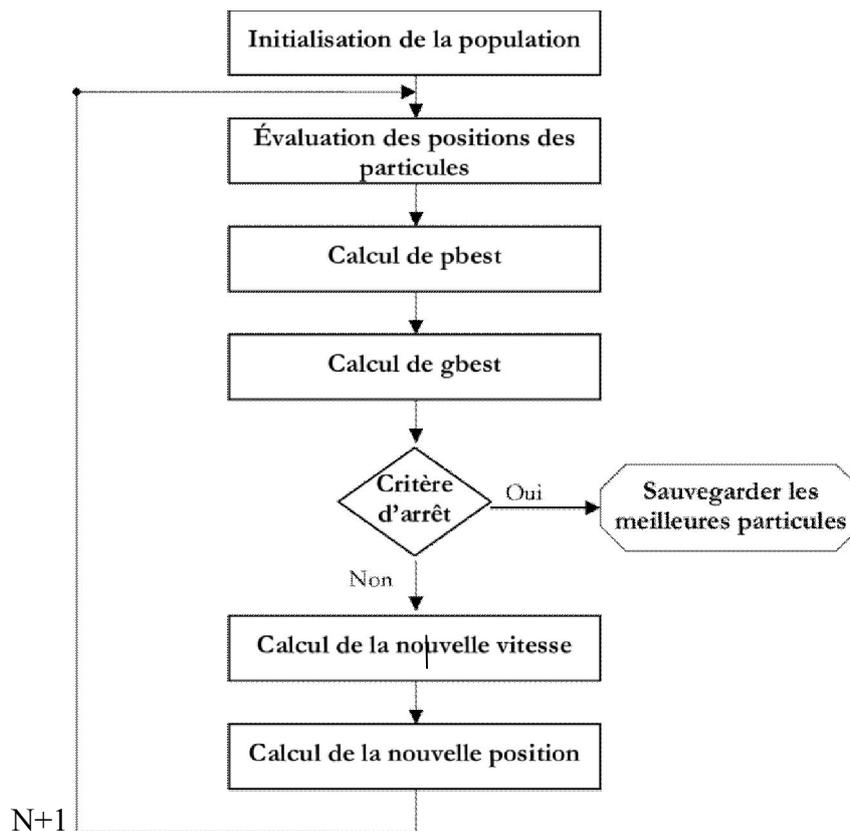


Figure (II.2) Organigramme général de la méthode PSO

II.4. Optimisation par Algorithme génétique

Les algorithmes génétiques sont des algorithmes évolutionnistes d'optimisation stochastique inspirés des mécanismes de l'évolution naturelle (sélection, adaptation, reproduction, recombinaison, mutation) élaborés par Charles Darwin. Ils sont des algorithmes itératifs de recherche globale dont le but est d'optimiser une fonction prédéfinie appelée critère ou fonction coût "fitness". Ils permettent de faire évoluer un ensemble initial de solutions vers un ensemble final. Les algorithmes génétiques sont utilisés dans l'automatique, le but de cette utilisation est d'optimiser au maximum les gains des régulateurs utilisés dans la commande. A partir des années 1990, quelques publications évoquent l'utilisation des AG pour la résolution de problèmes d'optimisation dans le domaine de la commande de systèmes électromécaniques.

II.4.1. Présentation de l'algorithme génétique (GA)

Selon Lerman et Ngouenet (1995)[6] un algorithme génétique est défini par :

- **Individu/chromosome/séquence** : une solution potentielle du problème A de longueur $l(A)$ une suite $A=\{a_1, a_2 \dots a_l\}$; avec $\forall i \in [1, l], a \in V=\{0,1\}$. Un chromosome est donc une suite de bits en codage binaire, appelé aussi chaîne binaire. Comme montré sur la figure (II.3). Dans le cas d'un codage non binaire, tel que le codage réel, la suite A ne contient qu'un point, nous avons $A=\{a\}$, $a \in R$.

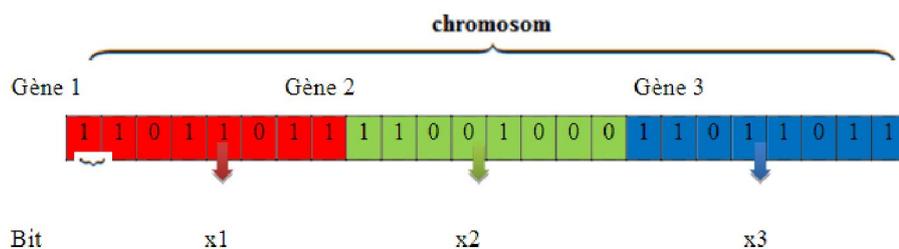


Figure (II.3) Chromosome avec son codage binaire

- **Environnement** : l'espace de recherche ;
- **Fonction de fitness** : la fonction - positive - que nous cherchons à maximiser.

Les algorithmes génétiques sont alors basés sur les phases suivantes (étapes mis en oeuvre) :

1. **Initialisation** : Une population initiale de 'N' chromosomes est tirée aléatoirement
2. **Codage et population initiale avec** Le codage est une modélisation d'une solution d'un problème donné sous forme d'une séquence de caractères « chromosome » où chaque caractère, dit aussi gène, représente une variable ou une partie du problème figure (II.3). La tâche principale consiste à choisir le contenu des gènes qui facilite la description du problème et respecte ses contraintes.
3. **Évaluation.** Chaque chromosome est décodé, puis évalué (fonction optimisation appelée fitness des individus) le résultat fournit par cette fonction va permettre de sélectionner ou de refuser un individu pour ne garder que les individus ayant le meilleur coût en fonction de la population courante [5].

4. **Sélection** : Création d'une nouvelle population de N chromosomes par l'utilisation d'une méthode de sélection appropriée.
5. **Reproduction possible** : Des opérateurs de croisement et mutation au sein de la nouvelle population, permettant de diversifier la population au cours des générations et d'explorer l'espace d'état.

II.4.2. Les Opérateurs :

Les opérateurs jouent un rôle prépondérant dans la possible réussite d'un GA. Nous en dénombrons trois principaux : **l'opérateur de sélection, de croisement** et de **mutation**. Si le principe de chacun de ces opérateurs est facilement compréhensible

II.4.2.a. L'opérateur de sélection :

Cet opérateur est chargé de "favoriser" les meilleurs individus. Plus formellement, va générer à partir de la population courante une nouvelle population par copie des individus choisis de la population courante. La copie des chaînes s'effectue en fonction des valeurs de la fonction d'adaptation. Ce procédé permet de donner aux meilleures chaînes, une probabilité élevée de contribuer à la génération suivante ; il y a deux types de sélection

La sélection par classement elle consiste à ranger les individus de la population dans un ordre croissant selon l'objectif et seuls les individus les plus forts sont conservés

La sélection par tournoi : le tournoi le plus simple consiste à choisir aléatoirement un nombre k d'individus dans la population et à sélectionner celui qui a la meilleure performance et qui participent à un tournoi (avec une probabilité $p \in [0; 1]$) sont remis ou sont retirés de la population,

II.4.2.b. L'opérateur de Croisement :

Il s'agit d'un processus essentiel pour explorer l'espace des solutions possible Pour La naissance d'un nouvel individu et cela faite par la prise aléatoire d'une partie des gènes de chacun des deux parents, montré par figure (II.4). Une fois la sélection terminée les individus sont aléatoirement répartis en couples. Les chromosomes parents sont alors copiés et recombinaison afin de produire chacun deux descendants ayant des caractéristiques issues des deux parents. Dans le but de garder quelques individus parents dans la prochaine population associée à l'algorithme génétique une probabilité de croisement, qui permet de décider si les parents seront croisés entre eux ou s'ils seront tout simplement recopiés dans la population suivante.

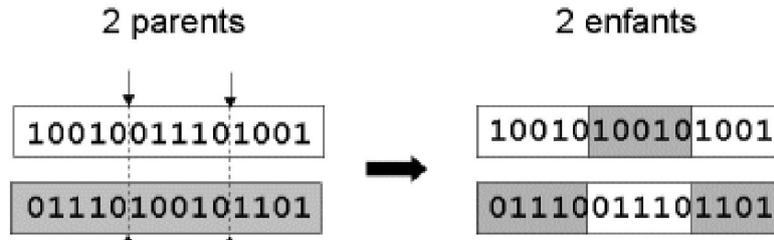


Figure (II.4) Croisement par deux points

II.4.2.c.l'opérateur de mutation[5] :

Génère de nouveaux gènes, afin de permettre d'explorer la totalité (en théorie) de l'espace d'état, ce qui correspond à la propriété d'ergodicité de parcours d'espace, essentielle aux AG pour leurs propriétés de convergence.

Son fonctionnement est comme suit : Pour les problèmes discrets, un gène du chromosome est tiré aléatoirement et sa valeur est remplacée par une des autres valeurs possibles (tirée aléatoirement elle aussi).

Dans le cas des problèmes continus, le gène est également tiré aléatoirement, et remplacé par une valeur aléatoire du domaine d'extension des gènes (espace d'état). La figure(II.5) suivante montre un exemple de mutation.

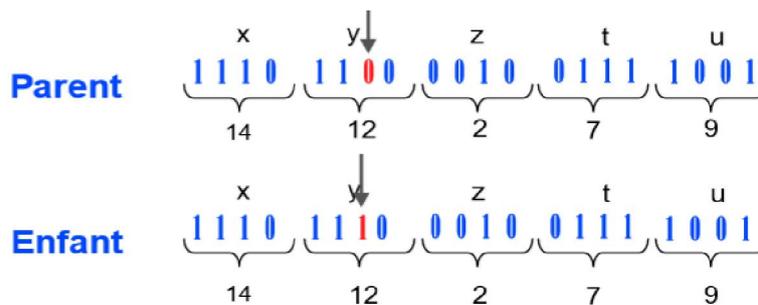


Figure (II.5) Opération de mutation

Le cycle de génération et de sélection de population est répété jusqu'à ce qu'un critère d'arrêt soit satisfait ; ce critère peut être notamment un nombre maximum de générations, un temps maximal de calcul, une valeur de fitness minimale, ou/et une convergence vers une solution satisfaisante c'est-à-dire le Retour à la phase d'évaluation jusqu'à l'arrêt de l'algorithme. L'organigramme de la figure (II.6) résume les étapes de l'algorithme génétique.

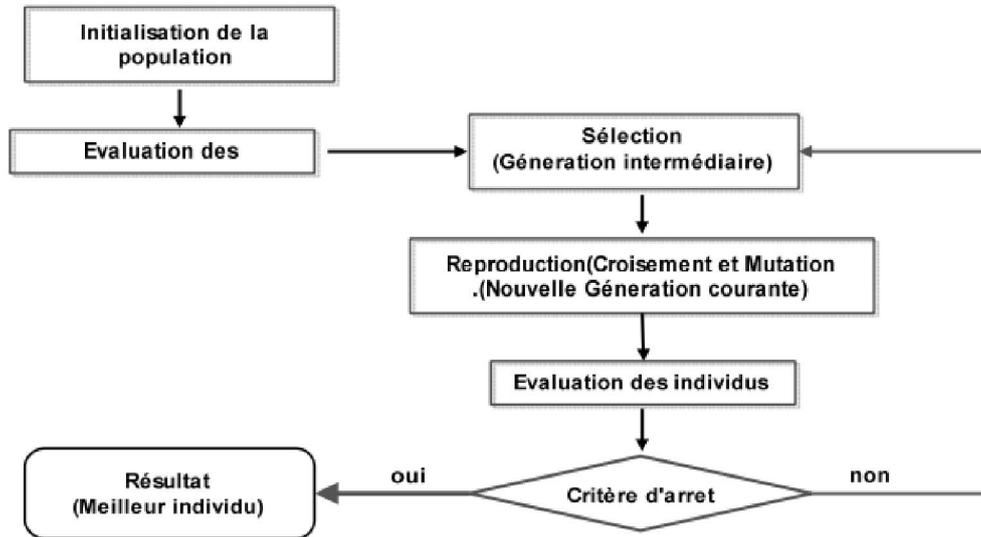


Figure (II.6) Organigramme de fonctionnement d'un algorithme génétique.

II.5. Ajustement des Paramètres d'un régulateur PID par optimisation

Si l'on considère que « p » est un vecteur qui contient les cinq paramètres du correcteur fractionnaire $PI^{\lambda}D^{\mu}$ (ou les trois paramètres du PID classique), et dans le but de régler et ajuster les gains de ces correcteurs, on peut faire appel à des méthodes d'optimisations pour avoir un résultat optimal. Ces paramètres seront donc représentés par un ensemble de particule ou de chromosomes, pour les algorithmes PSO et GA, respectivement. En suivant les étapes de l'algorithme, et pour un certain nombre d'itérations, on a une forte probabilité de trouver une solution fiable de notre problème d'optimisation. Ce problème étant, dans notre cas, défini par un critère de l'erreur de régulation en boucle fermée, comme le démontre la figure (II.7). Des critères de performances (fonction coût) sont définis, dès lors, à partir de cette erreur.

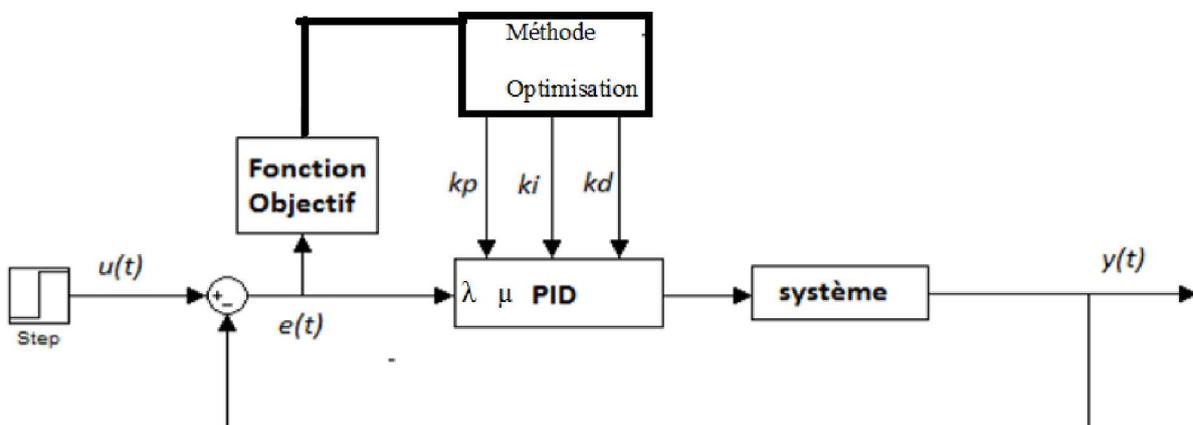


Figure (II.7) Approche de optimisation du fractionnaire $PI^{\lambda}D^{\mu}$ par les méthodes méta-heuristique

II.5.1. Critère de performances

Pour avoir une bonne précision dynamique d'un système à une entrée échelon, il faut que le régime transitoire soit caractérisé par un faible dépassement et un temps de réponse optimal. Pour cela, les paramètres d'un régulateur sont choisis de manière à minimiser l'erreur dynamique : $e(t) = y_{ref}(t) - y_s(t)$ (II.7)

Nous utiliserons deux critères dans ce mémoire qui sont les suivants :

II.5.1.a. L'intégrale du carré de l'erreur (ISE):

Il représente l'intégrale de l'erreur quadratique, il est donné par :

$$ISE = \int_0^{+\infty} e^2(t) dt \quad (II.8)$$

II.5.1.b. L'intégrale de l'erreur absolue (IAE)

Il représente l'intégrale de la valeur absolue de l'erreur, il est donné par :

$$IAE = \int_0^{+\infty} |e(t)| dt \quad (II.9)$$

Le critère IAE est souvent employé pour la simulation numérique des systèmes, cependant il est inapplicable pour le travail analytique du fait que la valeur absolue d'une fonction erreur n'est pas toujours exprimable sous forme analytique. Ce problème est surmonté par le critère d'ISE. Les critères ITAE et ITSE ont un multiplicateur additionnel qui est le temps de la fonction erreur et qui souligne les erreurs de longue durée. Ces critères sont le plus souvent appliqués dans les systèmes ayant besoin d'un temps de réponse rapide.

Conclusion

Dans ce chapitre Nous avons présenté le principe des algorithmes d'optimisation par PSO et GA et leurs implémentations, ainsi que leur utilisation pour l'ajustement des paramètres d'un régulateur PID classique ou $PI^\lambda D^\mu$ fractionnaire, et ce, en utilisant un critère de performance exprimé en fonction de l'erreur de régulation.

Chapitre III

Simulation et Application

Introduction

Le correcteur standard le plus utilisé dans l'industrie est le correcteur PID, car il permet de régler à l'aide des trois paramètres les performances du système de commande. Dans ce mémoire, nous avons présenté la nouvelle extension de ce correcteur, très utilisé, qui est le correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire. Avec ses deux paramètres additionnels λ et μ , qui représentent l'ordre d'intégration et de dérivation respectivement, il permet de mieux commander la dynamique des systèmes.

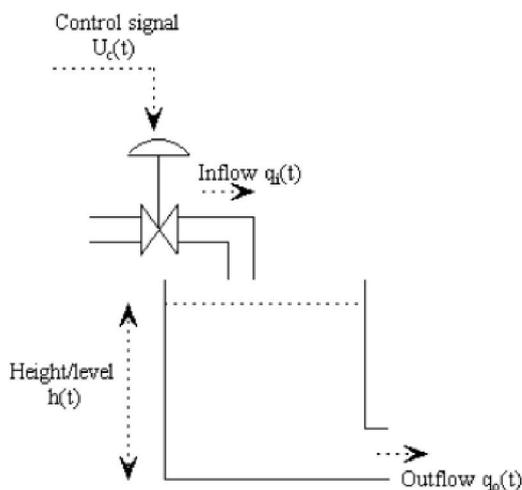
Pour mettre en évidence l'efficacité de ce type de correcteurs, nous nous proposons d'étudier quelques exemples d'applications réels et nous essayons de synthétiser ses correcteurs $PI^\lambda D^\mu$ d'ordres fractionnaires par des méthodes d'optimisations heuristiques ou ses correcteurs essaient de commander et améliorer le mieux possible les performances des systèmes. Comme applications, nous avons choisi :

- ✓ Un asservissement de niveau.
- ✓ Un moteur à courant continu.

L'objectif de ce chapitre est d'apercevoir les qualités et les améliorations obtenues en utilisant un correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire pour la commande des systèmes et les comparer avec les réponses de correcteur PID classique, et ce en fonction du critère et de l'algorithme d'optimisation choisie. Par ailleurs, nous nous intéresserons le moins possible à l'aspect modélisation.

III.1. Asservissement de niveau

Comme première application nous avons considéré l'asservissement de niveau de la figure (III.1). Le processus est composé d'un réservoir contenant du liquide, le niveau de ce dernier est commandé par une valve. Le signal d'entrée (consigne) dicte le niveau du liquide désiré dans le réservoir.



a) Schéma équivalent



b) Processus réel

Figure (III. 1) Asservissement de niveau

Ce processus peut être modélisé par une fonction de transfert de la forme suivante [23] :

$$G(s) = \frac{k}{(\tau s + 1)} e^{-Ls} \quad (\text{III.1})$$

En introduisant les paramètres réels du système, la fonction de transfert obtenue est :

$$G_1(s) = \frac{3.13}{1+433.33} e^{-50s} \quad (\text{III.2})$$

La réponse indicielle du système $G_1(s)$ est montrée par la Figure (III.2). Nous observons que Le temps de réponse est de l'ordre de 1.75×10^3 seconde avec un dépassement nul.

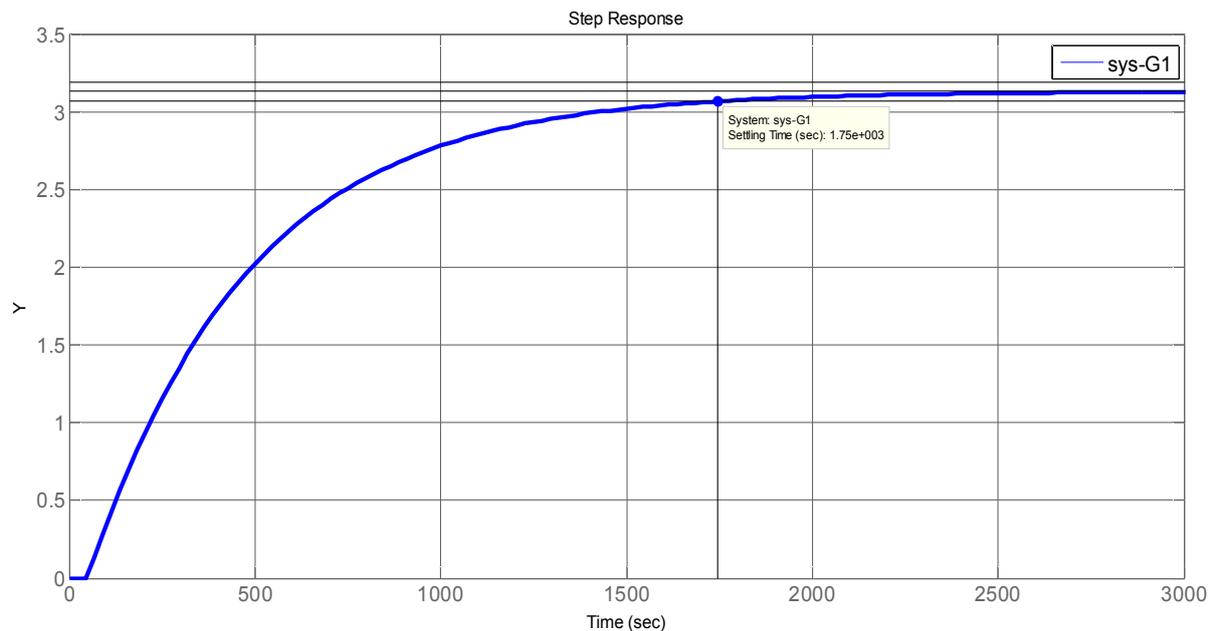


Figure (III. 2) Réponse indicielle du système.

Nous utiliserons dans ce qui suit les algorithmes d'optimisation définis dans le chapitre précédent, à savoir le GA et le PSO, afin de déterminer les paramètres optimaux du PID classique ainsi que le $PI^\lambda D^\mu$ fractionnaire. Les critères de performances utilisés sont l'ISE et l'IAE.

III.1.1.Optimisation par GA

A l'initialisation de l'algorithme, chaque chromosome contient trois gènes, qui correspondent aux paramètres du PID classique (K_p , K_i et K_d), et cinq gènes pour le $PI^\lambda D^\mu$ fractionnaire. La taille de la population initiale est au nombre de 20. Afin d'avoir une meilleure convergence de l'algorithme, les paramètres de départ pour K_p , K_i et K_d seront calculés à l'aide de la méthode Zeigler-Nichols (au lieu de mettre des valeurs aléatoires).

Le choix de l'objectif à minimiser est une étape très importante, puisque le résultat final en dépend fortement. Nous avons utilisé deux critères ; ISE et IAE.

III.1.1.a. Critère ISE :

La figure (III.3) représente les réponses indicielles des systèmes corrigés par un correcteur PID classique et un correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire. Les résultats en termes de performance sont montrés par le tableau (III.1).

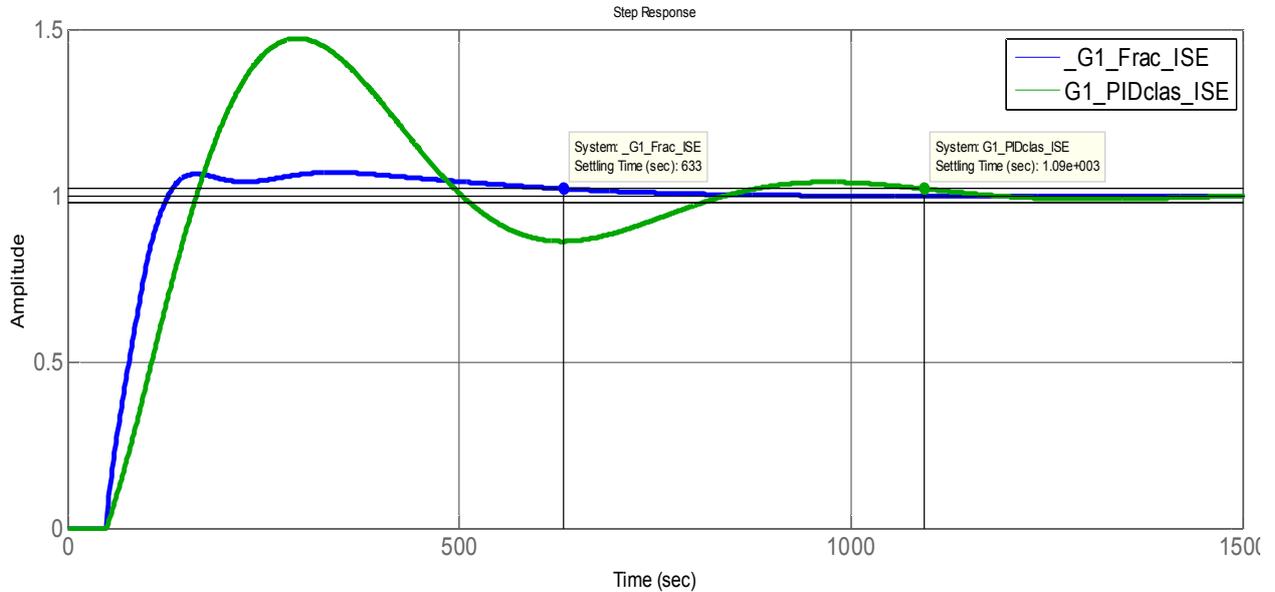


Figure (III. 3) Réponse indicielle de l’asservissement de niveau avec correction PID et $PI^\lambda D^\mu$ (critère ISE-GA)

« GA »	Paramètres	Cout(J)	T. de réponse (sec)	Dépassement(%)
PID	$k_p=1 ;$ $k_i=0.01$ $k_d=4.780$	128.3174	$1.09 \cdot 10^3$	47.2%
$PI^\lambda D^\mu$	$\lambda =0.9345 \quad \mu=0.4015$ $k_p=1 ;$ $k_i=0.01 ; k_d=4.78 ;$	69.6755	633	6.95%

Tableau (III. 1) Performances obtenues pour l’asservissement de niveau par le critère ISE

D’après les résultats du tableau (III.1), on remarque une amélioration considérable obtenue grâce au $PI^\lambda D^\mu$ d’ordre fractionnaire : une diminution du temps de réponse de 457 sec ainsi qu’une baisse du dépassement jusqu’à 40.25%. Ces résultats ont été obtenus suite à un bénéfice de coût d’une valeur de 58.65.

III.1.1.b. Critère IAE

La deuxième simulation a été faite en changeant le critère de performance par un intégral de l’erreur absolue (IAE). Les résultats obtenus par ce critère sont illustrés dans la figure (III.4) et le tableau (III.2).

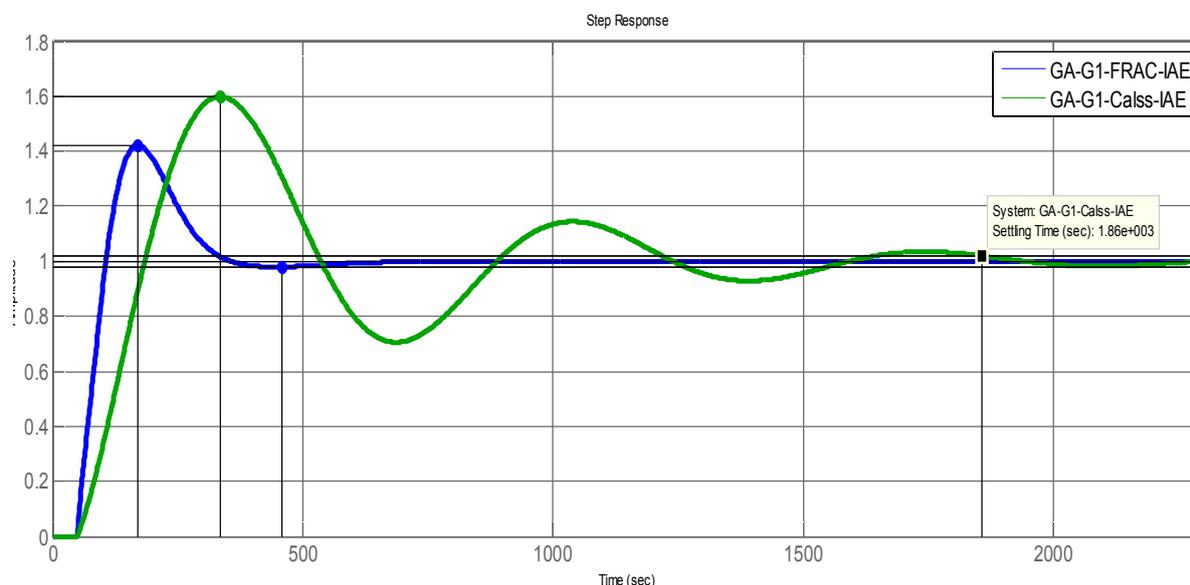


Figure (III. 4) Réponse indicielle de l'asservissement de niveau avec correction PID et $PI^\lambda D^\mu$ par critère IAE-GA

« GA »	Paramètres	Cout (J)	T. de réponse (sec)	Dépassement (%)
PID	$k_p=0.7$; $k_i=0.01$ $k_d=4.780$	380.1375	$1.86 \cdot 10^3$	59.9%
$PI^\lambda D^\mu$	$\lambda=0.856$ $\mu=0.3055$ $k_p=0.7$; $k_i=0.0343$ $k_d=4.1846$	137.72	459	42.3%

Tableau (III. 2) Performances obtenues pour l'asservissement de niveau par le critère IAE

D'après les résultats obtenu pour les deux correcteurs PID Classique et $PI^\lambda D^\mu$ fractionnaire avec le critère IAE, on remarque une amélioration de la réponse obtenue grâce au $PI^\lambda D^\mu$ d'ordre fractionnaire ; une diminution assez importante du temps de réponse est perçue (1401sec), ainsi qu'une baisse moyenne du dépassement jusqu'à 17.6%, correspondant à un bénéfice de coût modéré.

III.2. Optimisation par PSO

Dans le cas d'une optimisation par PSO, les trois paramètres du PID, ou les cinq paramètres d'un $PI^\lambda D^\mu$ forme une particule. Comme dans le cas du GA, l'initialisation des paramètres sera effectuée à base des valeurs obtenues par la méthode de Ziegler-Nichols

L'objectif est donc de trouver la meilleurs solution de position de chaque particule, notée position locale, puis en cherche la meilleur pour un nombre d'itération prédéfini. En fonction d'une minimisation d'un critère de coût (fonction fitness), on arrive donc à obtenir

une solution globale parfaite (meilleure position globale) synonyme d'un coût minimal (meilleure coût globale).

III.2.a. Critère ISE

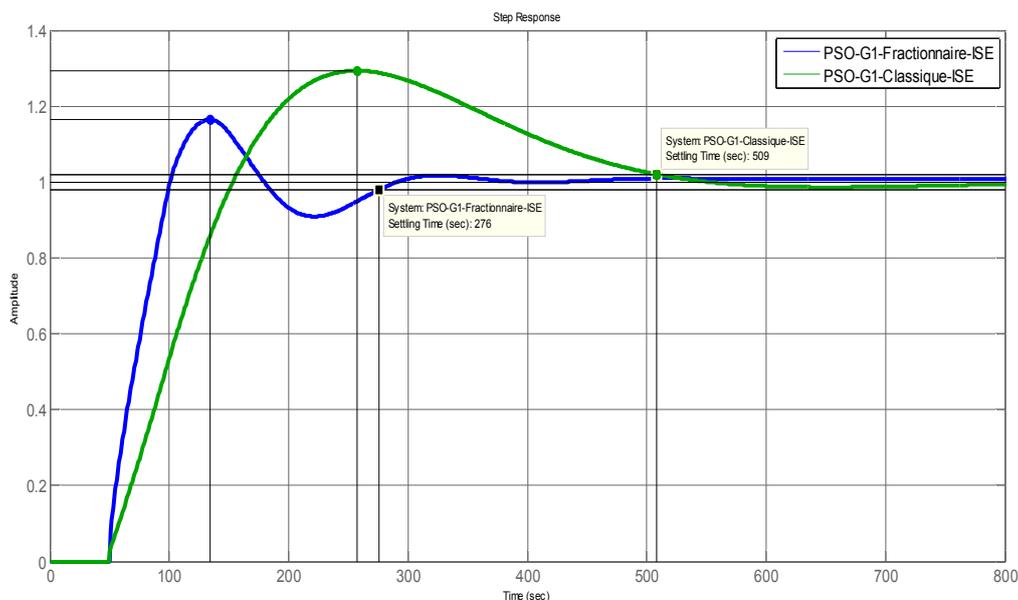


Figure (III. 5) Réponse indicielle de l'asservissement de niveau avec correction PID et $PI^\lambda D^\mu$ par critère ISE-PSO

« PSO »	Paramètres	Coût (J)	T. de réponse (sec)	Dépassement (%)
PID	$k_p=1.2964$; $k_i=0.0081$ $k_d=3.6584$	94.8605	509	29.3
$PI^\lambda D^\mu$	$\lambda = 0.9909$ $\mu = 0.5437$ $k_p=1.7758$; $k_i=0.0053$ $k_d=7.6235$;	65.1685	276	16.4

Tableau (III. 3) Performances obtenues pour l'asservissement de niveau par le critère ISE.(PSO)

D'après les résultats obtenus montrés par la figure(III.5), une diminution du temps de réponse de 233 sec ainsi qu'une baisse moyenne du dépassement jusqu'à 12.9% sont perçues pour un gain en coût de 29.692

III.2.2.b. Critère IAE

Les résultats obtenus par le critère IAE est sont montrés par la figure (III.6) et le tableau (III.4).

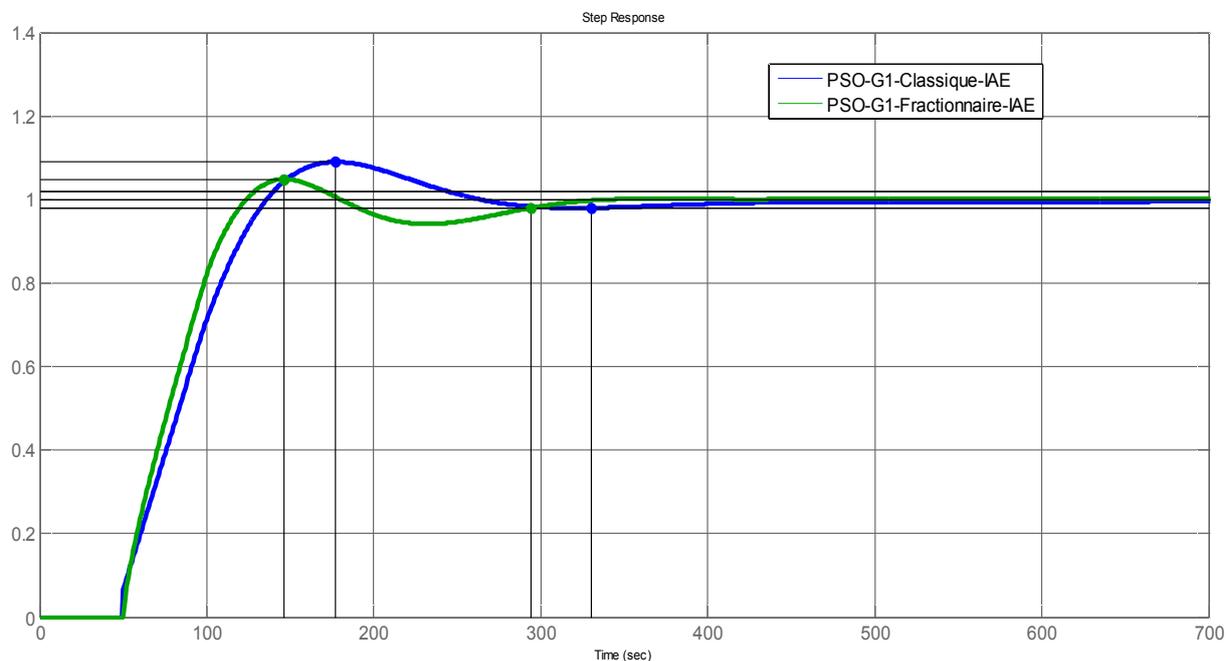


Figure (III. 6) Réponse indicielle de l'asservissement de niveau avec correction PID et $PI^\lambda D^\mu$ par critère IAE-PSO

« PSO »	Paramètres	Cout (J)	T. de réponse (sec)	Dépassement (%)
PID	$k_p=1.8246$; $k_i=0.038$ $k_d=9.3823$	98.173	330	09.01
$PI^\lambda D^\mu$	$\lambda = 0.8951$ $\mu = 0.3682$ $k_p=1.0315$ $k_i=0.0087$ $k_d=4.6585$	93.2491	294	04.92

Tableau (III. 4) Performances obtenues pour l'asservissement de niveau part le critère IAE.

Pour le cas du critère IAE, une diminution du temps de réponse de 36 sec, ainsi qu'une baisse du dépassement jusqu'à 4.09%, sont les résultats d'un bénéfice en coût d'une valeur de 4.933.

D'après ces résultats, on note ce qui suit

- Le $PI^\lambda D^\mu$ fractionnaire donne de meilleurs résultats par rapport au PID classique pour le système $G_1(s)$
- La meilleure valeur minimale de la fonction de coût (J) pour le système $G_1(s)$ est obtenue par le critère ISE
- la méthode PSO assure les meilleures performance possibles pour le système $G_1(s)$

Nous avons aussi conduit une comparaison en termes de performances des correcteurs PID fractionnaire dont les paramètres sont issus des algorithmes GA et PSO dans notre travail, avec les résultats obtenus dans la référence [23] par la fonction FMINCON. Comme illustrée sur la figure (III.7)

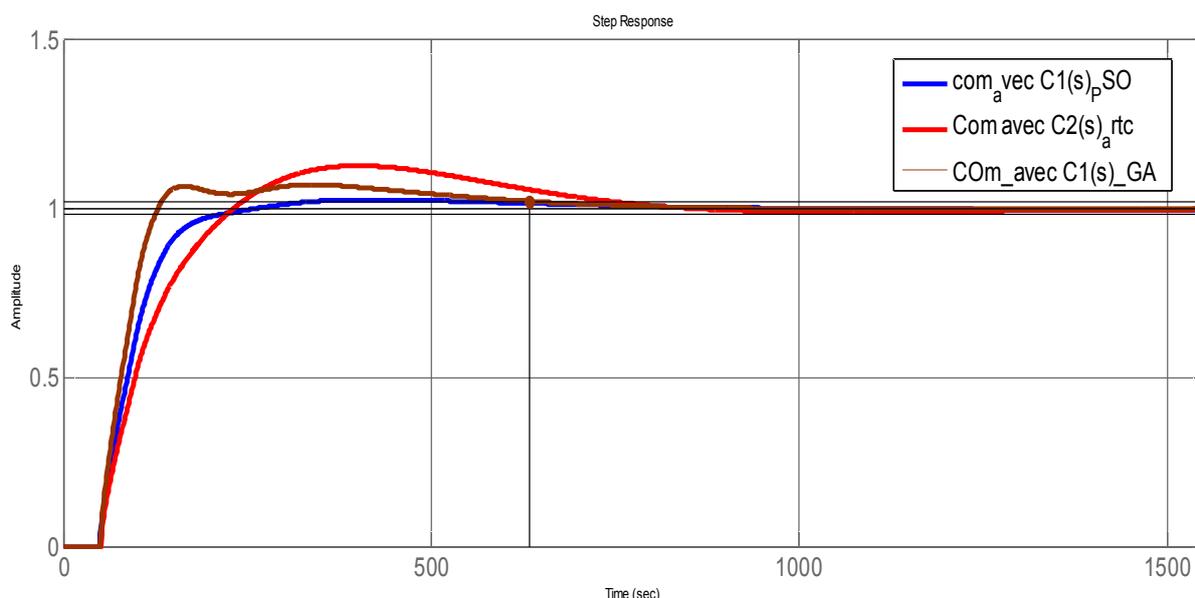


Figure (III. 7) les réponses de correcteur fractionnaire avec les résultats de référence

D'après les réponses sur la figure (III.7), nous constatons que la supériorité des performances obtenues à l'aide d'algorithmes métha-heuristique (PSO et GA dans notre cas) par rapport aux approches d'optimisation classiques (FMINCON- gradient).

III.3. Commande d'un moteur à courant continu

Un moteur à courant continu est une machine électrique. Il s'agit d'un convertisseur électromécanique permettant la conversion bidirectionnelle d'énergie entre une installation électrique parcourue par un courant continu et un dispositif mécanique, d'où l'énergie électrique est transformée en énergie mécanique.

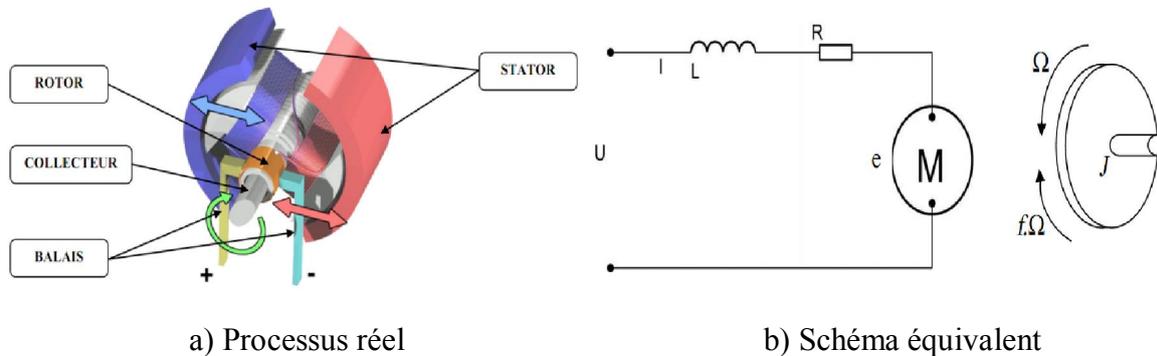


Figure (III. 8) Le moteur à courant continu

La fonction de transfert cherchée $G(s)$ est entre la tension entrante dans le moteur $U(s)$ et la vitesse de sortie $\Omega(s)$ [24], les valeurs numériques sont données dans le tableau (III.5).

$$G(s) = \frac{\Omega(s)}{U(s)} = \frac{K_c}{(R + Ls)(f + Js) + K_c K_e} \quad (\text{III.3})$$

Tension nominale	$U_{nom} = 24 \text{ V}$
Tension maximale	$U_{max} = 32 \text{ V}$
Courant permanent maximal	$I_{max} = 2.2 \text{ A}$
Résistance aux bornes	$R = 1.91 \Omega$
Constante de couple	$K_c = 60.3 \text{ mNm.A}^{-1}$
Constance de vitesse	$K_e = 60.3 \text{ mNm.A}^{-1}$
Inductance	$L = 0.63 \text{ mH}$
Moment d'inertie rapporté au rotor	$J = 10^{-4} \text{ kg.m}^2$
Coefficient de frottement visqueux	$f = 2.5 \times 10^{-6}$

Tableau (III. 5) Valeurs numériques d'un moteur à courant continu
(Maxon: Référence F2260/813)

En procédant à un asservissement en vitesse et en position du MCC, nous pouvons établir le modèle mathématique de la réponse en vitesse du moteur électrique, où elle est montré par les figures (III.8).

Sachant que La fonctions de transfert, de l'asservissement en vitesse était obtenue avec les valeurs numériques du tableau (III.5) données, par [24]:

$$G_V(s) = \frac{0.0603}{6.3 \times 10^{-8} s^2 + 1.91 \times 10^{-4} s + 36.41 \times 10^{-4}} \quad (III.4)$$

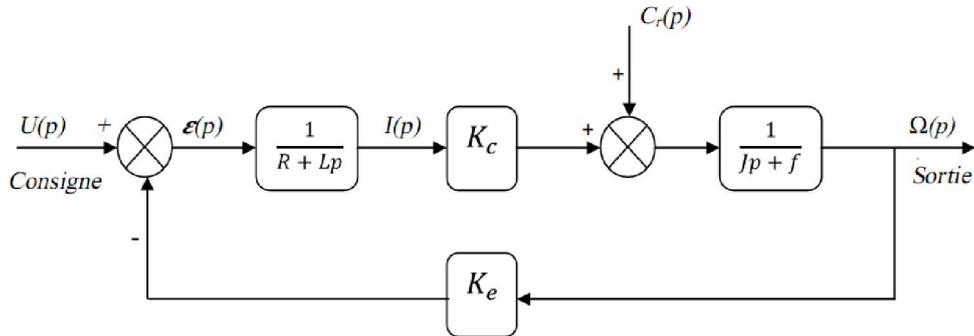


Figure (III. 9) Schéma bloc du Modèle de moteur électrique en vitesse

Le MCC peut fonctionner sous plusieurs modes, dans ce qui suit, nous considérons un moteur à courant continu à excitation séparée, l'objectif est d'annuler l'erreur statique, diminuer le dépassement et diminuer le temps de réponse et le temps de montée afin d'obtenir une réponse adéquate du procédé et de la régulation et d'avoir un système précis, rapide, stable et robuste (Figure (III.9))

La réponse indicielle du système exprimé par la vitesse de moteur $G_2(s)$ est montrée par la Figure (III.10). Nous observons que le temps de réponse est de l'ordre de 20.4×10^{-2} seconde avec un dépassement nul.

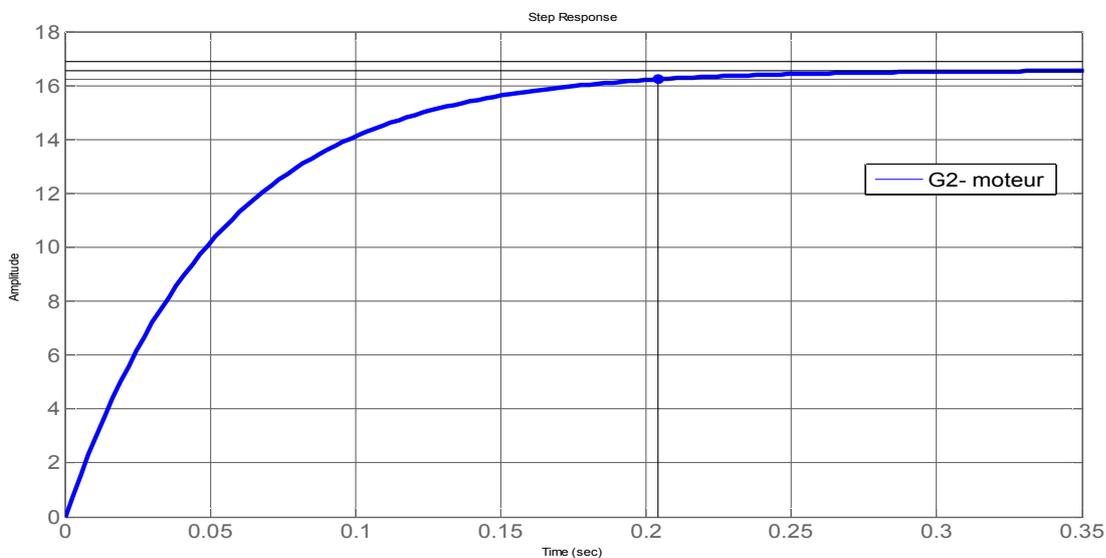


Figure (III. 10) Réponse indicielle du système. vitesse MCC

III.3.1. Régulation en vitesse

Après avoir essayé les différentes valeurs de K_u , nous avons relevé le gain d'oscillation $K_u=25$ et la période d'oscillation $T_u=3.6 \times 10^{-4}$. En appliquant les règles de la méthode de Ziegler-Nichols, nous déduisons les valeurs des paramètres: $K_p=20.83$, $T_i= 0.0037$, $T_d=0.000048$. . Ainsi, nous utiliserons ces paramètres pour l'initialisation des algorithmes d'optimisations par les méthodes GA et PSO.

III.3.2. Réglage par GA:

Avant de lancement de l'algorithme du GA, nous avons choisi une taille de population initiale égale à 20. Les chromosomes étant constitués de trois gènes, qui correspondent aux paramètres du PID classique K_p , K_i et K_d , et cinq gènes pour le cas du $PI^{\lambda}D^{\mu}$ fractionnaire.

III.3.2.a. Critère ISE

Une comparaison entre les performances obtenues par les deux correcteurs classique et fractionnaire sont dans le tableau (III.6) et La figure (III.11) pour le critère ISE.

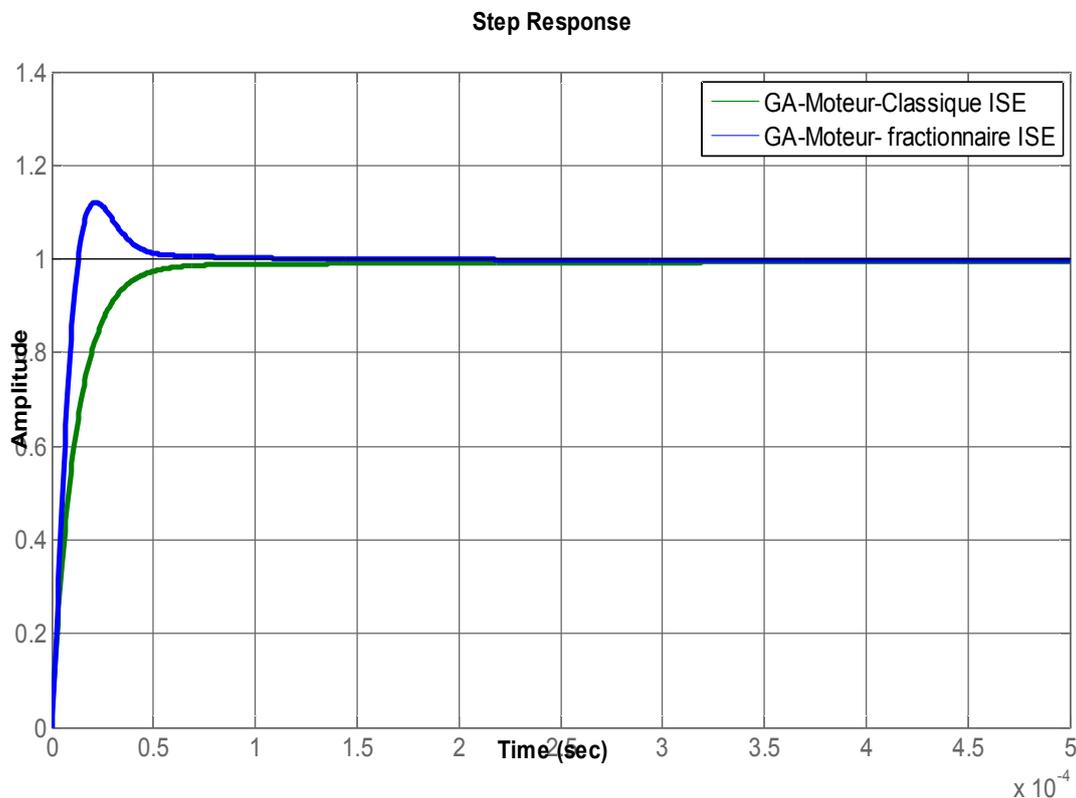


Figure (III. 11) Réponse indicielle de vitesse du moteur avec correction PID et $PI^{\lambda}D^{\mu}$ critère ISE-GA

« GA »	Paramètres	Cout (J)	T. de réponse (sec)	Dépassement (%)
PID	$k_p=173$ $k_i=-0.7975$ $k_d=0.0872$	1.5877×10^{-5}	erreur statique	0
$PI^\lambda D^\mu$	$\lambda = 0.6119$ $\mu=0.7117$ $k_p=368.147$ $k_i=-1.8537$ $k_d=4.7301$	3.977×10^{-6}	4.485×10^{-5}	12.2

Tableau (III. 6) Performances obtenues pour La vitesse de moteur par le critère ISE – GA.

D'après les résultats du tableau (III.6), on remarque une amélioration considérable obtenue grâce au $PI^\lambda D^\mu$ d'ordre fractionnaire : un bénéfice de coût d'une valeur 1.19×10^{-5} et annulation d'erreur statique, contrairement au PID classique.

III.3.2.b. Critère IAE

La deuxième simulation a été faite en changeant le critère de performance par un intégral de l'erreur absolue (IAE). Les résultats obtenus par ce critère sont illustrés dans la figure (III.12) et le tableau (III.7).

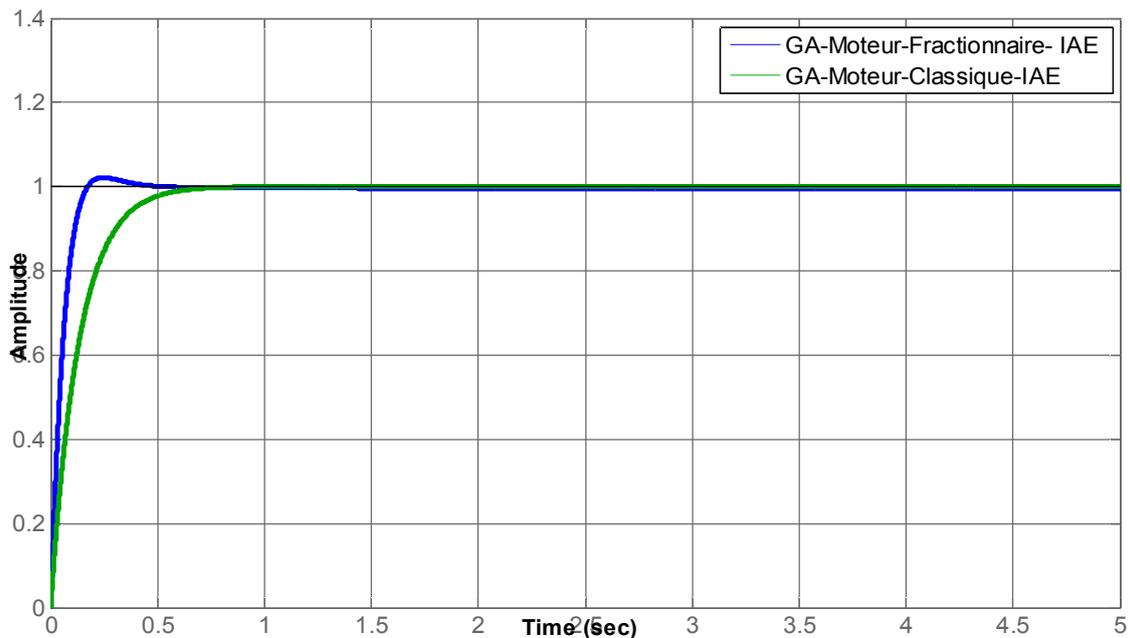


Figure (III. 12) Réponses indicielles de système vitesse du moteur après correction de $PI^\lambda D^\mu$ et PID classique. GA-IAE

« GA »	Paramètres	Coût (J)	T. de réponse (sec)	Dépassement (%)
PID	$k_p=244.2424$ $k_i=4.7901$ $k_d=0.0785$	1.3368×10^{-5}	5.15×10^{-5}	0.06
$PI^\lambda D^\mu$	$\lambda=-0.2728$ $\mu=0.9002$ $k_p=229.9392$ $k_i=2.4434$ $k_d=0.5992$	7.4354×10^{-6}	2.84×10^{-5}	2.21

Tableau (III. 7) Performances obtenues pour La vitesse de moteur part le critère IAE -GA

On remarque, à partir de ces résultats, une diminution du temps de réponse de 2.31×10^{-5} sec et un bénéfice de coût d'une valeur 5.9326×10^{-6} . Même si une augmentation du dépassement (2.21%) est perçue pour le correcteur $PI^\lambda D^\mu$ d'ordre fractionnaire, il reste tout de même plus rapide que celui corrigé par un PID classique

III.3.3. Réglage par PSO

Dans le cas d'une optimisation par PSO, les trois paramètres du PID, ou les cinq paramètres d'un $PI^\lambda D^\mu$ forme une particule. Comme dans le cas du GA, l'initialisation des paramètres sera effectuée à base des valeurs obtenues par la méthode de Ziegler-Nichols

L'objectif est donc de trouver la meilleure solution de position de chaque particule, notée position locale, puis en chercher la meilleure pour un nombre d'itération prédéfini (100 itérations). En fonction d'une minimisation d'un critère de coût (fonction fitness), on arrive donc à obtenir une solution globale parfaite (meilleure position globale) synonyme d'un coût minimal (meilleure coût globale).

III.3.3.a. Critère ISE

Les différents résultats obtenus pour ce critère sont données par la figure (III.13) et

le tableau (III.8) pour le système $G_2(s)$

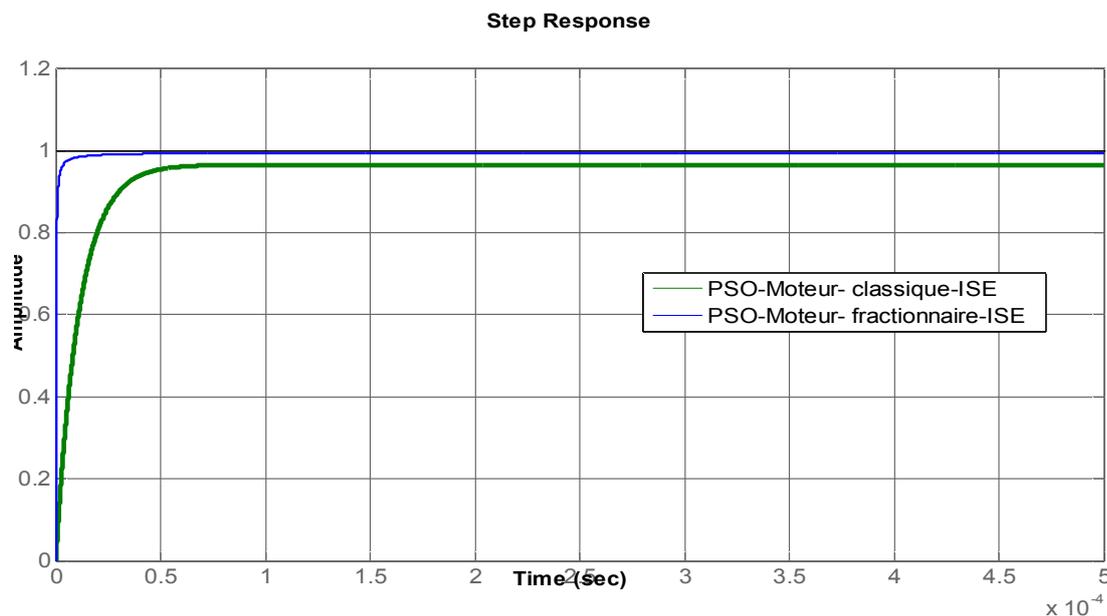


Figure (III. 13) Réponses indicielles de système vitesse de moteur après correction de $PI^\lambda D^\mu$ et PID classique. **PSO-ISE**

« PSO »	Paramètres	Coût (J)	T. de réponse (sec)	Dépassement (%)
PID	$k_p=0.2;$ $k_i=40.09$ $k_d=0.09$	6.5501×10^{-6}	erreur statique	0
$PI^\lambda D^\mu$	$\lambda = 0.6$ $\mu = 1.255$ $k_p=201.755$ $k_i=5.255 \quad k_d=0.1125$	3.0406×10^{-6}	7.19×10^{-6}	0

Tableau (III. 8) Performances obtenues pour moteur par le critère ISE -PSO

D'après les résultats obtenus, on remarque qu'il y a une amélioration grâce au $PI^\lambda D^\mu$ d'ordre fractionnaire ; correspondant à un bénéfice de coût d'une valeur de

3.5095×10^{-6} , ainsi que l'annulation de l'erreur statique, chose que le correcteur PID classique n'assure pas.

III.3.3.b. Critère IAE

Les résultats obtenus par le critère IAE est sont montrés par la figure (III.14) et le tableau (III.9) pour le système $G_2(s)$.

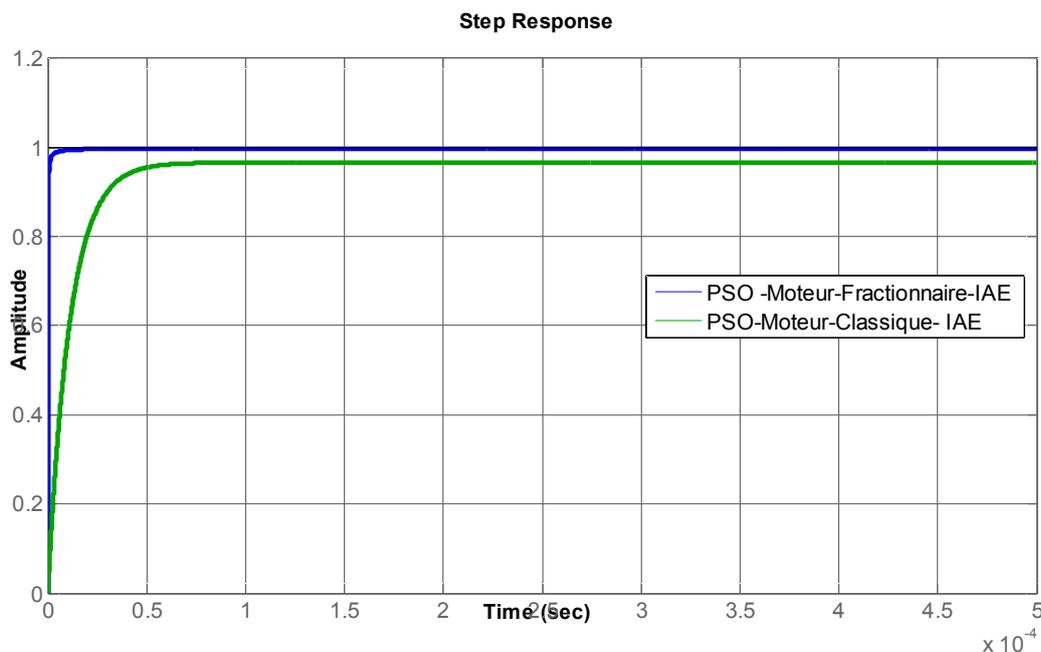


Figure (III. 14) Réponses indicielles de vitesse de moteur PSO-IAE

« PSO »	Paramètres	Coût (J)	T. de réponse (sec)	Dépassement (%)
PID	$k_p=0.085$; $k_i=0.4$ $k_d=0.09$	2.7895×10^{-5}	Erreur statique	0
$PI^\lambda D^\mu$	$\lambda = 0.676$ $\mu = 1.255$ $k_p=200.068$ $k_i=4.068$ $k_d=0.2855$	1.2544×10^{-6}	1.94×10^{-6}	0

Tableau (III. 9) Performances obtenues pour vitesse de moteur part le critère ISE-PSO-

D'après les résultats obtenu, un bénéfice de coût d'une valeur de 2.66406×10^{-5} résulte de l'utilisation du $PI^\lambda D^\mu$ d'ordre fractionnaire, ainsi qu'une annulation de l'erreur statique, qui reste toujours présente dans le cas du correcteur PID classique.

D'après les résultats de cette application, on note le suivant :

- le $PI^\lambda D^\mu$ fractionnaire donne améliore la réponse du système $G_2(s)$ et permet l'annulation de erreur statique
- la valeur minimale de la fonction de coût (J) pour le système $G_2(s)$ est fournie par le critère IAE
- l'algorithme PSO assure les meilleures performances du système $G_2(s)$

Conclusion

Suite à plusieurs tests effectués sur deux différents systèmes, asservissement de niveau et moteur à courant continu, et à l'aide de différents algorithmes (PSO-GA), différents correcteurs (PID- $PI^\lambda D^\mu$), et en optant pour différents critères (ISE-IAE), on a pu conclure que le correcteur $PI^\lambda D^\mu$ fractionnaire fournie de meilleure performance de le contrôle, et que l'algorithme PSO reste le plus performant lors de la détermination des paramètres de ce correcteur. Les critères utilisés fournissent des résultats variant selon l'application. Ces conclusion sont propres à ces deux cas de systèmes et ne peuvent donc pas être généralisé pour tous les systèmes, une possibilité d'avoir un résultat meilleur pour une autre approche reste donc envisageable.

Conclusion générale

Nous avons présenté dans ce mémoire une approche de commande de type $PI^\lambda D^\mu$ d'ordre fractionnaire, une variante du correcteur PID classique plus flexible. Afin de régler les paramètres de ce correcteur, nous avons fait appel à deux méthodes d'optimisations : l'optimisation par essaim particulaire (PSO) et l'algorithme génétique (GA). Il s'agit donc de trouver les paramètres $(\lambda, \mu, K_p, K_i, K_d)$ pour améliorer les performances de la boucle de régulation. Pour construire la fonction objective, nécessaire à l'évolution des solutions durant le processus d'optimisation, nous avons utilisé deux critères de performances, IAE et ISE. Nous avons également validé ces approches par une application sur deux systèmes d'ordres différents : un asservissement de niveau, et un moteur à courant continu.

Les résultats obtenus en terme de régulation, pour le nouveau correcteur $PI^\lambda D^\mu$ fractionnaire sont très encourageants, comparé au correcteur PID classique dont les paramètres optimaux sont aussi calculés à l'aide de méthodes d'optimisations. Cela justifie l'intérêt porté ces dernières années à ce type de correcteur fractionnaire et aux méthodes de sa conception. Cependant, il y a bien des cas où l'utilisation du correcteur PID classique reste largement suffisante afin d'obtenir le résultat désiré, car même si le $PI^\lambda D^\mu$ fractionnaire offre de meilleures performances, son utilisation n'est pas toujours nécessaire. Il s'agit donc d'utiliser le correcteur approprié afin de satisfaire ses besoins en régulation.

Notre étude s'est limitée sur le plan théorique/simulation. Comme perspectives, nous proposons l'implémentation de ce type de correcteur analogiquement (à base des composants électroniques) ou numériquement (avec des cartes d'implémentations numériques), afin de percevoir ses performance de contrôle sur des systèmes réel.

Bibliographies

- [1] Oustaloup A., *La commande CRONE (commande robuste d'ordre non entier)*, ©Hermès (Traité des Nouvelles Technologies - Série Automatique), Paris, 495 p, 1991.
- [2] I. Podlubny, *Fractional Differential Equations*, Academic Press, San Diego, USA, 1999.
- [3] B.M. Vinagre, V. Feliú, proc: 41st IEEE conference on decision and control, Las Vegas, December 9, 2002.
- [4] S. C. D. Roy, *On the realisation of a constant-argument immittance or fractional operator*, IEEE, Trans. On circuit theory, Vol. 14, no. 3, 264-274, 1967.
- [5] Jean-MARC Alliot, Nicolas Durand, Algorithmes Génétique et Institut national polytechnique de Toulouse; 14 Mars 2005
- [6] G. Aberial Cormier, Algorithmes Génétique Université de Moncton 2008
- [7] Y. Cooren, "Perfectionnement d'un algorithme adaptatif d'optimisation par essaim
- [8] J. Silva, «PID controllers for Time-Delay Systems», Springer, 2005
- [9] Livre: Automatique (contrôle et régulation : Patrick Prouvost). France, 2004
- [10] Philippe de Larminat, *Automatique appliquée*, 2ème édition revue et augmentée, chapitre 2, Hermès Lavoisier mars 1999
- [11] A. Charef, *Analogue realization of fractional order integrator, differentiator and fractional $PI^{\lambda}D^{\mu}$ controllers*, IEE Proceeding on Control Theory Applications, 135(6) : 714-720, 2006.
- [12] A. Oustaloup, *La dérivation non entière*, Hermès, Paris, 1995.
- [13] I. Podlubny, *Fractional Order Systems and $PI^{\lambda}D^{\mu}$ Controllers*, IEEE Transactions on Automatic Control, 44(1), pp 208–214, 1999.
- [14] R. Caponetto and L. Fortuna, *A new tuning strategy for a non-integer order PID Controller*, Proc of the 1st IFAC Workshop on Fractional Differentiation and its Applications,
- [15] B.M. Vinagre, I. Petráš, P. Merchan, and L. Dorcak. *Two digital realisations of fractional controllers: Application to temperature control of a solid*. Proc. European Control Conference (ECC2001), pages 1764-1767, Porto, Portugal, 2001.

- [16] Kennedy, J. and Eberhart, R. (1995). Particle Swarm Optimization. In the IEEE International Joint Conference on Neural Networks, pages 1942–1948, Perth
- [17] Craig W. Reynolds. Flocks, herds, and schools : A distributed behavioral model. *Computer Graphics*, 21(4) :25–34, 1987.
- [18] F. Heppner and U. Grenander. A stochastic nonlinear odel for coordinated bird flocks. AAAS Publication, Washington, DC, 1990.
- [19] M.T Huynh «Application de l'Optimisation par Essaim Particulaire à la représentation de matrice de préférence valuées» Université Libre De Bruxelles 2009.
- [20] M.T Huynh «Application de l'Optimisation par Essaim Particulaire à la représentation de matrice de préférence valuées» Université Libre De Bruxelles 2009.
- [21] R Mansouri « Contribution a l'analyse et la synthèse des systèmes d'ordre fractionnaire par la représentation d'état »thèse de Doctorat en Electrotechnique UMMTO 2008.
- [22] Clerc, M. and Kennedy, J. (2002). The Particle Swarm : Explosion, Stability, and Convergence in a Multi-Dimensional Complex Space. In *Proceedings of the IEEE Transactions on Evolutionary Computation*, volume VI, pages 58–73.
- [23] **Monje, C. A., Vinagre, B. M., Feliu, V., & Chen, Y. (2008). Tuning and auto-tuning of fractional order controllers for industry applications. *Control engineering practice*, 16(7), 798-812.**
- [24] **Bouiche Hachemi et Brahami Mohamed, *Commande PID d'un moteur à courant continu*. Mini projet. Université Abderrahmane Mira, Bejaia, 2009.**