جامعة سيدي محمد بن عبد الله بفاس
UNIVERSITÉ SIDI MOHAMED BEN ABDELLAH DE FES

كلية العلوم والتقنيات فاس
FACULTÉ DES SCIENCES ET TECHNIQUES DE FÈS

N° d'ordre : 107/2023

Centre des Etudes Doctorales : Sciences et Techniques et Sciences Médicales
Formation Doctorale : Sciences de l'Ingénieur, Sciences Physiques, Mathématiques et Informatique
Discipline : Génie électrique
Spécialité* : Génie électrique/Systèmes intelligents
Laboratoire : Systèmes Intelligents, Géoressources et Energies Renouvelables

Sciences de l'Ingénieur, Sciences Physiques, Mathématiques et Informatique

Thèse pour l'obtention de DOCTORAT

Conception et optimisation des circuits intégrés analogiques avancés

- Métaheuristiques et réseaux de neurones artificiels -

Préparée par : Mr Abdelaziz LBERNI

Soutenue le 16/12/2023 devant le jury :

Nom et Prénom	Grade	Etablissement	Qualité
Mohammed RAMDANI	PES	ESEO d'Anger, France	Président
Patrick SIARRY	PES	Université Paris-Est Créteil, France	Rapporteur
Mourad LOULOU	PES	Université de Sfax, ENIS-Sfax, Tunisie	Rapporteur
Fatima ERRAHIMI	PES	Faculté des Sciences et Techniques de Fès	Rapporteur
Lotfi KAMOUN	PES	Université de Sfax, ENIS-Sfax, Tunisie	Examinateur
Mustapha ALAMI	PES	Institut National des Postes et Télécommunications, Rabat	Examinateur
Abdelaziz AHAITOUF	PES	Faculté Polydisciplinaire de Taza	Examinateur
Malika ALAMI MARKTANI	PH	Ecole Nationale des Sciences Appliquées de Fès	Examinateur
AHAITOUF Ali	PES	Faculté des Sciences et Techniques de Fès	Directeur de thèse

Synthèse détaillée en Français

Résumé

Ce document est l'aboutissement de cinq années de travail. Les axes de recherche développés ont été motivés par le grand besoin en microélectronique d'une gestion automatique et simplifiée des tâches laborieuses de dimensionnement des circuits intégrés analogiques. En effet, ce processus de dimensionnement occupe une part importante du temps des concepteurs analogiques et peut être considéré comme un réel problème d'optimisation du monde actuel. Les problèmes d'optimisation à grande échelle sont généralement difficiles à résoudre de manière optimale. Cependant, au cours des dernières années, les métaheuristiques, capables de trouver rapidement des solutions optimales, sont devenues populaires et ont été implémentées dans des approches basées sur l'optimisation qui peuvent être utilisées en électronique pour dimensionner des circuits analogiques.

Cette thèse se concentre sur le développement et l'adaptation d'algorithmes et d'outils d'optimisation multi-objectifs qui peuvent gérer simultanément l'optimisation basée sur plusieurs critères pour la conception optimale de circuits intégrés analogiques. Dans ce cadre général, nous proposons alors une adaptation d'un algorithme d'optimisation inspiré du comportement des essaims de papillons, à la fois en version mono-objectif et multi-objectif, pour l'automatisation de la conception de circuits analogiques. Les deux versions sont testées sur deux circuits analogiques, un amplificateur opérationnel à deux étages et un convoyeur de courant de seconde génération. Comparés aux méthodes les plus connues, les algorithmes proposés gardent plusieurs de leurs avantages, surtout en terme de robustesse de la convergence. Nous avons également proposé une méthode basée sur la simulation pour générer les compromis de fonctionnement des circuits analogiques à différents régimes de fonctionnement des transistors, correspondant à différentes applications, comme les applications biomédicales et à large bande passante. Dans la deuxième partie, nous proposons un cadre basé sur la simulation assistée par des réseaux de neurones profonds conçus pour optimiser de multiples métriques pour les circuits analogiques. Cet environnement d'optimisation, qui comprend un moteur d'optimisation globale basé sur un algorithme évolutionnaire et un moteur d'optimisation locale basé sur un réseau de neurones profonds, permet des hybridations entre les algorithmes évolutionnaires, les approches d'optimisation, les simulateurs de circuits et les réseaux neuronaux profonds. Ce cadre est alors implémenté et testé pour dimensionner quatre circuits analogiques. Ses principales caractéristiques sont le faible coût d'adaptation et d'utilisation, une forte capacité à gérer des spécifications strictes en utilisant un modèle de réseaux de neurones profonds pour la prédiction de chaque performance, ainsi qu'une forte capacité d'optimisation pour atteindre des performances élevées. Contrairement aux techniques classiques, l'efficacité de l'environnement que nous proposons est mise en avant grâce à une réduction importante des temps de calculs, principalement due à la remarquable robustesse des prédictions des réseaux de neurones profonds.

Mots-clés : Optimisation multi-objectifs ; Métaheuristique ; Front de Pareto ; Réseau de neurones artificiels ; Métriques de performance ; Conception automatisée de circuits ; Amplificateurs CMOS ; Convoyeur de courant ; Layout.

Contents

Table des Matières IV
Liste des Abréviations V

Introduction générale **1**

I Etat de l'art de l'optimisation **7**

1 Optimisation et Métaheuristiques **9**
1 Introduction 9
2 Optimisation: concepts et définitions 9
2.1 Optimisation Mono- and Multi-Objectifs 10
2.2 MOWBOA utilisant la technique de Tchebysheff 10
3 Métaheuristique 11
4 Conclusion 12

2 Réseaux de neurones et leur application **13**
1 Introduction 13
2 Définition de l'apprentissage profond 14
3 Réseaux de neurones : définition et concepts 14
3.1 Neurone Biologique 15
3.2 Perceptron à Couche Unique 15
3.3 Fondamentaux des réseaux de neurones 15
3.4 Fonctions d'Activation 16
4 Entraînement des Réseaux de Neurones 16
4.1 Descente de Gradient 17
4.2 Algorithme de Rétropropagation 17
5 Conclusion 18

II Contributions **19**

3 Adaptation de BOA pour la Conception de Circuits Analogiques **21**
1 Introduction 21
2 MOWBOA pour la dimensionnement des CIs 21
3 Dimensionnement optimal du CCII 22
4 Conclusion 27

4 Méthode multi-objectifs basée sur la simulation pour la conception des circuits analogiques. 29
1 Introduction . . . 29
2 Approche d'optimisation multi-objectif basée sur la simulation . . . 29
3 Application: dimensionnement de CCII . . . 31
4 Conclusion . . . 35

5 Optimisation assistée par les réseaux de neurones profonds 37
1 Introduction . . . 37
2 Méthode d'optimisation assistée par DNN . . . 37
3 Application: Op-Amp Cascode Replié . . . 39
4 Conclusion . . . 43

6 Layout du CCII : Post-layout Simulations et Discussions 45
1 Introduction . . . 45
2 Création du Layout de CCII . . . 47
3 Simulation post-layout et discussions . . . 49
4 Conclusion . . . 50

Bibliography 53

Liste des Abréviations

ACAA	Algorithme de colonie d'abeilles artificielle
AE	Algorithme Évolutionnaire
AMS	Analogique et Mixte
ACO	Optimisation par colonies de fourmis
IA	Intelligence Artificielle
AMS	Analogique et à Signal Mixte
ANN	Réseau de Neurones Artificiels
BOA	Algorithme d'optimisation du papillon
CI	Circuit Intégré
CCII	Convoyeur de Courant de deuxième génération
CD	Distance d'encombrement
CFOA	Amplificateur Opérationnel à Rétroaction de Courant
CMOS	Semi-conducteur à oxyde de métal complémentaire
DL	Apprentissage Profond
DNN	Réseau de neurones profond
DRC	Design Rules Check
EKV	Enz, Krummenacher, Vittoz
FoM	Figure de Mérite
GA	Algorithme génétique
GCN	Réseaux de neurones à convolution de graphe
GP	Processus Gaussien
HCCII	Convoyeur de courant de deuxième génération à haute précision
IC	Circuit Intégré
IP	Propriété Intellectuelle
LVS	Layout Versus Schematic
ML	Apprentissage Automatique
MOPSO	Optimisation par essaims particulaires multi-objectif
MOS	Semiconducteur Métal-Oxyde
MOWBOA	Algorithme Multi-Objectif d'Optimisation des Papillons Pondérés min-max
NSGA	Algorithme génétique de tri non dominé
Op-Amp	Amplificateur Opérationnel
OTA	Amplificateur Opérationnel de Transconductance
PSO	Optimisation par essaim de particules
PVT	Processus, Tension et Température
RL	Apprentissage par Renforcement
SGD	Descente de gradient stochastique
SLP	Perceptron à Couche Unique
SVM	Machine à Vecteurs de Support

Introduction générale

Un voyage de mille lieues commence toujours par un premier pas.

– Lao-Tzu

Contexte et motivations

Aujourd'hui, les circuits analogiques constituent une part importante des puces utilisées dans les dispositifs de communication, les équipements médicaux portables et les véhicules électriques [Kumar et al. 2018] ... Grâce à l'utilisation croissante des *Intellectual Properties (**IPs**)* analogiques dans les conceptions de système sur une puce (System on Chip), on s'attend à ce que cette tendance continue de croître dans le futur [Ahmadyan 2016].

Compte tenu de la complexité des *Circuit Intégré (**CI**)*s utilisés dans les dispositifs modernes, les méthodologies traditionnelles de conception et de validation basées sur des simulations statistiques (Monte Carlo) sont inefficaces, coûteuses, sujettes à l'erreur. Elles sont bien souvent trompeuses lorsqu'il s'agit de prédire des comportements inconnus, les pires coins (worst corners) et de stresser les composants faibles du circuit. Pour répondre aux exigences du marché, il est essentiel d'optimiser et d'automatiser la conception des circuits analogiques.

Les problèmes d'optimisation consistent toujours à maximiser ou à minimiser une certaine quantité, on les retrouve partout. De nombreux problèmes du monde réel, notamment la gestion du temps, le routage des véhicules ou la conception de systèmes électroniques, peuvent être résolus en choisissant la meilleure solution parmi un grand nombre de solutions possibles. Pour de nombreux problèmes d'optimisation, si l'évaluation d'une seule solution est généralement assez rapide et facile, leur optimisation nécessite des calculs beaucoup plus difficiles dans la mesure où leur complexité augmente de manière significative avec la taille du problème. La question de savoir si ces problèmes difficiles, connus sous le nom de NP-difficile[1] peuvent être théoriquement traités efficacement reste une question ouverte en informatique.

Ces problèmes sont difficiles à résoudre de façon optimale dans un temps d'exécution raisonnable par les méthodes classiques. Pour obtenir de bonnes solutions à ce type de problèmes et aux problèmes d'optimisation à grande échelle, de nouvelles techniques d'approximation comme les métaheuristiques ont été développées et sont généralement préférées. En effet, les métaheuristiques ont été appliquées récemment avec succès dans le domaine de l'électronique, où la complexité de la réalisation des **CI**s augmente de jour en jour, rendant la création de designs impossible avec les techniques traditionnelles.

[1] Problème difficile non déterministe en temps polynomial

Contrairement aux méthodes de recherche locale qui se concentrent sur la structure du problème à résoudre, les métaheuristiques produisent généralement des solutions de manière aléatoire et s'approchent des solutions optimales de manière progressive et itérative. Ces méthodes ont fait leurs preuves, qu'elles soient utilisées comme des algorithmes autonomes ou hybridées dans des cadres ou des outils complexes. Bien qu'il soit prévisible qu'aucun algorithme ne soit le meilleur pour résoudre tous les problèmes, les métaheuristiques impliquent généralement un large spectre de variations potentielles qui utilisent une variété d'approches alternatives afin d'améliorer leurs performances pour un problème particulier.

Par ailleurs, les problèmes d'optimisation peuvent impliquer plusieurs critères à optimiser. Par exemple, si un coût doit être minimisé ou une qualité doit être maximisée, le processus d'optimisation aboutit généralement à une seule solution optimale. Par contre, la prise en compte de plusieurs objectifs implique généralement un contexte considérablement plus riche où plusieurs solutions de compromis incomparables doivent être explorées. La prise en compte des concepts multi-objectifs pour les défis de conception d'algorithmes est toute récente, même si les problèmes multi-objectifs traditionnels sont de plus en plus étudiés et appréhendés.

Travaux dans la même thèmatique

Pour remédier au processus à forte intensité de calcul et à la réduction du temps de mise sur le marché, des approches basées sur l'optimisation pour l'automatisation du dimensionnement des circuits analogiques ont été développées au cours des dernières décennies. Ces approches utilisent un processus d'optimisation itératif au lieu d'un plan de circuit pré-conçu où les paramètres de conception sont mis à jour de manière itérative. Les moteurs d'optimisation et d'évaluation sont les principaux composants de ces approches. Deux types de méthodes peuvent être classés en fonction du moteur d'évaluation : les méthodes basées sur les modèles et les méthodes basées sur les simulations. [S. Boyd et al. 2007; Jafari et al. 2010; Lin and P. Li 2015; Sallem et al. 2013; Liu, Fernández, et al. 2009; Tlelo-Cuautle et al. 2020; Castañeda-Aviña et al. 2020].

Dans le cas de l'optimisation basée sur les modèles, les concepteurs encodent leurs connaissances et leur expérience pour formuler les performances des **CI**s analogiques en termes de paramètres de conception et de contraintes imposées. Les algorithmes d'optimisation tels que la Programmation Géométrique [S. P. Boyd, Lee, et al. 2001], l'*Optimisation par essaim de particules (**PSO**)* [Fakhfakh et al. 2010], l'*Optimisation par colonies de fourmis (**ACO**)* [Akbari, Shokouhifar, et al. 2016], ou l'algorithme *Algorithme d'optimisation du papillon (**BOA**)* [Arora and Singh 2019] sont ensuite appliquées aux problèmes formulés pour obtenir une performance optimale. Bien que ces approches soient généralement rapides, le développement de formules précises pour les circuits est difficile et peut s'écarter largement des valeurs réelles.

Par contre, les approches basées sur la simulation considèrent les caractéristiques des circuits comme des fonctions de boîte noire. Elles effectuent des explorations guidées dans l'espace de recherche et cherchent à atteindre un minimum global en utilisant des algorithmes et des simulateurs de circuits.

D'autre part, les méthodes basées sur l'apprentissage et les méthodes basées sur des modèles de substitution sont récemment devenues des approches permettant d'explorer de façon efficace l'espace des solutions [Cao et al. 2022; Lyu et al. 2017; Liu, D. Zhao, et al.

2014; Liu et al. 2013; Kotti et al. 2018; Garbaya, Kotti, Fakhfakh, et al. 2020; Garbaya, Kotti, Drira, et al. 2018; Lourenço et al. 2019]. Les techniques d'apprentissage automatique sont souvent utilisées pour construire des modèles de substitution, qui sont des modèles d'approximation des simulations relativement peu coûteux en termes de temps de calcul. Elles sont utilisées en optimisation pour réduire les coûts et le temps de calcul en remplaçant les simulations. Ainsi, les auteurs de [Jafari et al. 2010] et de [Lin and P. Li 2015] proposent des mécanismes d'apprentissage basés sur les stratégies d'apprentissage, les réseaux neuronaux et l'apprentissage automatique basé sur *Machine à Vecteurs de Support (**SVM**)*. Cependant, le temps de préparation peut être plus long en raison de la nécessité d'évaluer les échantillons d'entraînement. Dans [Castañeda-Aviña et al. 2020], une optimisation mono-objectif sous contraintes est proposée afin que le circuit optimisé soit robuste aux variations de *Processus, Tension et Température (**PVT**)* et aux simulations de Monte Carlo. Dans [Tlelo-Cuautle et al. 2020] les autheurs proposent une procédure qui traite les contraintes pour s'assurer que les transistors sont au point de fonctionnement approprié, cette procédure est appliquée et testée pour la conception automatique de *Amplificateur Opérationnel de Transconductance (**OTA**)* et d'**OTA** cascode plié recyclé. Lyu et al. 2017 proposent une méthodologie hybride qui associe les deux techniques, celle basée sur les modèles et celle basée sur les simulations. Des modèles évolutifs en ligne sont utilisés au lieu de modèles hors ligne pré-construits. Les simulations de circuits sont invoquées pendant la procédure d'optimisation pour mettre à jour les modèles en ligne de manière progressive. Dans cette méthode, le *Processus Gaussien (**GP**)*, également connu sous le nom de kriging, est utilisé comme modèle de substitution. Le **GP** fournit non seulement des valeurs de prédiction mais aussi une erreur quadratique moyenne. Une fonction d'amélioration attendue (Expected Improvement) est proposée dans le but d'équilibrer l'exploration et l'exploitation.

Les auteurs de [Y. Li et al. 2019], mettent en œuvre une approche basée sur le modèle de substitution et des calculs parallèles dans la phase de recherche locale, puis un algorithme parallèle est utilisé dans la phase d'exploration globale. Étant donné que le dimensionnement parallèle des **CI**s analogiques est un domaine de recherche (par exemple, les simulations parallèles de différentes conceptions), qui dépend du problème particulier et de la capacité de calcul, il pourra être traité séparément et sort du cadre de ce manuscrit. En général, la régression **GP** est utilisée dans les méthodes de substitution pour la modélisation de l'espace de conception en raison de sa plus grande capacité d'apprentissage comparé aux autres alternatives [Shadab et al. 2022; S. Zhang et al. 2019; He et al. 2020; Liu, Deferm, et al. 2012]. Par conséquent, les méthodes typiques de synthèse et de dimensionnement des **CI**s analogiques utilisent **GP**.

Les méthodes de *Apprentissage par Renforcement (**RL**)* ont récemment été appliquées dans ce domaine en tant qu'algorithmes basés sur l'apprentissage. Les auteurs de [H. Wang et al. 2020] présentent le *Réseaux de neurones à convolution de graphe (**GCN**)*-**RL**, qui exploite le *Réseaux de neurones graphiques (**GNNs**)* et offre un cadre transférable. Bien que cette méthode donne de meilleurs résultats, elle exige des milliers de simulations pour converger et elle souffre de la difficulté à déterminer le choix de l'architecture, le vecteur d'observation et le système de récompense. Dans [Settaluri et al. 2020], AutoCkt est présenté comme un algorithme d'apprentissage par renforcement à sous-échantillonnage pour optimiser les performances des circuits en fonction des mesures discrètes prises dans l'espace de recherche. Il est plus efficace que l'algorithme d'évolution différentielle et les agents aléatoires **RL**. Malgré cela, il nécessite aussi un entraînement avec des milliers de simulations avant d'être déployé, ce qui est très coûteux.

Objectif

L'objectif de cette thèse est d'étudier les différents algorithmes d'optimisation multi-objectifs et leurs adaptations pour résoudre les problèmes de dimensionnement des circuits intégrés analogiques. Nous élaborons ainsi des méthodes efficaces pour concevoir, modéliser et optimiser les circuits intégrés analogiques. Ces méthodes permettent de dimensionner automatiquement les circuits analogiques tout en respectant les contraintes technologiques de conceptions. Sur la base des points mentionnés ci-dessus, nous proposons ainsi un environnement d'optimisation pour pallier aux inconvénients déjà rencontrés dans ce domaine. Cet environnement utilise les algorithmes évolutionnaires, les techniques d'optimisation, l'automatisation et les algorithmes de réseaux neuronaux profonds pour fournir des outils d'automatisation dans le contexte de l'optimisation multi-objectifs.

Contributions et Structure

Ce résumé de thèse reflète nos contributions en deux parties : La première partie (chapitre 1 et 2) aborde l'état de l'art des approches en optimisation. La deuxième partie (chapitre 3 à 6) présente un résumé de nos principales contributions pour le dimensionnement automatisé des circuits analogiques. Nous concluons le manuscrit avec les conclusions et les perspectives du travail réalisé.

Dans le Chapitre 1, nous définissons et discutons de l'optimisation mono-objectif et multi-objectif. Ensuite, nous présentons brièvement une description des métaheuristiques les plus populaires développées pour résoudre des problèmes mono-objectif et multi-objectif.

Dans le Chapitre 2, nous abordons les réseaux de neurones et leurs définitions fondamentales. Ensuite, nous décrivons les concepts généraux de l'apprentissage profond qui seront utilisés ultérieurement dans le contexte de l'optimisation.

Dans le chapitre 3, nous présentons un processus d'optimisation basé sur le modèle à petits signaux du circuit, utilisant un algorithme multi-objectifs appelé *Algorithme Multi-Objectif d'Optimisation des Papillons Pondérés min-max (**MOWBOA**)* pour l'optimisation multicritères de circuits analogiques. À titre d'exemples d'application, un *Convoyeur de Courant de deuxième génération (**CCII**)* et un *Amplificateur Opérationnel (**Op-Amp**)* deux étages ont été optimisés afin d'évaluer l'algorithme proposé. Dans ce résumé, nous ne présentons que les résultats du **CCII**. Plusieurs expériences d'optimisation ont été effectuées afin d'étudier l'effet de la longueur du canal des transistors et du courant de polarisation sur sa performance principale, ainsi que pour tester et évaluer la robustesse de l'algorithme proposé.

Dans le chapitre 4, nous présentons une méthode basée sur la simulation visant à générer l'ensemble non dominé des fronts de Pareto d'un circuit pour différentes applications et à différents régimes de fonctionnement des transistors, à savoir les régions de faible inversion, linéaire et de saturation. Ce cadre d'optimisation est proposé pour résoudre et surmonter les difficultés de modélisation des circuits analogiques dans différents régimes. Une attention particulière est accordée à l'algorithme *Optimisation par essaims particulaires multi-objectif (**MOPSO**)-Distance d'encombrement (**CD**)*, qui est utilisé dans ce contexte basé sur la simulation, en tant que métaheuristique la plus appropriée à appliquer dans le domaine du dimensionnement des circuits analogiques. Cette approche est testée pour la conception de quatre circuits analogiques, à savoir deux convoyeurs de

courant différents, un *Amplificateur Opérationnel à Rétroaction de Courant (**CFOA**)* et un amplificateur à transconductance **OTA**. Dans ce résumé, nous présentons les résultats du **CCII**.

Dans le chapitre 5, nous avons franchi une étape en proposant un environnement d'optimisation basé sur la simulation assistée par le *Réseau de neurones profond (**DNN**)* pour le dimensionnement automatisé de blocs de construction analogiques haute performance. L'environnement proposé se compose d'un moteur d'optimisation globale basé sur un *Algorithme Évolutionnaire (**AE**)* et d'un moteur d'optimisation locale basé sur les **DNNs**. Cet environnement est implanté et testé sur quatre circuits analogiques. Ses principales caractéristiques sont un coût de mise en œuvre réduit, une grande capacité à traiter des spécifications strictes grâce aux modèles **DNN** qui permettent de prédire chaque caractéristique du circuit, ainsi qu'une forte capacité d'optimisation qui conduit à des performances élevées. De même, nous présentons ici les résultats du **Op-Amp** cascode Replié.

Dans le Chapitre 6, nous rappelons les principales tâches de conception de *layout*, abordant les défis spécifiques rencontrés dans la conception de *layout* analogique. Nous présentons le *layout* du **CCII** prédimensionnés dans le chapitre 3. Ensuite, nous procédons à l'extraction des parasites et réalisons des simulations post-*layout* pour évaluer l'effet des extractions parasitaires sur les performances du circuit. Nos discussions approfondies évoluent vers une interprétation des résultats de simulation, fournissant de précieuses indications sur l'efficacité des méthodes proposées.

Enfin, ce manuscrit se termine par des conclusions et des perspectives sur les travaux effectués, ainsi que par quelques annexes qui fournissent des informations supplémentaires.

Publications

Les articles suivants ont été, par ordre chronologique, acceptés et ont été présentés dans des conférences internationales au cours de cette thèse :

Articles de journaux :

- **Lberni**, **Abdelaziz**, Marktani, Malika ALAMI, Ahaitouf, Abdelaziz, Ahaitouf, Ali. Analog Circuit Sizing Based on Evolutionary Algorithms and Deep Learning. **Expert Systems With Applications**, **2023**, p. 121480. [2]

- **Lberni**, **Abdelaziz**, Sallem, Amin, Marktani, Malika Alami, et al. Influence of the operating regimes of mos transistors on the sizing and optimization of cmos analog integrated circuits. **AEU-International Journal of Electronics and Communications**, **2022**, vol. 143, p. 154023. [3]

- **Lberni**, **Abdelaziz**, Marktani, Malika Alami, Ahaitouf, Abdelaziz, Ahaitouf, Ali. Whale Inspired Optimization Algorithm for Optimal Design of Low Voltage Amplifier. **Statistics, Optimization & Information Computing**, **2022**, vol. 10, no 1, p. 135-146. [4]

[2] Accessible sur https://doi.org/10.1016/j.eswa.2023.121480
[3] Accessible sur https://doi.org/10.1016/j.aeue.2021.154023
[4] Accessible sur https://doi.org/10.19139/soic-2310-5070-1204

- **Lberni**, **Abdelaziz**, Marktani, Malika Alami, Ahaitouf, Abdelaziz, Ahaitouf, Ali. Efficient butterfly inspired optimization algorithm for analog circuits design. **Microelectronics Journal**, **2021**, vol. 113, p. 105078. [5]
- **Lberni**, **Abdelaziz**, Marktani, Malika Alami, Ahaitouf, Abdelaziz, Ahaitouf, Ali. An efficient optimisation-based design of current conveyor performances. **International Journal of Computer Aided Engineering and Technology**, **2021**, vol. 18, no 1-3, p. 167-180. [6]

Articles de conférence :

- **Lberni**, **Abdelaziz**, Marktani, Malika ALAMI, Ahaitouf, Abdelaziz, Ahaitouf, Ali. 1-V 87-nW CMOS Rail-to-Rail Amplifier Using an Optimization methodology. In : **The 34th International Conference on Microelectronics (ICM)**. IEEE, **2022**. p. 54-57. [7]
- **Lberni**, **Abdelaziz**, Marktani, Malika Alami, Ahaitouf, Abdelaziz, Ahaitouf, Ali. Bulk-driven current conveyor optimization using simulation-based method. In : **E3S Web of Conferences**. EDP Sciences, **2022**. p. 01017. [8]
- **Lberni**, **Abdelaziz**, Sallem, Amin, Marktani, Malika Alami, Ahaitouf, Ali. An efficient multi-objective simulation-based approach for low voltage low power analog ics. In : **2020 IEEE International Conference on Design & Test of Integrated Micro & Nano-Systems (DTS)**. IEEE, **2020**. p. 1-5. [9]
- **Lberni**, **Abdelaziz**, Marktani, Malika Alami, Ahaitouf, Abdelaziz, Ahaitouf, Ali. Adaptation of the whale optimization algorithm to the optimal sizing of analog integrated circuit: Low voltage amplifier performances. In : **2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)**. IEEE, **2020**. p. 1-6. [10]
- **Lberni**, **Abdelaziz**, Ahaitouf, Abdelaziz, Marktani, Malika Alami, Ahaitouf, Ali. Sizing of second generation current conveyor using evolutionary algorithms. In : **2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)**. IEEE, **2019**. p. 1-5. [11]

Chapitres de livres :

- **Lberni**, **Abdelaziz**, Sallem, Amin, Marktani, Malika Alami, et al. Simulation-based optimization for automated design of analog/rf circuits. In : **WITS 2020**. Springer, Singapore, **2022**. p. 389-399. [12]
- **Lberni**, **Abdelaziz**, Marktani, Malika ALAMI, Ahaitouf, Abdelaziz, Ahaitouf, Ali. Application of hpsgwo to the optimal sizing of analog active filter. In : **International Conference on Electronic Engineering and Renewable Energy**. Springer, Singapore, **2020**. p. 309-315. [13]

[5] Accessible sur https://doi.org/10.1016/j.mejo.2021.105078
[6] Accessible sur https://doi.org/10.1504/IJCAET.2023.127794
[7] Accessible sur https://ieeexplore.ieee.org/document/10005382
[8] Accessible sur https://doi.org/10.1051/e3sconf/202235101017
[9] Accessible sur https://ieeexplore.ieee.org/document/9196166
[10] Accessible sur https://ieeexplore.ieee.org/document/9314428
[11] Accessible sur https://ieeexplore.ieee.org/document/9068896
[12] Accessible sur https://doi.org/10.1007/978-981-33-6893-4_37
[13] Accessible sur https://doi.org/10.1007/978-981-15-6259-4_32

Part I

Etat de l'art de l'optimisation

Chapter

1 Optimisation et Métaheuristiques

Chapter content

1 Introduction 9

2 Optimisation: concepts et définitions 9

2.1 Optimisation Mono- and Multi-Objectifs 10

2.2 MOWBOA utilisant la technique de Tchebysheff 10

3 Métaheuristique 11

4 Conclusion 12

1 Introduction

Dans ce chapitre, nous présentons l'optimisation mono- et multi-objectif. Ensuite, nous introduisons pour la première fois un algorithme d'optimisation multi-objectif utilisant la technique de Tchebychev, pondérée inspiré du papillon pour l'optimisation de la conception des circuits analogiques. Enfin, nous proposons une brève description des métaheuristiques les plus populaires développées pour résoudre des problèmes mono-objectif et multi-objectif.

2 Optimisation: concepts et définitions

L'optimisation est le processus qui consiste à trouver une ou plusieurs variables, maximisant ou minimisant une certaine fonction ou un ensemble de fonctions. On distingue donc deux types de problèmes d'optimisation : les problèmes d'optimisation mono-objectif et multi-objectifs.

Les méthodes d'optimisation les plus utilisées sont les techniques de gradient, qui estiment le gradient de la fonction objectif afin de déterminer une direction de recherche [Kaveh 2014]. Comparées aux algorithmes stochastiques, les techniques de gradient fournissent des solutions efficaces pour les problèmes convexes. Cependant, elles ne peuvent pas résoudre facilement les problèmes non convexes. D'autres algorithmes, connues sous le nom de métaheuristiques, sont des méthodes stochastiques. Ces approches sont très appropriées pour la recherche globale en raison de leur capacité à explorer l'espace de

recherche en un temps de calcul raisonnable. Elles donnent de bons résultats pour la plupart des problèmes d'optimisation dans différents domaines. Elles sont largement utilisées dans la conception et l'optimisation de divers systèmes, notamment en génie électrique, mécanique, industriel et civil.

2.1 Optimisation Mono- and Multi-Objectifs

Dans l'optimisation mono-objectif (cas où n=1), on cherche soit le maximum, soit le minimum d'une fonction f, dans un espace de recherche bien défini Ω. Le problème d'optimisation est défini par :

$$\begin{array}{lll} \text{minimize} & \text{f}(x)=(f_1(x),...,f_n(x)), & x \in \Omega \\ \text{subject to} & g_j(x) \leq 0 & \text{pour } j = 1,\ldots,m \\ & h_k(x) = 0 & \text{pour } k = 1,\ldots,p, \\ & x_{li} \leq x_i \leq x_{ui} & \text{pour } i = 1,\ldots,D \end{array} \tag{1.1}$$

Où $x \in \mathbf{R}^D$ représente les D variables de conception, où chaque variable x_i est limitée par une borne inférieure, x_{li}, et supérieure, x_{ui}. $g(x)$ et $h(x)$ sont respectivement des contraintes d'inégalité et d'égalité. Pour simplifier, dans ce document, nous aborderons les problèmes d'optimisation sous l'aspect minimisation. Ainsi, maximiser une fonction f est équivalent à minimiser -f, $maximiser(f(x)) \iff minimiser(-f(x))$.

Dans l'optimisation multi-objectifs (cas où $n \geq 2$), Le but est de minimiser l'ensemble des objectifs f_i. L'objectif serait de chercher le vecteur x qui optimise chacune des fonctions objectif.

2.2 MOWBOA utilisant la technique de Tchebysheff

MOWBOA est un algorithme proposé pour les problèmes d'optimisation multi-objectifs. Cet algorithme utilise la technique de Tchebysheff pondérée. D'abord, nous transformons le problème multi-objectif en un problème pondéré de type min-max (ou technique de Tchebysheff pondérée) : [Lberni et al. 2021; Ma et al. 2017].

$$\begin{array}{lll} \text{minimize} & \max\limits_{i\in\{1,\ldots,n\}} \{\omega_i | f_i(x) - f_i^\circ|\}, & x \in \Omega \\ \text{subject to} & g_j(x) \leq 0 & \text{pour } j = 1,\ldots,m \\ & h_k(x) = 0 & \text{pour } k = 1,\ldots,p \end{array} \tag{1.2}$$

où $\omega_i \in \mathbf{R}^n$ sont les poids, typiquement $\omega_i > 0$ et $\sum_{i=1}^n \omega_i = 1$. f_i° est le *point d'utopie*, c'est la valeur souhaitée de l'objectif i^{th}. Dans la conception des circuits analogiques, le *point d'utopie* peut être fixé par les décideurs (spécialistes de la conception analogique). Ici, chaque f_i° est déterminé par l'algorithme standard **BOA** en utilisant l'optimisation mono-objectif.

En fait, chaque solution de (1.2) est une solution optimale de Pareto du problème multi-objectif original (1.1). Par conséquent, l'ensemble optimal de Pareto peut être obtenu en variant les poids ω_i. En utilisant des variations spécifiques des poids (par $1/N_{\text{PF}}$, N_{PF} est la taille du front de Pareto), on peut obtenir un front de Pareto complet.

D'après [Qingfu Zhang and H. Li 2007], la méthode de Tchebycheff peut fournir de bons résultats, mais elle présente un inconvénient dans sa fonction de coût total, qui n'est

pas lisse (elle a une forme min-max). Par conséquent, nous introduisons une nouvelle technique pour traiter ce problème, en ajoutant un nouveau paramètre inconnu, λ :

$$\begin{aligned} &\text{minimize} && \lambda \\ &\text{subject to} && \max_{i\in\{1,\dots,n\}} \{\omega_i|f_i(x) - f_i^\circ|\} \leq \lambda \\ & && g_j(x) \leq 0 \quad \text{pour } j = 1,\dots,m \\ & && h_k(x) = 0 \quad \text{pour } k = 1,\dots,p \end{aligned} \tag{1.3}$$

Il est clair que $\max\{\omega_i|f_i(x) - f_i^\circ|\} \leq \lambda$ est équivalent à $\omega_i|f_i(x) - f_i^\circ| \leq \lambda$, $\forall\omega_i$ pour $i \in \{1, ..., n\}$ et $x \in S$.

Le problème de l'équation (1.3) peut être écrit comme suit :

$$\begin{aligned} &\text{minimize} && \lambda \\ &\text{subject to} && \omega_i|f_i(x) - f_i^\circ| - \lambda \leq \lambda \\ & && g_j(x) \leq 0 \quad \text{pour } j = 1,\dots,m \\ & && h_k(x) = 0 \quad \text{pour } k = 1,\dots,p \end{aligned} \tag{1.4}$$

3 Métaheuristique

Les métaheuristiques sont des techniques d'optimisation inspirées de processus naturels ou de comportements d'organismes sociaux. Elles sont utilisées pour résoudre une grande diversité de problèmes complexes. Dans ce contexte, nous allons donner une petite description de quleques métaheuristiques que nous avons utilisé dans notre travail, à savoir l'*Algorithme génétique (**GA**)*, l'acnsga, l'**PSO**, l'*Algorithme de colonie d'abeilles artificielle (**ACAA**)* et l'**BOA** (leurs descriptions détaillées sont présentées dans le rapport de thèse). Chacune de ces approches présente des caractéristiques distinctes et des domaines d'application spécifiques, offrant ainsi une gamme d'outils puissants pour résoudre des problèmes d'optimisation variés.

- Algorithme génétique :

  GA est une métaheuristique basée sur l'évolution biologique. Il opère sur une population de solutions candidates, lesquelles sont représentées sous forme de chromosomes. À chaque génération, une sélection, un croisement et une mutation sont appliqués pour faire évoluer la population vers une solution optimale. Cette méthode a trouvé de nombreuses applications dans des domaines tels que l'optimisation combinatoire, l'apprentissage automatique et la résolution de problèmes complexes [Choi et al. 2015].

- Algorithme génétique de tri non dominé :

  *Algorithme génétique de tri non dominé (**NSGA**)* est spécifiquement conçu pour résoudre des problèmes d'optimisation multi-objectif. Il classe les solutions en plusieurs niveaux de dominance, permettant ainsi de trouver un ensemble de solutions optimales qui ne sont pas dominées les unes par les autres. Cette caractéristique en fait un outil précieux pour les problèmes où plusieurs critères de performance sont en jeu [S. Zhao et al. 2005].

- Algorithme génétique de tri non dominé-II :

  NSGA-II est une amélioration du **NSGA** original. Il introduit des opérations de sélection plus efficaces et une stratégie de conservation des solutions non dominées pour améliorer la convergence et la diversité de l'ensemble de solutions. Cette amélioration renforce la capacité de l'algorithme à trouver un ensemble de solutions optimales pour des problèmes multi-objectifs [Kchaou et al. 2016].

- Optimisation par essaim de particules :

  PSO s'inspire du comportement de coopération des oiseaux ou des poissons en groupe. Il utilise un ensemble de "particules" qui se déplacent dans l'espace de recherche pour trouver la meilleure solution. Cette approche est particulièrement efficace pour les problèmes continus et non linéaires [Tlelo-Cuautle et al. 2020].

- Algorithme de colonie d'abeilles artificielle :

  L'**ACAA** s'inspire du comportement des abeilles dans la recherche de nourriture. Il utilise une population d'abeilles pour explorer l'espace de recherche et trouver la meilleure solution. Cette métaheuristique est souvent utilisée pour les problèmes d'optimisation stochastiques et non linéaires [Patel and Thakker 2018].

- Algorithme d'optimisation du papillon :

  L'**BOA** s'inspire du mouvement de recherche des papillons dans leur environnement. Il utilise des opérations de vol et de recherche pour explorer l'espace de recherche et trouver la solution optimale. Bien que moins courant que d'autres métaheuristiques, il offre une approche intéressante pour certains types de problèmes d'optimisation [Lberni et al. 2021].

Chacune de ces métaheuristiques présente des avantages et des limites, et le choix de la méthode dépendra du type de problème à résoudre et des contraintes spécifiques qui lui sont associées.

4 Conclusion

Dans ce chapitre, nous avons fourni une vue d'ensemble des techniques d'optimisation, tant pour l'optimisation mono-objectif que multi-objectif. Ensuite, nous avons présenté l'algorithme **MOWBOA** utilisant la technique de Tchebychev. Enfin, le chapitre se conclut en abordant également les métaheuristiques largement utilisées dans le domaine de l'électronique.

Chapter

2 Réseaux de neurones et leur application

Chapter content

1 Introduction 13
2 Définition de l'apprentissage profond 14
3 Réseaux de neurones : définition et concepts 14
3.1 Neurone Biologique 15
3.2 Perceptron à Couche Unique 15
3.3 Fondamentaux des réseaux de neurones 15
3.4 Fonctions d'Activation 16
4 Entraînement des Réseaux de Neurones 16
4.1 Descente de Gradient 17
4.2 Algorithme de Rétropropagation 17
5 Conclusion 18

1 Introduction

Au cours des dernières années, l'*Apprentissage Profond (**DL**)* s'est imposé comme un outil puissant pour résoudre des problèmes complexes dans divers domaines. Le **DL** est une branche de l'*Apprentissage Automatique (**ML**)* basée sur les *Réseaux de Neurones Artificiels (**ANNs**)* capables d'apprendre des motifs complexes et des relations dans les données. Contrairement aux approches de programmation traditionnelles, les algorithmes de **DL** apprennent directement à partir des données, leur permettant de reconnaître des motifs et de faire des prédictions sans avoir besoin de programmation explicite.

Ce chapitre présente un résumé sur les réseaux de neurones, en commençant par une introduction au concept et à la définition des **ANNs**. Nous abordons également les

techniques d'entraînement des réseaux de neurones, y compris l'algorithme de descente de gradient et l'algorithme de rétropropagation.

2 Définition de l'apprentissage profond

ML désigne le processus d'entraînement des ordinateurs à apprendre à partir de données et à améliorer leurs performances sur des tâches spécifiques sans être explicitement programmés. Cela implique l'utilisation d'algorithmes pour identifier des motifs dans les données et réaliser des prédictions ou prendre des décisions basées sur ces motifs.

DL est une sous-catégorie de l'apprentissage automatique qui implique l'utilisation des **ANNs** avec de multiples couches pour résoudre des problèmes complexes. Ces réseaux sont modélisés sur la structure du cerveau humain, ce qui leur permet de traiter de grandes quantités de données. En analysant ces ensembles de données, les algorithmes de **DL** peuvent identifier des motifs et des relations qui seraient difficiles voire impossibles pour les humains à détecter par eux-mêmes. La relation entre *Intelligence Artificielle (**IA**)*, **ML** et **DL** est illustrée dans la Figure 2.1.

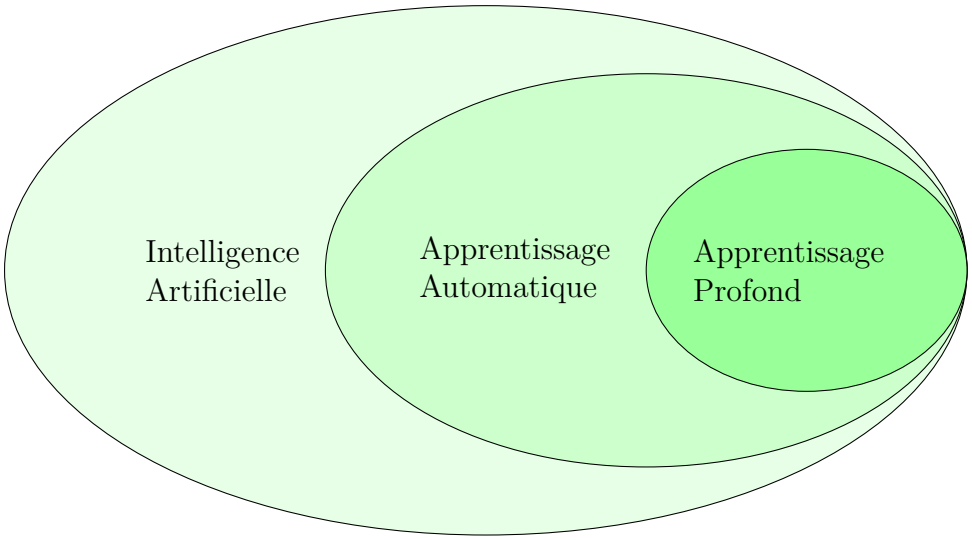


Figure 2.1: *Relation entre l'Intelligence Artificielle, l'Apprentissage Automatique et l'apprentissage profond.*

3 Réseaux de neurones : définition et concepts

Les **ANNs** sont un type de modèle d'apprentissage automatique qui imite le traitement de l'information par le cerveau humain. Contrairement aux ordinateurs numériques conventionnels, qui fonctionnent de manière linéaire et séquentielle, le cerveau humain est un système complexe, non linéaire et parallèle capable d'effectuer des calculs beaucoup plus rapidement. Les **ANNs** tentent de reproduire ce comportement en utilisant des éléments de traitement interconnectés appelés neurones, qui travaillent ensemble pour résoudre des problèmes complexes [Gurney 2018].

3.1 Neurone Biologique

Tous les animaux possèdent des neurones biologiques qui constituent l'unité fonctionnelle fondamentale de leur système nerveux. Un neurone biologique est la cellule de base du système nerveux, composée d'un corps cellulaire, de dendrites, d'un axone et de boutons synaptiques. La Figure 2.2 montre une structure générale d'un neurone. Les dendrites reçoivent les signaux, l'axone les transmet, et les boutons synaptiques libèrent des neurotransmetteurs pour influencer d'autres neurones. Le fonctionnement repose sur la transmission électrochimique des signaux. Les neurones forment des réseaux complexes dans le système nerveux, permettant le traitement de l'information et le contrôle des fonctions physiologiques du corps [Patterson and Gibson 2017].

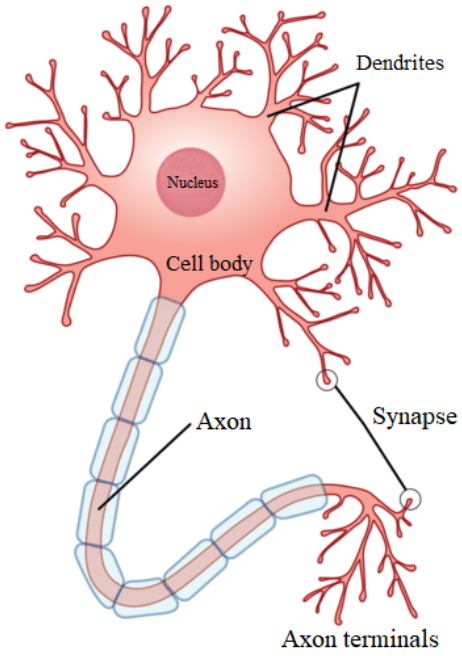


Figure 2.2: *Structure générale d'un neurone.*

3.2 Perceptron à Couche Unique

La Figure 2.3 montre un neurone de base avec plusieurs entrées. Dans les **ANNs**, chaque neurone est généralement composé de trois composants principaux : un poids (θ), un biais (b), et une fonction d'activation (f).

Un *Perceptron à Couche Unique (**SLP**)* est un type de réseau de neurones qui se compose d'une seule couche de neurones artificiels, également appelés perceptrons. Chaque perceptron reçoit de multiples entrées et produit une seule sortie, qui peut être utilisée comme entrée pour une autre couche ou comme sortie finale. Dans un **SLP**, les entrées sont multipliées par les poids correspondants, puis additionnées avec le biais, et la valeur résultante passe à travers une fonction d'activation. La sortie de la fonction d'activation est la sortie du perceptron et est exprimée par :

$$a = f(\sum_{i=1}^{k} \theta_i \mathrm{x}_i + b) \tag{2.1}$$

Les poids sont initialement assignés des valeurs aléatoires et sont ajustés lors du processus d'apprentissage pour minimiser une fonction de perte, généralement en utilisant des algorithmes d'entraînement spécifiques qui seront discutés plus tard dans ce chapitre. Le processus d'apprentissage implique d'alimenter le **SLP** avec un ensemble d'exemples d'entraînement, et d'ajuster les poids après chaque exemple pour améliorer la précision des prédictions du **SLP**.

3.3 Fondamentaux des réseaux de neurones

Les paramètres essentiels des réseaux de neurones sont les poids et les biais. Les poids sont des coefficients appliqués aux entrées, les biais sont des constantes ajoutées avant

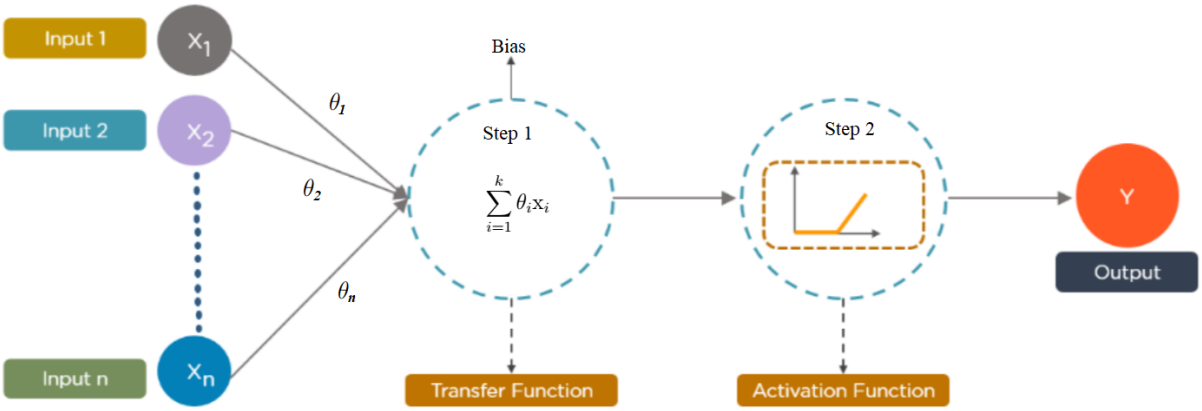


Figure 2.3: *Architecture d'un perceptron à couche unique.*

l'activation. Ils sont cruciaux pour l'apprentissage.

Un **ANN** se compose de trois types de couches : la couche d'entrée, les couches cachées et la couche de sortie.

- Entrée : Reçoit les données initiales.
- Cachées : Traitent les données pour extraire des caractéristiques.
- Sortie : Produit la prédiction finale ou le résultat de classification.

3.4 Fonctions d'Activation

Les fonctions d'activation sont cruciales en apprentissage profond car elles introduisent la non-linéarité, permettant des représentations plus complexes des données d'entrée. Sans elles, les transformations seraient purement linéaires, limitant la capacité du réseau. Le choix de la fonction influence la performance : linéaire pour l'entrée, sigmoïde pour la classification binaire, softmax pour les multiples classes, et ReLU pour les couches cachées, résolvant le problème du gradient qui disparaît (ces fonctions sont présentées en détails dans le rapport de thèse). Choisir la bonne fonction est essentiel pour des réseaux de neurones précis et efficaces.

4 Entraînement des Réseaux de Neurones

L'optimisation est cruciale pour l'entraînement des réseaux de neurones. Le but est de minimiser la fonction de coût mesurant la performance du modèle. Nous abordons les algorithmes essentiels, notamment la Descente de Gradient et la Rétropropagation. Ils ajustent les paramètres pour améliorer les prédictions sur les données d'entraînement.

4.1 Descente de Gradient

Cet algorithme permet de mettre à jour les paramètres dans la direction du gradient négatif de la fonction de coût. Il continue jusqu'à la minimisation de la fonction ou l'atteinte d'un critère d'arrêt.

Les paramètres $\theta \in \mathbf{R}^d$ sont mis à jour selon :

$$\theta_{k+1} = \theta_k - \eta \frac{\partial J(\theta_k)}{\partial \theta_k} \qquad (2.2)$$

où η est le taux d'apprentissage. Son choix est crucial, trop petit ralentit la convergence, trop grand peut conduire à des problèmes. La figure 2.5 illustre les effets des taux d'apprentissage différents. Leur réglage demande attention, souvent basé sur l'expérience ou des références externes.

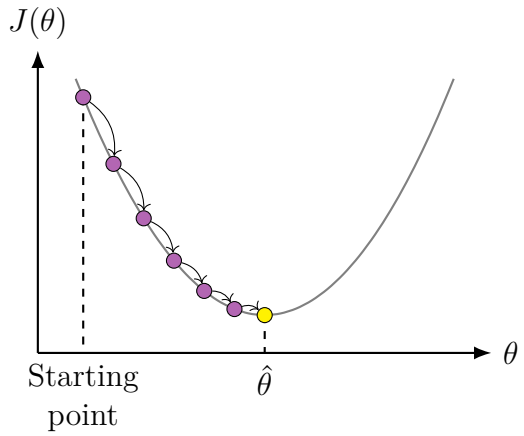


Figure 2.4: *Principe de fonctionnement de la descente de gradient pour une variable unidimensionnelle.*

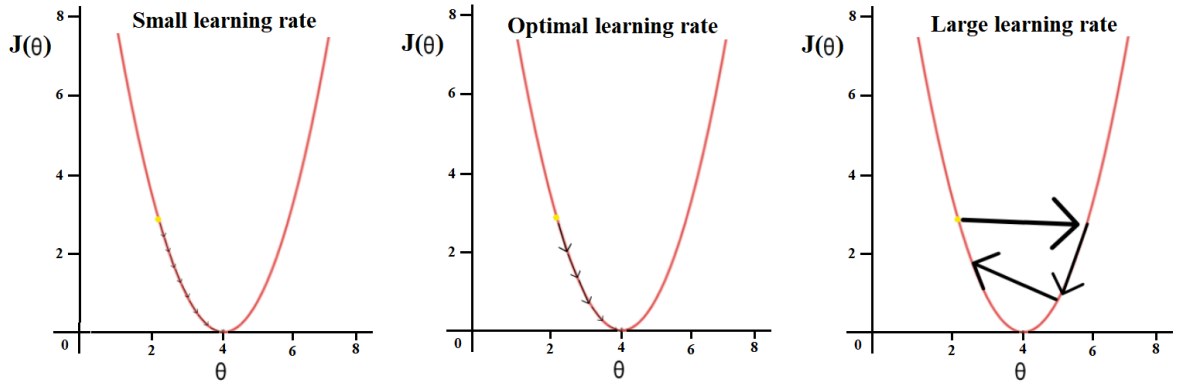


Figure 2.5: *Descente de gradient avec des taux d'apprentissage faibles, optimaux et élevés.*

4.2 Algorithme de Rétropropagation

L'algorithme de rétropropagation est un algorithme d'entraînement couramment utilisé pour les réseaux de neurones multi-couches. Il propage l'erreur en sens inverse du réseau, de la couche de sortie à la couche d'entrée en passant par les couches intermédiaires.

L'erreur est calculée comme la différence entre la sortie prédite et la sortie réelle. Cette erreur est ensuite propagée en sens inverse à travers le réseau et utilisée pour ajuster les poids des connexions entre les neurones de chaque couche.

L'algorithme de rétropropagation utilise une technique appelée la règle de la chaîne pour calculer le gradient de l'erreur par rapport aux poids du réseau. Ce gradient est ensuite utilisé pour ajuster les poids de manière à minimiser l'erreur. L'un des avantages de l'algorithme de rétropropagation est qu'il peut être utilisé pour entraîner des réseaux de neurones avec plusieurs couches cachées. Cependant, cet algorithme peut être coûteux

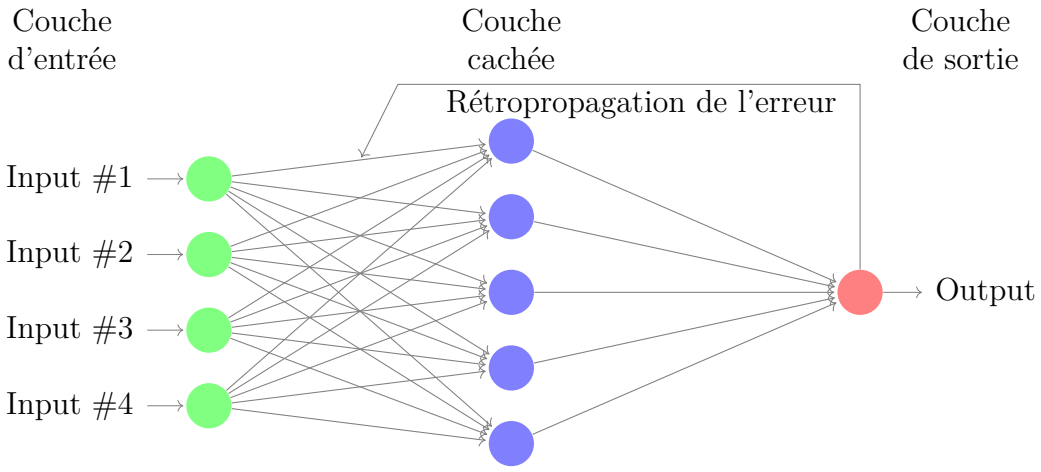


Figure 2.6: *Algorithme de rétropropagation.*

en termes de calcul et peut nécessiter une grande quantité de données d'entraînement pour obtenir de bonnes performances.

5 Conclusion

DL s'est imposé comme un outil puissant pour apprendre des motifs complexes et des relations au sein des données, sans nécessiter une programmation explicite de chaque caractéristique. Ceci est rendu possible par l'utilisation de **ANNs**, capables de reproduire le fonctionnement du cerveau humain. Dans ce chapitre, nous avons présenté la définition et les concepts de réseaux de neurones. Nous avons également exploré les méthodes d'entraînement des réseaux neuronaux, telles que les algorithmes de descente de gradient et de rétropropagation.

Part II

Contributions

Chapter

3 Adaptation de BOA pour la Conception de Circuits Analogiques

Chapter content

1 Introduction 21
2 MOWBOA pour la dimensionnement des CIs 21
3 Dimensionnement optimal du CCII 22
4 Conclusion 27

1 Introduction

Dans ce chapitre, nous présentons pour la première fois un algorithme d'optimisation multi-objectif, utilisant la technique de Tchebysheff, pondérée inspirée du papillon pour l'optimisation de la conception des circuits analogiques [Lberni et al. 2021]. À titre d'exemple de conception, un **CCII** est optimisé en tant que circuit en mode courant. Nous étudions l'effet de la longueur du canal et du courant de polarisation sur sa principale performance. L'algorithme d'optimisation proposé permet d'obtenir une haute fréquence de coupure et une petite surface d'occupation. Le présent chapitre est organisé comme suit : nous présentons d'abord l'optimisation mono- et multi-objectif. Nous décrivons l'approche d'optimisation multi-objectif proposée. Puis, nous présentons des résultats et des discussions. Nous montrons également l'éfficacité de la méthode proposée en la comparant aux résultats de la littérature.

2 MOWBOA pour la dimensionnement des CIs

La méthode proposée d'optimisation basée sur des modèles petits signaux pour la conception de CI est illustrée dans la Figure 3.1. Il s'agit d'un processus d'optimisation itératif qui implique la génération de valeurs initiales pour les variables de conception à l'aide d'un algorithme. Le modèle à petit signal du circuit est ensuite utilisé par un algorithme métaheuristique pour évaluer les fonctions objectifs et les contraintes à chaque itération. Si les contraintes sont satisfaites, les solutions non dominées sont comparées aux autres et conservées uniquement si elles sont meilleures. Sinon, de nouvelles variables d'entrée

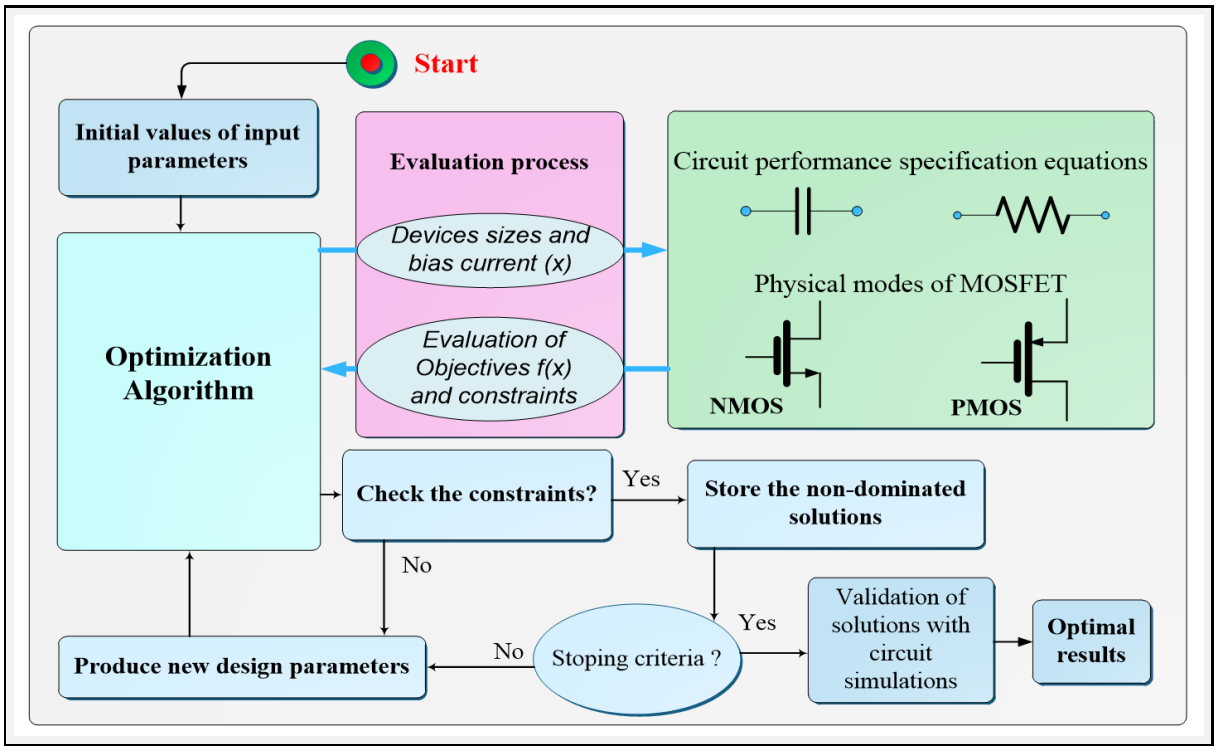


Figure 3.1: *Structure de l'optimisation basée sur les modèles proposée.*

sont générées pour l'évaluation de nouvelles performances. Le processus d'optimisation se poursuit jusqu'à ce que le critère d'arrêt est satisfait. Enfin, les solutions non dominées sont validées à l'aide de simulations de circuit dans des conditions typiques et des scénarios de pire cas (i.e. "worst-case") (variations du processus, de la tension d'alimentation et de la température), afin d'assurer une très forte probabilité de bon fonctionnement des circuits. Cette méthode est implémentée dans MATLAB 2018b. Toutes les expériences ont été réalisées sur un ordinateur avec un processeur Intel(R) Core(TM) i7 - 7820HQ @ 2,90 GHz (8 CPU) et 16 Go de mémoire. Ici, la méthode proposée est appliquée pour le dimensionnement optimal du **CCII**, traité comme un problème à objectifs multiples pour tester l'algorithme à objectifs multiples proposé.

3 Dimensionnement optimal du CCII

La figure 3.2 montre le **CCII** positif à boucle translinéaire CMOS [Soliman and Saad 2010]. X et Y sont respectivement les bornes d'entrée de courant et de tension. Z est la borne de sortie de courant. I_0 est le courant de polarisation externe. La boucle translinéaire est composée de M1-M4 qui assurent $V_X = V_Y$. Le courant de polarisation est imposé par le miroir de courant formé par les transistors M9-M13. Les transistors M5-M6 et M7-M8 forment les miroirs de courant qui assurent la production du même courant dans le nœud X et le nœud Z, $i_Z = i_X$.

Le **CCII** est simulé en utilisant la technologie *Semi-conducteur à oxyde de métal complémentaire (**CMOS**)* de 0,18 µm. Les tensions de seuil PMOS et NMOS sont respectivement d'environ -0,49 V et 0,42 V au coin typique. Les tensions d'alimentation ont été fixées à ±1.5V. Plusieurs paramètres de conception entrent en jeu dans ce circuit, à savoir les largeurs de canal ($W_{1,...,13}$) et les longueurs de canal ($L_{1,...,13}$) ainsi que le courant

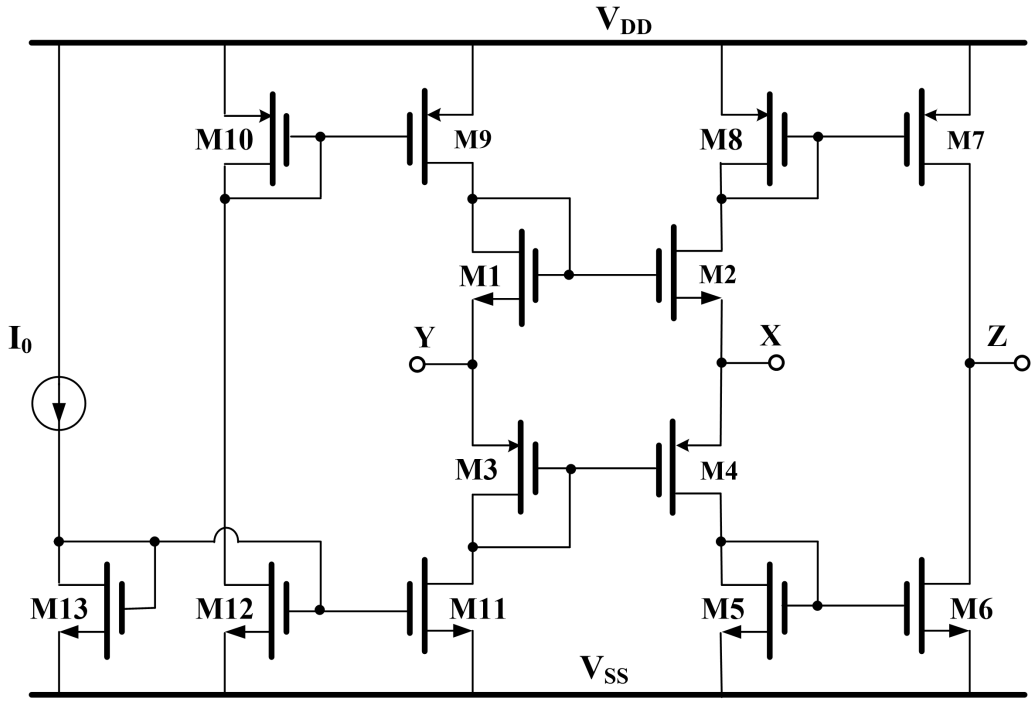


Figure 3.2: *Schéma du CCII.*

de polarisation I_0.

Afin que le **CCII** fonctionne correctement et pour faciliter le processus de dimensionnement et assurer le matching des transistors, nous supposons que tous les transistors avec les mêmes types de canaux ont les mêmes paramètres de canal (W_{P}, L_{P} pour les transistors PMOS et W_{N}, L_{N} pour les transistors NMOS), ce qui signifie la même transconductance.

Dans ce circuit, cinq variables de conception sont disponibles, à savoir les longueurs de canal (L_n, L_p) et les largeurs de grille (W_n, W_p) des transistors ainsi que le courant de polarisation I_0. Leurs limites sont données dans l'équation (3.1). Le but ici est d'obtenir un ensemble de solutions avec le meilleur compromis entre la fréquence de coupure et la résistance d'entrée intrinsèque.

$$\begin{cases} \mathrm{W_{n,p}} \in [1, 200]\mu\mathrm{m} \\ \mathrm{L_{n,p}} \in \{0.18, 0.36, 0.54\}\mu\mathrm{m} \\ I_0 \in \{40, 60, 80, 100\}\mu\mathrm{A} \end{cases} \tag{3.1}$$

Au cours du processus d'optimisation, on minimise la résistance intrinsèque ($\mathrm{R_X}$), en maximisant la fréquence de coupure $f_{\text{-3dB}}$, tout en respectant la contrainte du fonctionnement des transistors dans la région de saturation.

Afin d'évaluer les effets de la longueur du canal sur la performance du circuit, trois expériences ont été considérées. La première se produit pour $\mathrm{L} = 1\mathrm{L_{min}}$, la deuxième se produit pour $\mathrm{L} = 2\mathrm{L_{min}}$ et la troisième se produit pour $\mathrm{L} = 3\mathrm{L_{min}}$. Le $\mathrm{L_{min}}$ est le paramètre technologique du procédé CMOS choisi, qui correspond à la longueur de canal minimale, 0,18 µm. Dans chaque expérience, l'optimisation est effectuée pour quatre valeurs différentes de courant de polarisation afin d'analyser son effet sur la performance du circuit.

Pour comparaison, nous avons également effectué l'optimisation avec les algorithmes

NSGA II et **MOPSO** [Lima Moreto et al. 2019; Sibalija 2019]. Le nombre maximal d'itérations pour tous les algorithmes est de 1000, avec une population de taille 30. Dans toutes les expériences, la fréquence de coupure et la résistance intrinsèque sont explicitement optimisées, tandis que la surface est implicitement optimisée. Les figures 3.3, 3.4 et 3.5 montrent les front de Pareto obtenus pour les trois expériences menées avec les longueurs de canal L_{min}, $2L_{min}$ et $3L_{min}$, respectivement. Dans chaque figure, on retrouve les compromis optimaux obtenus pour quatre dissipations de puissance différentes $P_1 = 120\mu W$, $P_2 = 180\mu W$, $P_3 = 240\mu W$ et $P_4 = 300\mu W$. Cela correspond aux quatre courants de polarisation listés dans l'équation (3.1).

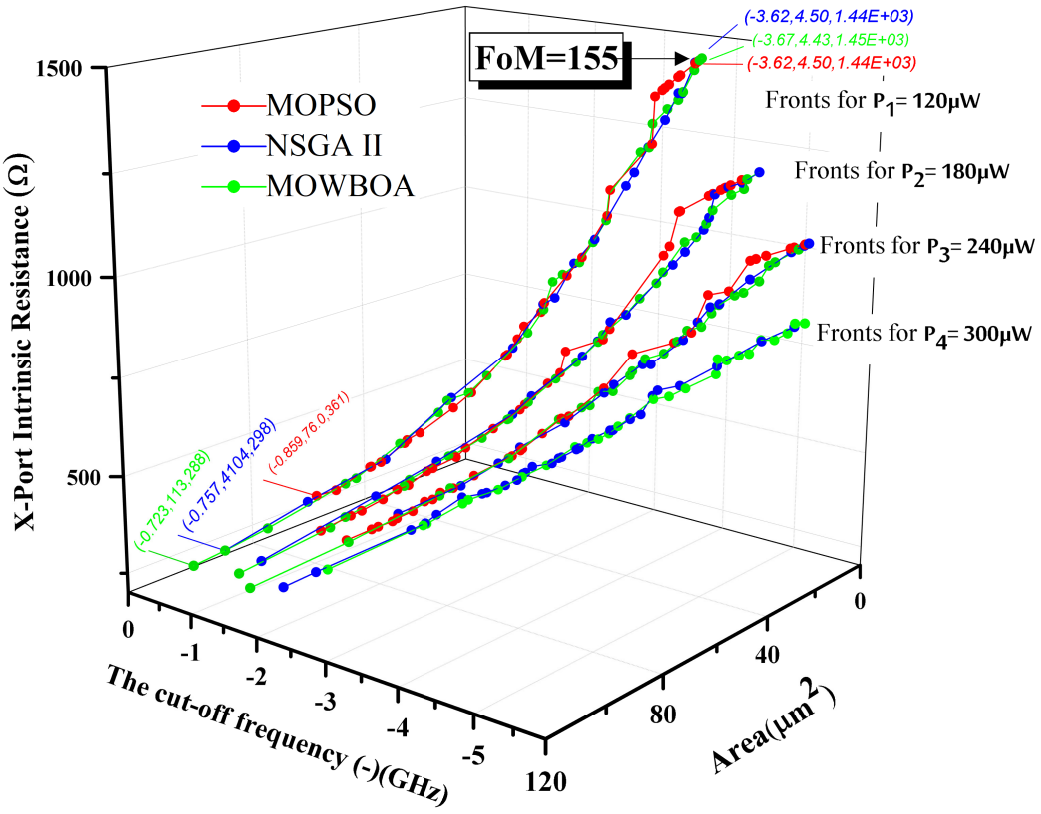


Figure 3.3: *Fronts de Pareto associés à différentes valeurs de puissance dissipée obtenues par les différents algorithmes dans le cas de* $L = 1L_{min}$.

Au cours de cette étude, nous appliquons la métrique de l'*hyper-volume* pour évaluer la distribution et la diversité des solutions de fronts de Pareto obtenus. En gardant à l'esprit que la meilleure approximation du front de Pareto correspond à la valeur la plus élevée de l'hyper-volume.

Ainsi, les statistiques de l'*hyper-volume* sont présentées dans le tableau 3.1. Le temps de convergence pour atteindre le front de Pareto optimal est respectivement de 7,45s, 17,20s et 18,39s pour **NSGA** II, **MOWBOA** et **MOPSO**. En termes de temps de convergence, l'algorithme **MOWBOA** est plus rapide que l'algorithme **MOPSO** et plus lent que l'algorithme **NSGA** II. De plus, l'algorithme **MOWBOA** fournit des valeurs de l'*hyper-volume* plus élevés que les autres algorithmes dans tous les cas, ce qui montre qu'il donne les meilleurs compromis en approchant les fronts de Pareto idéaux.

Afin de comparer le design optimisé aux études récemment publiées, la *Figure de*

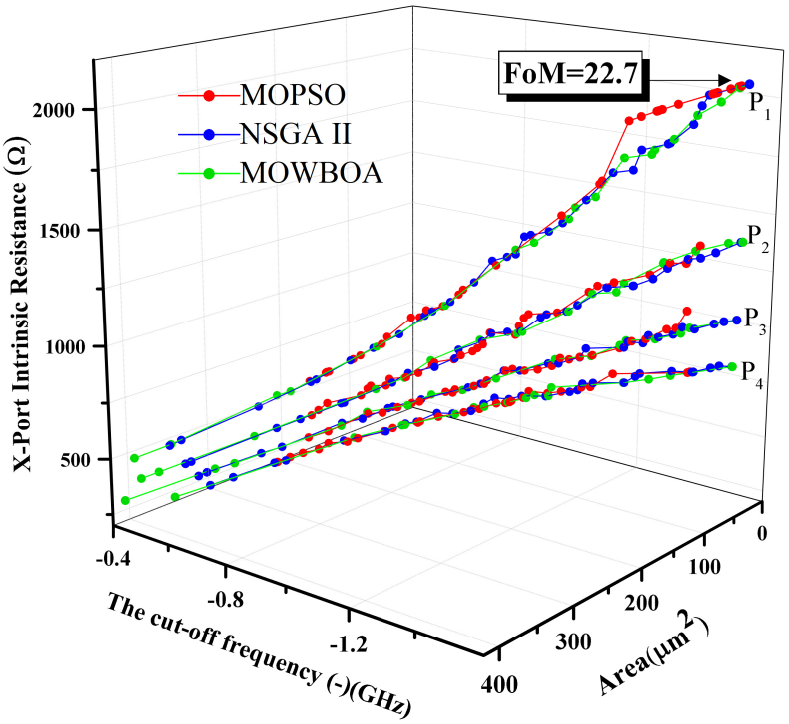


Figure 3.4: *Fronts de Pareto associés à différentes valeurs de puissance dissipée obtenues par les différents algorithmes dans le cas de* $L = 2L_{min}$.

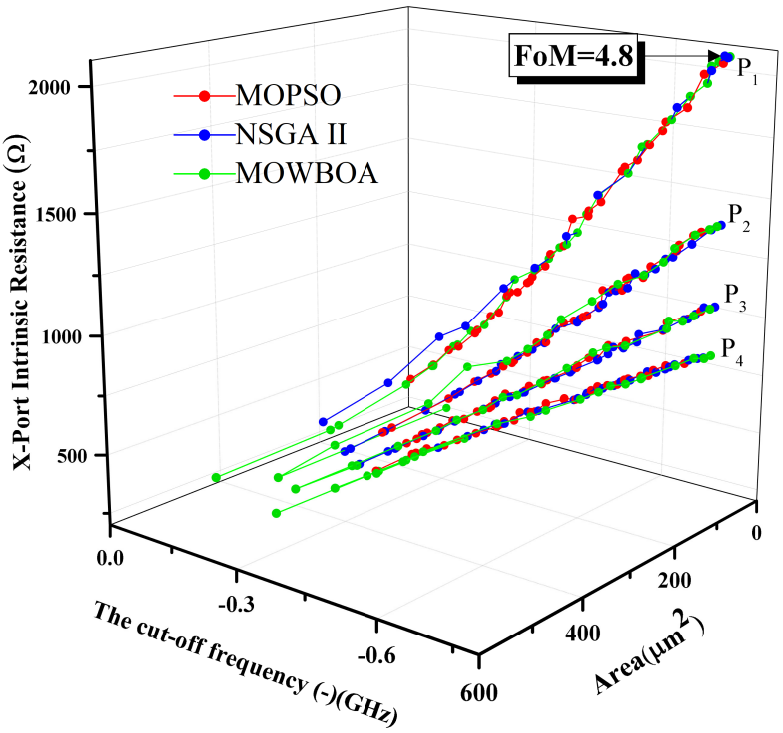


Figure 3.5: *Fronts de Pareto associés à différentes valeurs de puissance dissipée obtenues par les différents algorithmes dans le cas de* $L = 3L_{min}$.

Table 3.1: *Statistiques de l'hyper-volume.*

	$1L_{min}$			$2L_{min}$			$3L_{min}$		
P_{diss}	MOSPO	NSGA II	MOWBOA	MOSPO	NSGA II	MOWBOA	MOSPO	NSGA II	MOWBOA
120µW	0.90	0.899	**0.92**	0.884	0.899	**0.904**	0.859	0.865	**0.867**
180µW	0.872	**0.912**	**0.912**	0.857	0.888	**0.897**	0.829	0.848	**0.848**
240µW	0.856	0.895	**0.899**	0.826	0.868	**0.875**	0.803	0.823	**0.825**
300µW	0.819	0.853	**0.875**	0.771	0.842	**0.849**	0.709	0.786	**0.797**

*Mérite (**FoM**)* donné par l'équation (3.2) est utilisé.

$$\text{FoM} = \frac{f_{\text{-3dB}}[\text{GHz}].\text{L}_{\text{min}}^2[\text{nm}^2]}{\text{P}_{\text{diss}}[\text{mW}].A_{\text{CCII}}[\mu\text{m}^2].\text{R}_{\text{X}}[\Omega]} \tag{3.2}$$

Dans cette **FoM**, P_{diss} est la dissipation de puissance, A_{CCII} représente la surface totale des grilles du **CCII**. L_{min} est mis au carré afin de ne pas favoriser les implémentations utilisant des technologies de **CI** avec L_{min} plus petit et des surfaces correspondantes plus petites. En appliquant la formule de l'équation (3.2), la meilleure **FoM** trouvé est de 155, 22,7 et 4,8 respectivement pour L_{min}, $2L_{min}$ et $3L_{min}$ (voir figures 3.3, 3.4 and 3.5).

Les paramètres de conception correspondants sont indiqués dans le tableau 3.2. Il faut savoir que la **FoM** change en fonction des caractéristiques du circuit. C'est donc au concepteur de choisir la meilleure situation pour son cas d'étude. Par exemple, le tableau 3.3 présente les résultats de la meilleure solution du front de Pareto qui correspond aux paramètres de conception indiqués dans le tableau 3.2, dans le cas de $1L_{min}$.

Table 3.2: *Paramètres de conception optimisés du CCII.*

Paramètre	Valeur		
Expérience	Pour $1L_{min}$	Pour $2L_{min}$	Pour $3L_{min}$
W_n [µm]	1	1	1.54
W_p [µm]	2.915	2.92	4.71
I_0 [µA]	40	40	40

Table 3.3: *Résultats d'optimisation et de simulation du CCII correspondant a meilleur FoM trouvé.*

	Optimisation (**MOWBOA**)	Simulation	Écart en pourcentage [%]
$R_X[\Omega]$	1449.46	1444.44	0.3
$f_{\text{-3dB}}$[GHz]	3.672	3.82	3.87

Le tableau 3.4 montre une comparaison entre les résultats obtenus par l'algorithme proposé et par d'autres méthodes citées dans la littérature. On remarque que le **CCII** optimisé, offre la meilleure **FoM** correspondant à une fréquence de coupure la plus élevée, une surface la plus faible, une dissipation de puissance moyenne et une résistance R_X peu élevée. Cependant, il faut noter que pour les applications nécessitant une résistance plus faible, une autre solution des fronts de Pareto donnés ci-dessus peut être choisie.

Table 3.4: *Comparaison des performances du CCII.*

Références	Tech.[µm]	Supply [V]	P_{diss}[mW]	A_{CCII}[µm^2]	R_X[Ω]	f_{3dB}[GHz]	**FoM**
[X. Zhang and C. Wang 2019]	0.18	±0.8	3.785	470.39	223.5	1.327	0.108
[El Dor et al. 2016]	0.35	±2.5	0.25	141.42	545	0.771	4.9
[El Dor et al. 2014]	0.35	± 2.5	0.25	108.4	607	0.925	6.91
[Khateb et al. 2019]	0.18	0.5	0.509E^{-3}	13400	3000	0.578E−3	0.089
Ce travail	**0.18**	**±1.5**	**0.12**	**4.41**	**1444.44**	**3.67**	**155**

4 Conclusion

Dans ce chapitre, pour la première fois, un algorithme multi-objectif inspiré des papillons est proposé pour l'optimisation des circuits analogiques. Cet algorithme est utilisé dans le cadre d'optimisation basé sur les modèles pour le dimensionnement optimal du **CCII**. Les effets de la longueur du canal et du courant de polarisation sur la performance du convoyeur de courant ont été abordés. En utilisant la métrique de performance, la méthode proposée est plus performante en se rapprochant des fronts de Pareto idéaux, comparée à d'autres méthodes d'optimisation largement utilisées.

Une nouvelle **FoM** est proposé et évalué pour le **CCII**. Par rapport aux résultats précédemment publiés, la méthode proposée offre la meilleure **FoM**, correspondant à une fréquence de coupure élevée, une surface minimale et une dissipation de puissance moyenne,

Chapter

4 Méthode multi-objectifs basée sur la simulation pour la conception des circuits analogiques.

Chapter content

1 Introduction 29
2 Approche d'optimisation multi-objectif basée sur la simulation 29
3 Application: dimensionnement de CCII 31
4 Conclusion 35

1 Introduction

Dans ce chapitre, nous proposons une méthode basée sur la simulation pour générer l'ensemble non dominé des fronts de Pareto. Cette méthode permet de surmonter les difficultés liées à la modélisation des circuits analogiques dans différents régimes. La performance de cette méthode est testée et évaluée sur plusieurs circuits analogiques à différents régimes de fonctionnement des transistors tels que la région faible inversion, la région linéaire et la région de saturation. Mais nous ne présenterons ici que les résultats pour un seul circuit, à savoir le **CCII**. La méthode proposée est utilisée pour dimensionner le **CCII** en utilisant la technologie **CMOS** 0.18 µm. Les résultats montrent que cette méthode peut être utile pour la conception optimale des circuits intégrés, afin de répondre aux défis de la conception automatisée dans le domaine de la microélectronique.

2 Approche d'optimisation multi-objectif basée sur la simulation

Dans cette section, nous proposons une approche d'optimisation de circuits analogiques basée sur la simulation et les algorithmes multi-objectifs. Sa structure générale est représentée dans la Figure 4.1.

Les éléments clés de cette approche sont l'algorithme d'optimisation, utilisé pour trouver des solutions non-dominées, et le simulateur SPICE, qui utilise le modèle *Enz, Krummenacher, Vittoz (**EKV**)*. Le modèle **EKV** est un modèle populaire largement utilisé pour caractériser le comportement des transistor *Semiconducteur Métal-Oxyde (**MOS**)*

dans la technologie **CMOS**. Il convient particulièrement à la conception de circuits analogiques, offrant un bon compromis entre précision et complexité. Cette méthode utilise le simulateur SPICE pour évaluer les performances du circuit, vérifier les contraintes et confirmer le régime de fonctionnement des transistors.

Les données d'entrée du concepteur sont les spécifications du circuit, la technologie, les paramètres d'entrée et leur limites. Le concepteur crée la topologie du circuit et les bancs d'essai (testbench) sous forme de *netlist* au niveau du circuit (.sp). La *netlist* décrit la connectivité d'un circuit et inclut les modèles technologiques des différents composants. Elle doit également inclure les variables d'entrée, y compris les paramètres des transistors, condensateurs, résistances, etc., en tant que paramètres (*.param*). Elle spécifie également le type d'analyse (*.op*, *.ac*, *.dc*, etc.) et les méthodes de mesure des caractéristiques du circuit. La *netlist* peut également inclure des variations du processus, de la tension d'alimentation et de la température. Un exemple simple de *netlist* pour un amplificateur à deux étages avec son schéma est donné dans la Figure 4.2. L'un des principaux avantages des netlists est qu'elles évitent l'utilisation des équations du circuit, souvent difficiles à obtenir. Elles permettent également des représentations hiérarchiques des circuits analogiques et peuvent aider dans diverses tâches d'automatisation de la conception.

Les principales étapes de l'approche proposée :
Une fois que les spécifications sont définies, y compris les objectifs et les contraintes de la conception du circuit, et que la netlist est créée pour l'analyse spécifique, l'approche proposée commence par générer un ensemble de solutions liées à la *netlist* correspondante qui modélise précisément le circuit. Ensuite, elle lance des simulations pour évaluer les performances du circuit sous différents paramètres d'entrée. En d'autres termes, elle manipule la *netlist* et communique avec le simulateur HSPICE pour exécuter les simulations

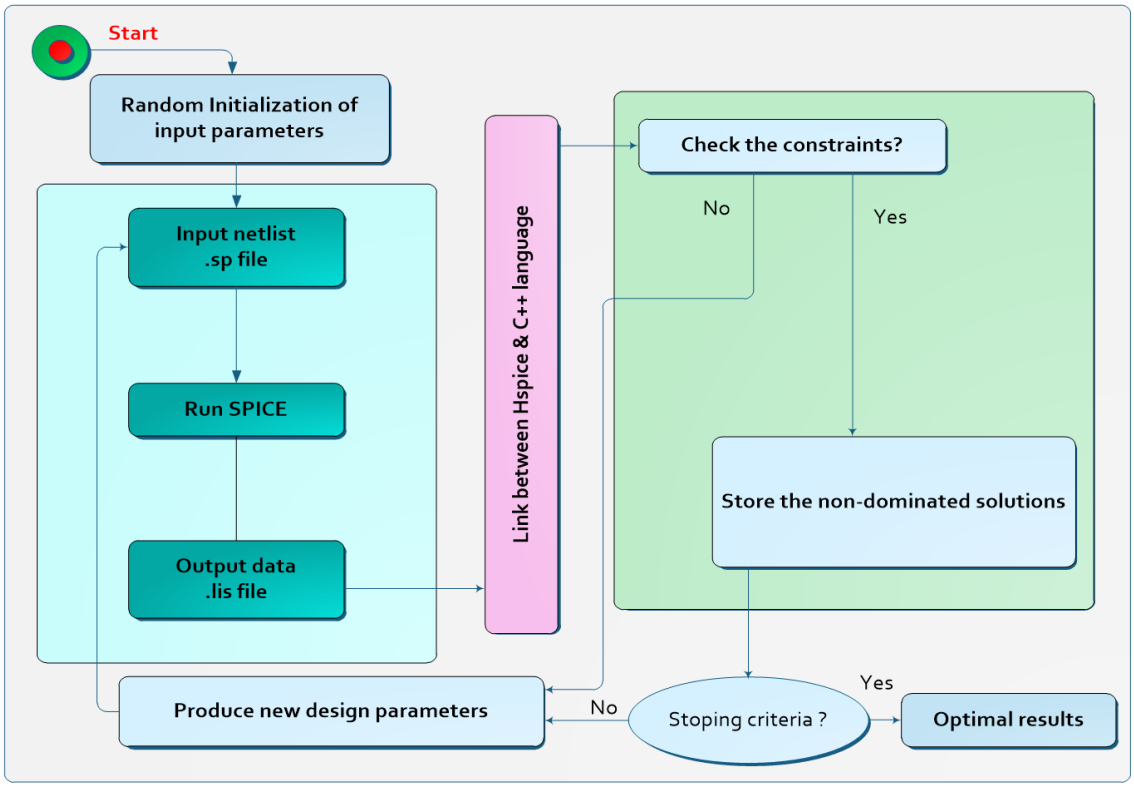


Figure 4.1: *Outil proposé pour l'optimisation des circuits analogiques.*

```
********** Test bench **********
********** Variables Descriptions ***
 .param w1=4u w2=2u w3=12u  ...
********** MODEL Descriptions ***
 .MODEL modn NMOS (.....)
 .MODEL modp PMOS (.....)
      ......
********** NETLIST Description ***
  vdd   vdd     0    vdd
  vss   vss     0    vss
  I0   net1    vdd    I0
  M8   net1   net1   vdd   vdd  modp   w4  l4  N4
  M5   net2   net1   vdd   vdd  modp   w3  l3  N3
  M7   vout   net1   vdd   vdd  modp   w5  l5  N5
  M1   net3   vinn  net2   vdd  modp   w1  l1  N1
  M2   net4   vinp  net2   vdd  modp   w1  l1  N1
  M3   net3   net3   vss   vss  modn   w2  l2  N2
  M4   net4   net3   vss   vss  modn   w2  l2  N2
  M6   net4   vout   vss   vss  modn   w6  l6  N6
  Cc   net5   vout    Cc
  Rc   net4   net5    Rc
********** Testbench settings
  CL     vout       0  CL
  V_ip1  vinp  0  dc  0.8  ac 1m
  V_in1  vinn  0  dc  0.8  ac  0
*** SIMULATION Commands ***
 .ac  dec  10  1  10g
 .probe vdb(*)
*Measure statement for the dc gain
 .measure ac dcgain find vdb(out) at=10
*The unity-gain frequency
 .meas ac ugf when vdb(out)=0 cross=1
*The phase margin
 .meas ac pm find vp(out) when vdb(out)=0
 .end
```

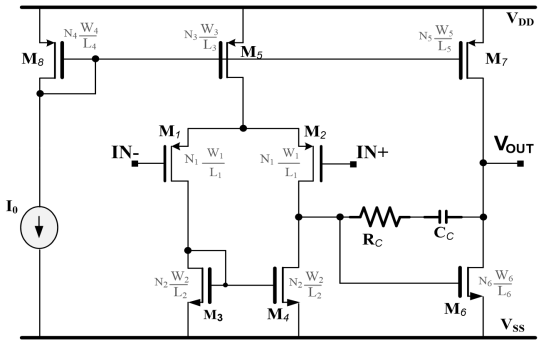


Figure 4.2: *SPICE Netlist pour l'analyse AC (.ac) d'un amplificateur à deux étages.*

pour les paramètres d'entrée générés par l'algorithme d'optimisation. Puis, elle extrait les données de sortie, y compris différentes caractéristiques du circuit telles que le gain, la bande passante, la consommation d'énergie, les régimes de fonctionnement des transistors, etc., à partir du fichier de sortie (.lis), et vérifie également les contraintes. Si les contraintes sont vérifiées, les résultats sont stockés dans une archive externe. Ensuite, l'algorithme met à jour les paramètres d'entrée dans la netlist et exécute des simulations pour évaluer les nouvelles performances du circuit. Si les contraintes sont vérifiées, l'archive externe est mise à jour. Ensuite, le processus d'optimisation se poursuit jusqu'à ce que les critères d'arrêt soient atteints. La partie d'optimisation, qui est le **MOPSO-CD**, intègre une métrique **CD** qui aide à maintenir la diversité dans l'archive externe en montrant les compromis entre les différents objectifs. Cela peut aider les concepteurs à identifier l'ensemble optimal de valeurs de paramètres pour une application du circuit.

Enfin, le concepteur peut déterminer à quel point l'ensemble optimal de valeurs de paramètres est robuste face aux variations des paramètres du circuit et valider la conception de circuit optimisée en utilisant des analyses supplémentaires, telles que des simulations **PVT** et/ou des mesures physiques, pour s'assurer qu'elle fonctionne comme prévu.

3 Application: dimensionnement de CCII

Toutes les expériences ont été menées sur un ordinateur avec un Intel Core i7-7820HQ GHz, une RAM de 16G. Toutes les simulations sont effectuées en technologie **CMOS** 0,18

µm.

Pour les applications utilisant des transistors polarisés en dessous du seuil, les circuits sont alimentés par une tension d'alimentation de ±0.5V.

Pour les applications à haute performance qui nécessitent une large bande passante, les circuits sont polarisés avec une tension d'alimentation de ±1.5V.

Nous avons effectué l'optimisation en utilisant deux algorithmes **MOPSO-CD** et **NSGA** II. Dans toutes les expériences, nous avons utilisé 200 itérations, et une taille de population de 50. Ces algorithmes sont utilisés ici avec la même configuration de paramètres, comme indiqué dans [Azadeh et al. 2017].

Table 4.1: *Paramètres de MOPSO-CD et NSGA II.*

NSGA II		Taux de croisement		Taux de mutation	Taille de l'archive
		0.7		0.5	100
MOPSO	c_1	c_2	w	Taux de mutation	Taille de l'archive
	2.5	2.5	0.8	0.5	100

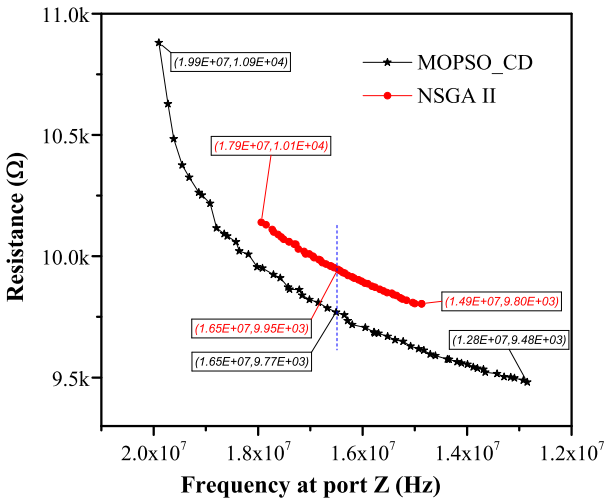
Le **CCII**, déjà présenté sur la figure **??**, est choisi comme test pour valider la méthode proposée dans ce chapitre. Plusieurs paramètres de conception sont utilisés ici à savoir les largeurs (W_1-W_{13}) et les longueurs (L_1-L_{13}) des transistors ainsi que le courant de polarisation I_0. Le bon fonctionnement de ce circuit est assuré par les relations de correspondance entre les transistors (M11≡M12≡M13≡M5≡M6, M1≡M2, M3≡M4, et M9≡M10≡M7≡M8). Nous supposons que tous les transistors ayant les mêmes types de canal ont les mêmes longueurs de canal. Ainsi, il y a sept paramètres de conception indépendants. Par conséquent, le vecteur des paramètres de conception qui doivent être optimisés pour ce circuit est donné: $X_i = (W_1, W_3, W_5, W_7, L_1, L_3, I_0)$

Les largeurs et longueurs des canaux des transistors peuvent être respectivement variées dans $[1, 100]$ µm et $[0.18, 1]$ µm. Le courant de polarisation peut être varié dans une plage de $[0.1, 100]$ µA. Nous avons considéré de larges intervalles pour ces paramètres, car le circuit est optimisé dans tous les régimes de transistors.

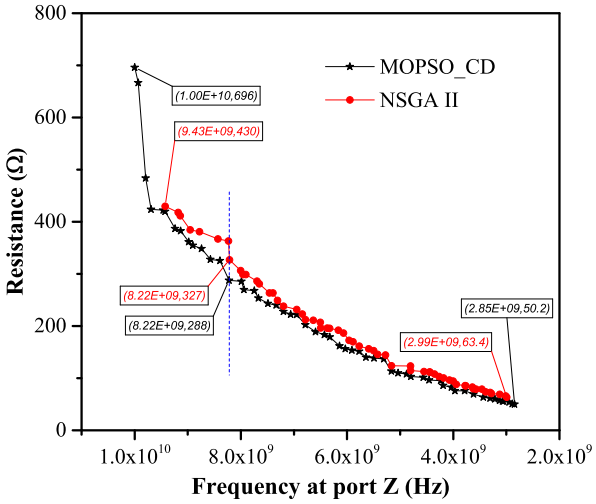
L'objectif est de maximiser la fréquence de coupure élevée au port Z, f_{ciZ}, et de minimiser la résistance parasite au port X, R_{X}, en tenant compte des contraintes du régime de fonctionnement du transistor.

Pour démontrer l'utilité de la méthode proposée et étudier l'influence des régimes de fonctionnement des transistors sur le dimensionnement et la performance du **CCII**, nous optimisons ces caractéristiques pour les trois régimes de transistors : faible inversion, linéaire et saturation. Le but principal est de fournir un front de Pareto de ces caractéristiques dans chaque cas. Ainsi, la figure 6.4 montre les fronts de Pareto approximatifs générés par les deux algorithmes pour les trois régimes.

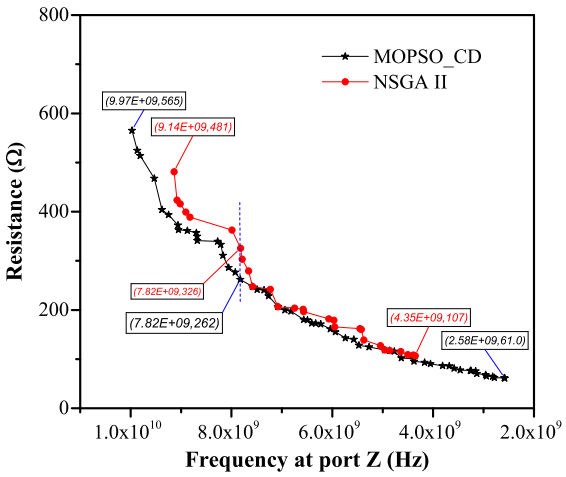
Les deux algorithmes trouvent des solutions réalisables dans chaque expérience. Cependant, il est bien connu que la polarisation en faible inversion entraîne des tailles élevées, une faible fréquence de fonctionnement et des résistances importantes par rapport aux circuits fonctionnant en forte inversion. En fait, en faible inversion, la densité de porteurs est faible par rapport à celle disponible dans les régions linéaire et saturation, ce qui rend difficile la possibilité d'atteindre rapidement un courant souhaitable. Par conséquent, un courant donné nécessite une collecte de porteurs sur une grande surface, ce qui entraîne une augmentation de la résistance impliquée et finalement un temps de réponse plus long correspondant à des basses fréquences de réponse par rapport aux autres cas.



(a) Régime de faible inversion.



(b) Régime linéaire.



(c) Régime de saturation.

Chaque front de Pareto représente ici le meilleur compromis entre la résistance parasite R_X et la fréquence f_{ciZ}. Les points noirs et rouges représentent les solutions obtenues respectivement par les algorithmes MOPSO-CD et NSGA II.

Figure 4.3: *Fronts de Pareto obtenus par les deux algorithmes dans les différentes expériences.*

Nous comparons les différentes plages caractéristiques (fréquence, résistance parasite et consommation électrique) dans la Figure 6.4, en notant que la consommation électrique n'est pas représentée car elle est optimisée implicitement. Sa valeur est obtenue à partir du courant de polarisation I_0 défini comme paramètre de conception.

En ce qui concerne les plages de performances, on peut observer que les conceptions dans le régime de faible inversion ont une bande passante plus faible et une résistance intrinsèque plus élevée par rapport aux conceptions dans les régions linéaire et de saturation. Cela est conforme au fait que les circuits fonctionnant en faible inversion ont des tailles plus grandes, une fréquence de fonctionnement plus faible et des résistances plus importantes par rapport aux circuits fonctionnant en saturation ou en forte inversion, comme déjà mentionné.

Dans le régime de faible inversion, la bande passante est d'environ 13 à 20 MHz pour **MOPSO-CD** et d'environ 15 à 18 MHz pour **NSGA** II, tandis que la résistance intrinsèque se situe entre 9,5 et 10,9 kΩ pour **MOPSO-CD** et entre 9,8 et 10,1 kΩ pour **NSGA** II. De plus, la consommation électrique se situe dans la plage de micro-watts, allant de 85 à 100 µW pour **MOPSO-CD** et de 90 à 96 µW pour **NSGA** II. De telles conceptions conviennent aux applications biomédicales nécessitant des tensions inférieures à 0,5V et une faible consommation d'énergie, telles que les filtres d'instrumentation [Yucehan and Yuce 2022], les amplificateurs [Sharan et al. 2020] et les capteurs de lecture [Ferri et al. 2009].

Pour les deux autres régimes, la bande passante se situe dans la plage du gigahertz et la résistance intrinsèque dans la plage de l'ohm. La consommation électrique se situe entre 255-300µW. Nous atteignons des plages de bande passante plus élevées et de plus petites plages de résistance parasite dans les deux expériences, ce qui fait des conceptions de **CCII** obtenues des amplificateurs de tension haute fréquence utilisés dans des applications telles que l'amplification Radiofréquence et conditionnement du signal. En d'autres termes, ils sont utilisés pour amplifier et préparer les signaux haute fréquence pour un traitement ultérieur ou une transmission dans des applications Radiofréquence. Ils peuvent également être utilisés pour mettre en œuvre des filtres actifs d'ordre élevé pour des signaux hautes fréquences dans les systèmes de communication et les applications audio.

Les valeurs des caractéristiques correspondantes aux points extrêmes de front du Pareto obtenus sont fournies dans chaque expérience de la Figure 6.4. Si on compare les solutions du front de Pareto des deux algorithmes à certains points, on peut observer que la solution de **MOPSO-CD** est meilleure que celle de **NSGA** II. Des exemples de solutions sont indiqués au milieu de chaque front de Pareto. Par exemple, dans le régime de faible inversion, à 16,5 MHz, **MOPSO-CD** atteint une faible résistance R_X de 9,95 kΩ, tandis que **NSGA** II atteint 9,77 kΩ. De plus, lors de la comparaison des différentes plages de variations des solutions réalisables obtenues par les deux algorithmes, on peut voir que **MOPSO-CD** a généré de manière cohérente des solutions avec une bande passante élevée, une faible résistance et une faible consommation d'énergie sur les trois expériences. De plus, **MOPSO-CD** présente les plages de variations les plus étendues des caractéristiques optimisées, ce qui indique que l'utilisation de la technique **CD** contribue à maintenir la diversité des solutions. Ces résultats suggèrent également que **MOPSO-CD** est bien adapté pour le dimmensionnement des circuits avec une faible consommation d'énergie et une faible résistance parasite.

La tableau 4.2 résume les valeurs des métriques de performance pour la conception optimale de **CCII** obtenues à la fois par les algorithmes **MOPSO-CD** et **NSGA** II. Le temps de convergence et les valeurs statistiques de l'hyper-volume et de la métrique C

pour chaque expérience sont fournies dans le tableau.

Table 4.2: *Comparaison des valeurs des métriques de performance du CCII optimisé.*

	Temps de convergence (h)		Hyper-volume		Métrique C	
Expériences	MOPSO-CD	NSGA II	MOPSO-CD	NSGA II	$C(F_1/F_2)$	$C(F_2/F_1)$
Régime de faible inversion	**3.12**	3.24	**0.86**	0.50	**1**	0
Régime linéaire	**3.04**	3.13	**0.94**	0.72	**27/28**	0
Régime de saturation	**3.18**	3.25	**0.91**	0.34	**22/26**	1/40

Note: F_1 et F_2 sont respectivement les fronts de Pareto de MOPSO-CD et de NSGA II.

On remarque que **MOPSO-CD** obtient de bons résultats dans les trois expériences, en affichant le temps de convergence le plus court et les valeurs d'hyper-volume les plus élevées. Les valeurs d'hyper-volume mesurent la convergence et la diversité des solutions obtenues. Plus l'hyper-volume est élevé, plus l'algorithme est capable d'approcher le front de Pareto réel. Dans les expériences menées dans les trois régions, **MOPSO-CD** obtient des valeurs d'hyper-volume de 0,86, 0,94 et 0,91 respectivement, tandis que **NSGA** II obtient des valeurs de 0,50, 0,72 et 0,34 pour les mêmes expériences.

Les valeurs de la métrique C donnent également une indication de la qualité des solutions obtenues. Les valeurs $C(F_1/F_2)$ et $C(F_2/F_1)$ représentent la distance entre les fronts de Pareto approximatifs de **MOPSO-CD** et **NSGA** II, respectivement. **MOPSO-CD** présente de meilleurs résultats en termes de métrique C dans les trois expériences par rapport à **NSGA** II. En conclusion, sur la base des résultats du tableau 4.2, il apparaît que **MOPSO-CD** est plus efficace que **NSGA** II pour la conception de circuits **CCII** optimaux.

4 Conclusion

Dans ce chapitre, un outil efficace d'optimisation multi-objectif basé sur la simulation est présenté et testé sur le **CCII**. Cet outil permet la conception optimale des circuits analogiques pour différentes applications, notamment des applications biomédicales ou portables où les transistors sont polarisés en dessous du seuil. Cela permet de concevoir des circuits alimentés par une tension inférieure à 0,5 V. D'autre part, des applications à haute performance nécessitent une bande passante élevée où les transistors sont polarisés dans les régions linéaire et saturation. Selon ces applications, plusieurs expériences ont été utilisées pour le **CCII** et tous les résultats ont été obtenus en utilisant des simulations SPICE en technologie **CMOS** 0,18 µA. Les résultats obtenus pour l'application où les transistors sont polarisés dans les régions linéaire et saturation ont une consommation d'énergie inférieure à 300 µW et une gamme de fréquence de l'ordre du gigahertz. Pour les applications à faible tension et faible puissance, les conceptions résultantes ont une consommation de puissance inférieure à 100 µW et des fréquences de l'ordre du mégahertz. D'après les mesures de performance, c'est-à-dire le temps de convergence, les statistiques de la métrique C et de l'hyper-volume, la méthode proposée avec **MOPSO-CD** permet de mieux approcher les vrais fronts de Pareto, par rapport à la **NSGA** II. Les résultats obtenus sont excellents pour aider les concepteurs à choisir la meilleure conception pour une application spécifique.

Chapter

5 Optimisation assistée par les réseaux de neurones profonds

Chapter content

1 Introduction 37
2 Méthode d'optimisation assistée par DNN 37
3 Application: Op-Amp Cascode Replié 39
4 Conclusion 43

1 Introduction

Récemment, de nombreuses études ont tenté d'exploiter les progrès de l'**IA** dans la conception des circuits *Analogique et à Signal Mixte (**AMS**)*. Les deux principales applications de l'**IA** dans la conception **AMS** sont l'optimisation automatisée du dimensionnement des circuits et l'amélioration de la précision des modèles de performance. Les algorithmes d'apprentissage profond destinés à accélérer le dimensionnement des circuits intégrés analogiques attirent beaucoup d'attention. Ils ont connu un succès remarquable. Cependant, la plupart des recherches dans ce domaine ne prennent en compte que quelques spécifications typiques de conception des circuits analogiques. Nous proposons une méthode de dimensionnement des circuits analogiques. La performance du cadre proposé est testée sur un circuit analogique.

2 Méthode d'optimisation assistée par DNN

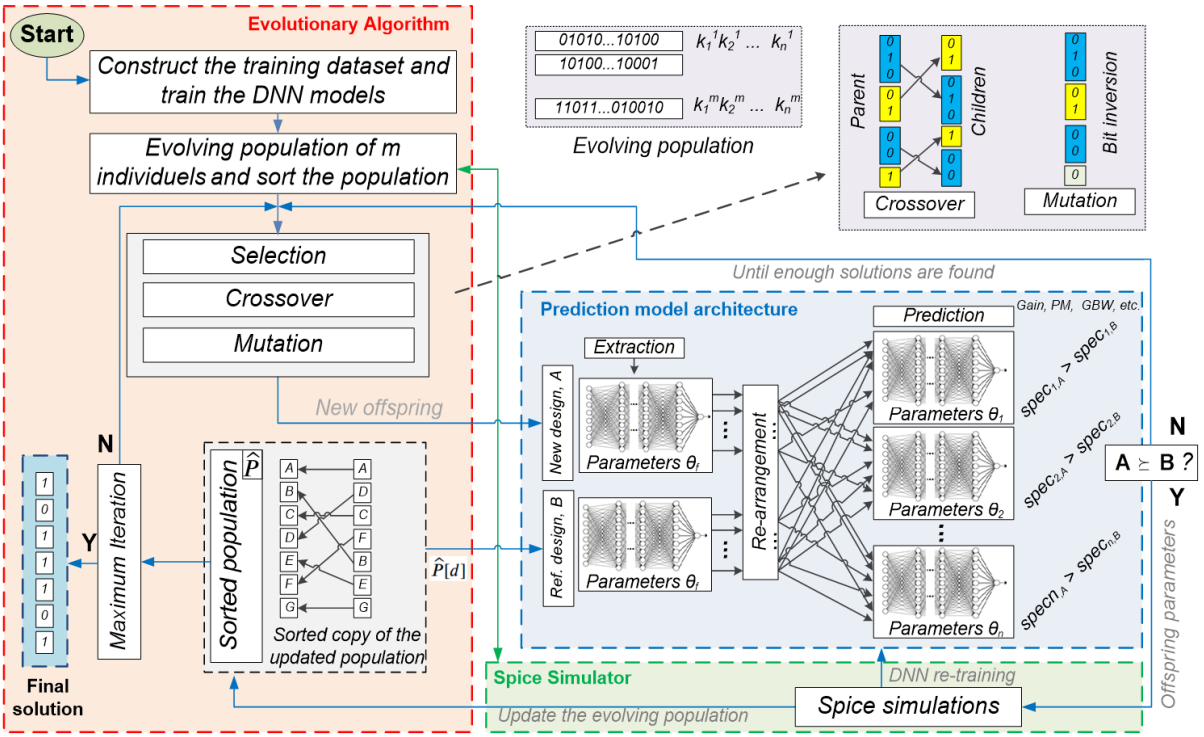
Nous introduisons une approche d'optimisation assistée par le **DNN** où nous fusionnons l'**AE**, les concepts de population et les **DNN**s. Les principales caractéristiques de l'approche proposé sont les suivantes :

1) Un algorithme **AE** combiné avec des simulations SPICE est utilisé pour améliorer l'efficacité de l'optimisation et fournir des résultats de haute précision lors de l'optimisation globale.

2) Une recherche locale est intégrée pour améliorer les performances de l'optimisation globale. Au cours de cette phase, le modèle **DNN** est utilisé pour évaluer les performances des circuits analogiques.

3) Une évaluation du modèle **DNN** est utilisée à la place des appels de simulations SPICE pour le classement des conceptions candidates, ce qui accélère la recherche du minimum local.

Pour le réentraînement du **DNN**, des simulations des meilleures conceptions candidates trouvées sont effectuées. La figure 5.1 montre le cadre proposé pour l'optimisation de la conception des circuits intégrés. Il se compose principalement de modèles de prédiction artificiels et d'un **AE** basé sur la population interagissant avec un simulateur de circuit pendant la phase d'optimisation, lorsque le modèle de prédiction trouve de meilleures solutions.



La méthode du tournoi est utilisée pour effectuer des opérations de croisement et de mutation sur l'individu parental. Le processus est répété jusqu'à la génération finale.

Figure 5.1: *Cadre proposé pour l'optimisation de la conception des CIs.*

Le cadre commence avec une population générée aléatoirement en évaluant tous les individus K_i, de la population (simulations SPICE). Ensuite, à chaque itération, nous utilisons l'**AE** pour créer une nouvelle population et sélectionner la nouvelle génération en fonction des caractéristiques critiques. Après avoir généré de nouveaux échantillons, nous ne faisons pas que les simuler et les comparer à l'ancienne population pour savoir s'ils seront pris en compte dans la génération suivante. Nous utilisons le modèle de prédiction **DNN** illustré par la zone bleue en pointillé pour prédire les résultats de tout nouvel échantillon, c'est-à-dire, voir s'il est meilleur que l'échantillon trié de la population courante. Si la réponse est oui, nous effectuons des simulations SPICE et nous les prenons en compte dans la génération suivante.

L'**AE** continue de générer de nouvelles conceptions jusqu'à ce que suffisamment d'échantillons soient trouvés. Si le critère d'arrêt n'est pas rempli, nous procédons aux opérations de croisement et de mutation. Il est possible que la prochaine génération ait une distribution de probabilité différente de celle de leurs parents, sur laquelle le **DNN** a été entraîné. Pour résoudre ce problème, pour chaque nouvelle génération approuvée, nous mettons à jour la population. Nous ré-entraînons les modèles de prédiction **DNN** avec les nouvelles données de simulation en utilisant *Descente de gradient stochastique (**SGD**)*, comme indiqué dans la zone verte en pointillé. Le processus ci-dessus est répété jusqu'à ce que les critères d'arrêt soient remplis. L'ensemble du cadre est présenté, étape par étape, dans le pseudo-code de l'algorithme 1.

Algorithm 1 Pseudo-code de l'algorithme d'optimisation assistée par DNN

1: La taille de la population m, le nombre maximum d'itération max_{iter}, les probabilités de mutation et de croisement, l'indice de référence d, les hyper-paramètres de **DNN**.
2: Étant donné une population initiale P=$\{(k_1^1, k_2^1, ..., k_n^1), (k_1^2, k_2^2, ..., k_n^2), ..., (k_1^m, k_2^m, ..., k_n^m)\}$ simulée aléatoirement, n présente la taille du paramètre dans le **DNN** ;
3: Pré-entraînement de **DNN**;
4: **for** $t = 1, 2, ..., max_{iter}$ **do**
5: La population évolutive est triée $\widehat{P} \leftarrow P$ par $cost(x)$
6: Sélection d'un modèle de référence $\leftarrow \widehat{P}[d]$
7: **while** P_i.length $< d$ **do**
8: $p_1, p_2 \leftarrow$ Sélectionnez deux parents à partir de P en utilisant la sélection par tournoi binaire;
9: $k_1, k_2 \leftarrow$ Générer de nouveaux descendants en appliquant la mutation et le croisement;
10: $P_i \leftarrow P_i \cup k_1 \cup k_2$
11: **end while**
12: **while** P_i.length $< d$ **do**
13: $\mathcal{P} \leftarrow$ Probabilité de sortie pour les entrées P_i et $\widehat{P}[d]$;
14: **if** $\mathcal{P}[i] = 1$ **then**
15: Effectuer des simulations SPICE;
16: **else**
17: Générer une nouvelle génération (lignes 9-11);
18: **end if**
19: **end while**
20: Mise à jour de la population $\widehat{P}$;
21: Ré-entraîner le modèle de prédiction **DNN** en utilisant **SGD**;
22: **end for**

3 Application: Op-Amp Cascode Replié

Plusieurs expériences ont été effectuées, qui clarifient certains aspects de la méthode proposée et démontrent ses capacités sur une variété de circuits analogiques. Mais, nous ne présentons ici que les résultats du rail-to-rail **Op-Amp** cascode replié. Ce circuit est optimisé en utilisant la technologie CMOS 180 nm. Nous utilisons un ensemble complet de spécifications strictes utilisées dans la pratique de la conception dans le monde réel. Ces spécifications proviennent des AC, DC et TR. Ces analyses ont toutes des relations entre elles. Selon les applications, certaines de ces caractéristiques peuvent être considérées comme plus importantes que d'autres. Pour des raisons de généralité, toutes ces

spécifications sont considérées comme importantes. Comme déjà mentionné, l'objectif d'optimisation d'un circuit intégré analogique est de trouver la conception optimale correspondant à la meilleure fonction de coût (c'est à dire la figure de mérite) qui optimise plusieurs caractéristiques. Nous comparons notre méthode avec un algorithme évolutionnaire. En outre, les résultats de l'optimisation sont également comparés à ce qui a été rapporté précédemment. Pour tenir compte des effets aléatoires liés aux différentes méthodes utilisées, nous avons effectué dix exécutions pour chaque méthode. Le cadre proposé est codé en Python, et toutes les expériences d'optimisation sont menées sur un PC équipé d'un Intel Core i7-7820HQ 2,9 GHz, une RAM de 16G.

Pour mettre en évidence les avantages de la méthode proposée, nous optimisons l'amplificateur opérationnel cascode replié rail-to-rail. La figure 5.2 présente le schéma de ce circuit. Il comporte 35 paramètres de conception, notamment les largeurs et les longueurs des transistors, le courant de polarisation, les résistances de compensation et les condensateurs. Les plages de conception de la recherche sont indiquées dans le tableau 5.1. Pour réduire l'effet de modulation du canal, la longueur du canal de tous les transistors est fixée à $3 \times L_{\text{min}}$.

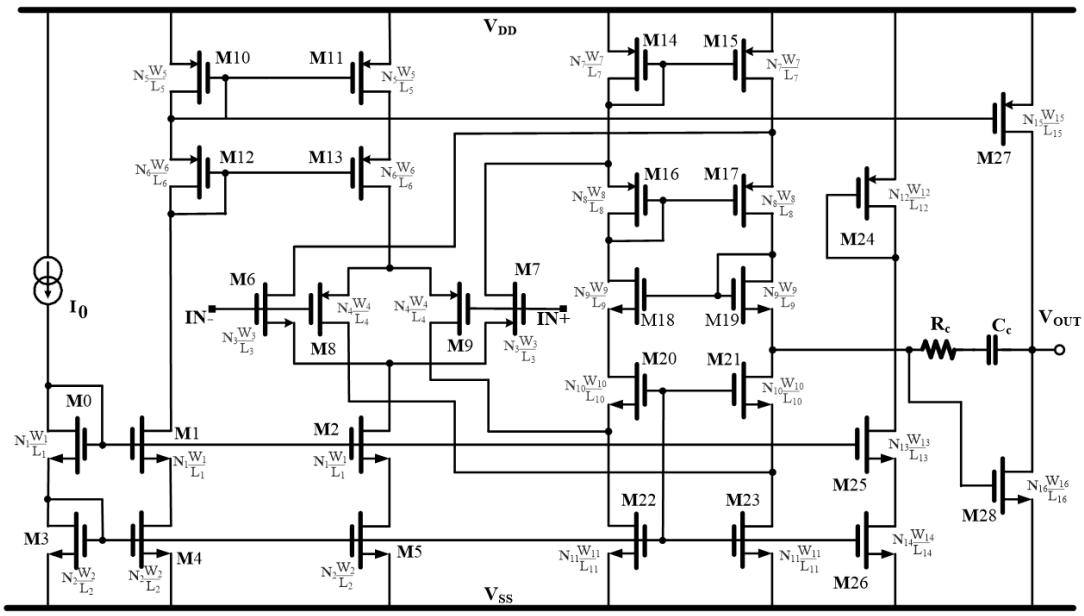


Figure 5.2: *Schéma de l'amplificateur opérationnel cascode replié rail-to-rail.*

L'objectif est de trouver les paramètres de conception optimaux de sorte que le circuit réponde aux spécifications données dans le tableau 5.2. Dans cette expérience, tous les transistors doivent fonctionner dans le régime de saturation. Des fonctions et des contraintes différentes de celles rapportées peuvent être utilisées.

Dix exécutions sont effectuées, et toutes satisfont entièrement les spécifications requises. La méthode proposée converge en 283 simulations (équivalant à 24,3 minutes, y compris la population initiale générée de 100 modèles), ce qui est inférieur au budget de simulation dans le cas des algorithmes évolutionnaires (10100 simulations, qui équivalent à 2,21 heures). Le tableau 5.3 présente les paramètres types de la fonction de coût optimale, tandis que le tableau 5.4 présente les fonctions correspondantes. Les deux méthodes ont trouvé des solutions qui répondent à toutes les spécifications. Cependant, la méthode proposée a obtenu la meilleure performance avec un temps de convergence plus court.

Table 5.1: *Paramètres de conception et leurs limites.*

Paramètre	lb_i	ub_i	Paramètre	lb_i	ub_i
$W_{0,1,2}$	1	20	$W_{3,4,5}$	1	20
$W_{6,7}$	1	20	$W_{8,9}$	1	20
$W_{10,11}$	1	20	$W_{12,13}$	1	20
$W_{14,15}$	1	20	$W_{16,17}$	1	20
$W_{18,19}$	1	20	$W_{20,21}$	1	20
$W_{22,23}$	1	20	W_{24}	1	20
W_{25}	1	20	W_{26}	1	20
W_{27}	1	20	W_{28}	1	20
$m_{0,1,2}$	1	20	$m_{3,4,5}$	1	20
$m_{6,7}$	1	20	$m_{8,9}$	1	20
$m_{10,11}$	1	20	$m_{12,13}$	1	20
$m_{14,15}$	1	20	$m_{16,17}$	1	20
$m_{18,19}$	1	20	$m_{20,21}$	1	20
$m_{22,23}$	1	20	m_{24}	1	20
m_{25}	1	20	m_{26}	1	20
m_{27}	1	20	m_{28}	1	20
C_{C}	2	8	R_C	10	100
I_{bias}	1	100			

W et L sont exprimés en µm. R_{C} et C_{C} sont respectivement exprimés en kΩ et pF. I_{bias} est exprimé en nA.

Table 5.2: *Spécifications de conception et performance de la solution trouvée.*

Performance	Exigences
DC gain [dB]	≥ 70
GBW [MHz]	≥ 10
PM [°]	≥ 60
CMRR [dB]	≥ 60
PSRR [dB]	≥ 60
Settling time [ns]	≤ 100
Offset [mV]	≤ 1
Power [mW]	≤ 1

Le tableau 5.5 présente les résultats statistiques des deux méthodes. Dans ce tableau, *N-sim* est le nombre moyen de simulations pour atteindre la conception possible (moyenne sur dix exécutions). Le taux de réussite est le nombre d'exécutions positives divisé par le nombre total d'exécutions. La valeur de la fonction de coût est prise en compte pour les dix exécutions. La conception obtenue est très performante. La valeur moyenne de la fonction de coût est la meilleure et offre un avantage significatif en termes de qualité de solution. Le temps de convergence est également très court, ce qui permet de relever le principal défi des coûts de l'apprentissage profond.

Les résultats de la comparaison sont présentés dans le tableau 5.6. Par rapport aux résultats précédemment publiés, la conception obtenue est plus optimisée, améliorant la figure de mérite, FoMs. Par exemple, notre méthode atteint la valeur maximale de $\mathrm{FoM_S}$, (273693), qui est très supérieure au meilleur résultat obtenu précédemment par la référence [Akbari, Biabanifard, et al. 2014], soit 872.22. Ce travail est également très bon en termes de produit gain bande passante GBW, de la marge de phase PM, de temps de stabilisation, de taux de rejection de mode commun CMRR et de dissipation de puissance,

Table 5.3: *Paramètres de conception typiques obtenus par la méthode proposée.*

Paramètre	valeur	Paramètre	valeur	Paramètre	valeur
$W_{0,1,2}$	13.5	$W_{3,4,5}$	1.0	$W_{6,7}$	3.5
$W_{8,9}$	2.0	$W_{10,11}$	10.0	$W_{12,13}$	10.5
$W_{14,15}$	17.0	$W_{16,17}$	17.5	$W_{18,19}$	2.0
$W_{20,21}$	2.5	$W_{22,23}$	16.5	W_{24}	11.0
W_{25}	13.0	W_{26}	15.0	W_{27}	7.5
W_{28}	9.5	$m_{0,1,2}$	8	$m_{3,4,5}$	8
$m_{6,7}$	1	$m_{8,9}$	19	$m_{10,11}$	13
$m_{12,13}$	17	$m_{14,15}$	12	$m_{16,17}$	17
$m_{18,19}$	1	$m_{20,21}$	3	$m_{22,23}$	9
m_{24}	8	m_{25}	4	m_{26}	12
m_{27}	6	m_{28}	8	C_C	7
R_C	9.8	I_{bias}	91		

W est en µm. R_C et C_C sont respectivement en kΩ et en pF. I_{bias} est en nA.

Table 5.4: *Exigences de conception de l'amplificateur cascade repliée.*

Performance	valeur	Performance	valeur
DC gain	76.4[dB]	GBW	16.23 [MHz]
PM	162.6[°]	CMRR	125.72 [dB]
PSRR	79.63[dB]	Settling time	100 [ns]
Offset	5.98 [µV]	Power	0.593 [µW]

Table 5.5: *Résultats statistiques des deux méthodes.*

méthode	algorithme évolutionnaire	Notre méthode
N-sim	10100	283
taux de réussite	10/10	10/10
Cost (Best)	0.7545	0.6542
Cost (Worst)	0.8426	0.8065
Cost (Mean)	0.8088	0.7279
Cost (Median)	0.8133	0.7368
Temps de convergence (heures)	2.21	0.405

qui ne sont pas inclus dans les **FoMs**.

Table 5.6: *Comparaison des performances des différentes conceptions de l'amplificateur cascade repliée.*

Metrics	Tech.	Supply	Gain	CMRR	PSRR	C_L	Power	GBW	PM	SR	t_{set}	FoM_S	FoM_L
Ref.	[nm]	[V]	[dB]	[dB]	[dB]	[pF]	[µW]	[MHz]	[°]	[V/µs]	[ns]	[MHz.pF/mW]	[V.pF/(µs.mW)]
[Qisheng Zhang et al. 2017]	180	1	50.7	NA	NA	15	192	1.15	88.7	0.81	716	89.84	63.28
[Qisheng Zhang et al. 2017]	180	1	57.6	NA	NA	15	192	2.25	87	1.24	307	**175.78**	96.87
[Tlelo-Cuautle et al. 2020]	180	±0.9	68	106.2	67	5	7000	67	66.51	17	NA	47.86	12.14
Ce travail	180	1.8	69.04	127.57	70.15	10	404.40	3.67	91	17.6	100	90.75	**435.21**

4 Conclusion

Nous avons proposé une solutions d'optimisation basé sur la simulation assistée par le **DNN** pour le dimensionnement automatique des circuits analogiques à haute performance. Cette outil se compose d'un moteur d'optimisation globale basé sur un algorithme évolutionnaire et d'un moteur d'optimisation locale basé sur un **DNN**. Cette approche est efficiente grâce au faible coût d'adaptation, à la forte capacité à gérer des spécifications strictes en utilisant un modèle **DNN** pour la prédiction de chaque performance. Elle offre également une forte capacité d'optimisation pour atteindre des performances élevées comme le démontre le cas de test. La méthode proposée permet d'atteindre le dimensionnement optimal en utilisant seulement quelques centaines de simulations avec une efficacité de 10-25 échantillons. Cette efficacité provient du modèle **DNN** utilisé pour le classement des conceptions candidates pour chaque performance.

Chapter

6 Layout du CCII : Post-layout Simulations et Discussions

Chapter content

1 Introduction 45
2 Création du Layout de CCII 47
3 Simulation post-layout et discussions 49
4 Conclusion 50

1 Introduction

Après avoir traité l'optimisation de dimensionnement des circuits intégrés analogiques dans les chapitres 3 à 5, ce chapitre traite des aspects fondamentaux de la conception de design des masques des circuits intégrés analogiques. Ces aspects sont essentiels pour comprendre le travail présenté dans le reste de ce manuscrit.

Le problème de la conception de design des masques consiste à transformer un circuit dimensionné en une représentation physique, appelée *layout*. Le but de cette représentation physique est de décrire et d'illustrer les géométries détaillées de la puce sur les masques photolithographiques nécessaires au processus de fabrication des semi-conducteurs. Ces géométries comprennent les designs des masques des composants du circuit, leurs interconnexions électriques avec des *vias* entre les couches isolantes... La figure 6.1 illustre le flux de création du *layout*.

Le circuit d'entrée peut être fourni sous forme du *netlist*, comme discuté dans le chapitre 4, il est généralement présenté sous forme de schéma visuellement accessible. Cela peut s'expliquer par deux raisons principales :

1) Tout d'abord, contrairement au monde numérique où les circuits sont synthétisés à partir de spécifications de haut niveau, les circuits analogiques sont créés manuellement par des experts en conception, nécessitant un format pratique pour la modification.

2) Ensuite, la transformation d'un circuit dimensionné en un *layout* fonctionnel implique la prise en compte de nombreuses contraintes et d'objectifs de conception qui

découlent de la compréhension de la fonctionnalité du circuit. Cela signifie que la lisibilité d'un schéma est non seulement essentielle pour le concepteur, mais aussi pour le concepteur du *layout*.

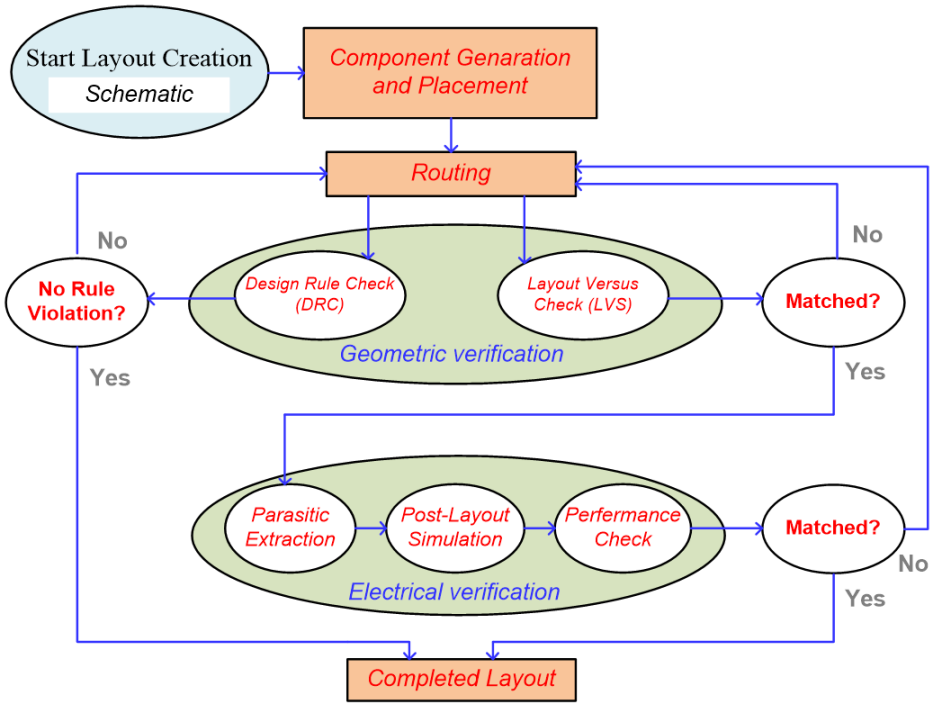


Figure 6.1: *Flux de conception du Layout des CIs.*

La création de dessin des masques des **CIs** est suivie d'une vérification géométrique et électrique pour vérifier la fonctionnalité du produit final. Cependant, le comportement des circuits électriques peut changer de manière non anticipée pendant le processus de dimensionnement des circuits. La vérification électrique est effectuée pour vérifier si le circuit après la création du *layout* satisfait aux spécifications électriques et aux contraintes définies. Pour effectuer et finaliser cette vérification électrique, un modèle électrique doit être extrait de la géométrie du *layout* ainsi que des informations technologiques.

D'autre part, la vérification géométrique garantit que la géométrie du *layout* respecte les règles spécifiques au processus de fabrication des **CIs**. Deux techniques essentiels de la vérification des *layouts* sont la verification **DRC** (pour Design Rules Check) et la verification **LVS** (pour Layout Versus Schematic).

- Design Rule Check:

  La vérification **DRC** évalue la géométrie physique par rapport à un ensemble de règles de conception prédéfinies qui dictent les dimensions minimales et maximales, les espacements et autres contraintes géométriques nécessaires à une fabrication correcte et à une fonctionnalité fiable. Il identifie et corrige les violations pour éviter des problèmes potentiels lors de la fabrication.

- Layout Versus Schematic Check:

  La vérification **LVS** vérifie que le *layout* correspond précisément au schéma du circuit prévu. Tout écart entre le *layout* et le schéma est signalé pour être analysée et corrigée ultérieurement.

2 Création du Layout de CCII

Dans cette section, nous présentons le *layout* du **CCII** qui est optimisé dans le Chapitre 3 du manuscrit. Le **CCII** est optimisé en utilisant la technologie **CMOS** 0.18µm au chapitre 3. La Table 6.1 rappelle les caractéristiques de conception et les paramètres de l'une des meilleures solutions du front de Pareto trouvées par notre méthode d'optimisation.

Table 6.1: *Caractéristiques et paramètres optimaux du CCII.*

paramètres	W_n [µm]	W_p [µm]	L_n [µm]	L_p [µm]	I_0 [µA]
Valeur	1.00	2.91	0.18	0.18	40

Caractéristiques	R_X[Ω]	f_{-3dB}[GHz]	consomation de puissance [µW]
Valeur	1444	3.82	120

Ensuite, nous procédons à la création du *layout*. Nous avons abordé la partie optimisation et venons de décrire les tailles des composants, comme indiqué dans les deux premières lignes du tableau, qui seront utilisées dans notre circuit. Maintenant, nous devons établir une méthode pour combiner ces deux aspects. Pour ce faire, Cadence Virtuoso met à notre *layout* un outil appelé '*Layout XL*'. Cet outil, après la création du schéma et les simulations, nous permet de générer un masque contenant les éléments du schéma avec les valeurs de leurs paramètres. La Figure 6.2 illustre le fonctionnement général du *Layout XL*.

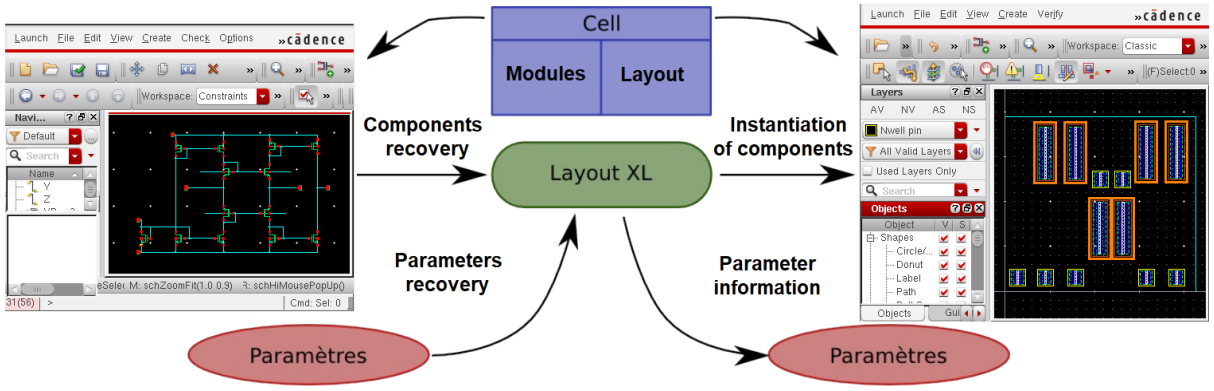


Figure 6.2: *Transfert des paramètres du schéma aux masques.*

Pour pouvoir utiliser *Layout XL*, nous devons d'abord associer le modèle technologique et regrouper dans la même "cellule" (dans le sens de Cadence) notre vue schématique et une vue du *Layout*. Ensuite, nous devons définir des paramètres dans les masques qui correspondent à ceux de la technologie du modèle. De cette manière, *Layout XL* instanciera le *layout* des composants utilisés dans le schéma tout en récupérant la valeur de chaque paramètre prédéfini dans le schéma.

Après l'instanciation des composants, nous procédons au placement et au routage, en tenant compte de toutes les règles et contraintes examinées ci-dessus. Le *layout* finale du **CCII** est illustrée dans la Figure 6.3, avec tous les composants et les differentes broches d'entrée et de sortie. Les transistors PNOS sont placés dans la moitié supérieure et les transistors NMOS sont placés dans la moitié inférieure du *layout*. Dans les sections NMOS et PMOS, des miroirs de courant à source commune assurent une symétrie par la technique de centroïde commun. Ainsi, nous garantissons qu'ils sont uniformément soumis aux erreurs et variations du processus, préservant ainsi leur appariement tout au long des

étapes de fabrication. De plus, nous utilisons simplement deux couches métalliques : métal-1 (en couleur bleue) pour le routage horizontal et métal-2 (en couleur rouge) pour le routage vertical. Limiter le nombre de couches métalliques simplifie le processus de fabrication, le rendant plus rentable et moins sujet aux erreurs. Cela réduit également le nombre de vias et de contacts métalliques, ce qui à son tour diminue la résistance et la capacitance parasites.

Une autre technique largement appliquée est l'utilisation de structures factices "dummy devices". Ici, dans la section NMOS, nous avons ajouté un transistor NMOS ("dummy transistor") pour améliorer la symétrie, en d'autres termes, pour garantir que chaque élément actif interagit de manière symétrique avec les éléments adjacents.

Les transistors sont entourés par des anneaux de garde qui sont connectées à la tension la plus basse (VSS) dans le circuit pour les transistors NMOS et à la tension la plus élevée (VDD) pour les transistors PMOS. Ces anneaux de garde assurent une bonne isolation des transistors et réduisent ainsi les interférences électromagnétiques et le couplage entre les signaux haute fréquence. De plus, elles empêchent les fuites de courant des couches de diffusion à travers le substrat, ce qui peut entraîner des modes de fonctionnement indésirables des transistors. La surface totale de la puce est seulement de $13.43 \times 14.84 \mu m^2$.

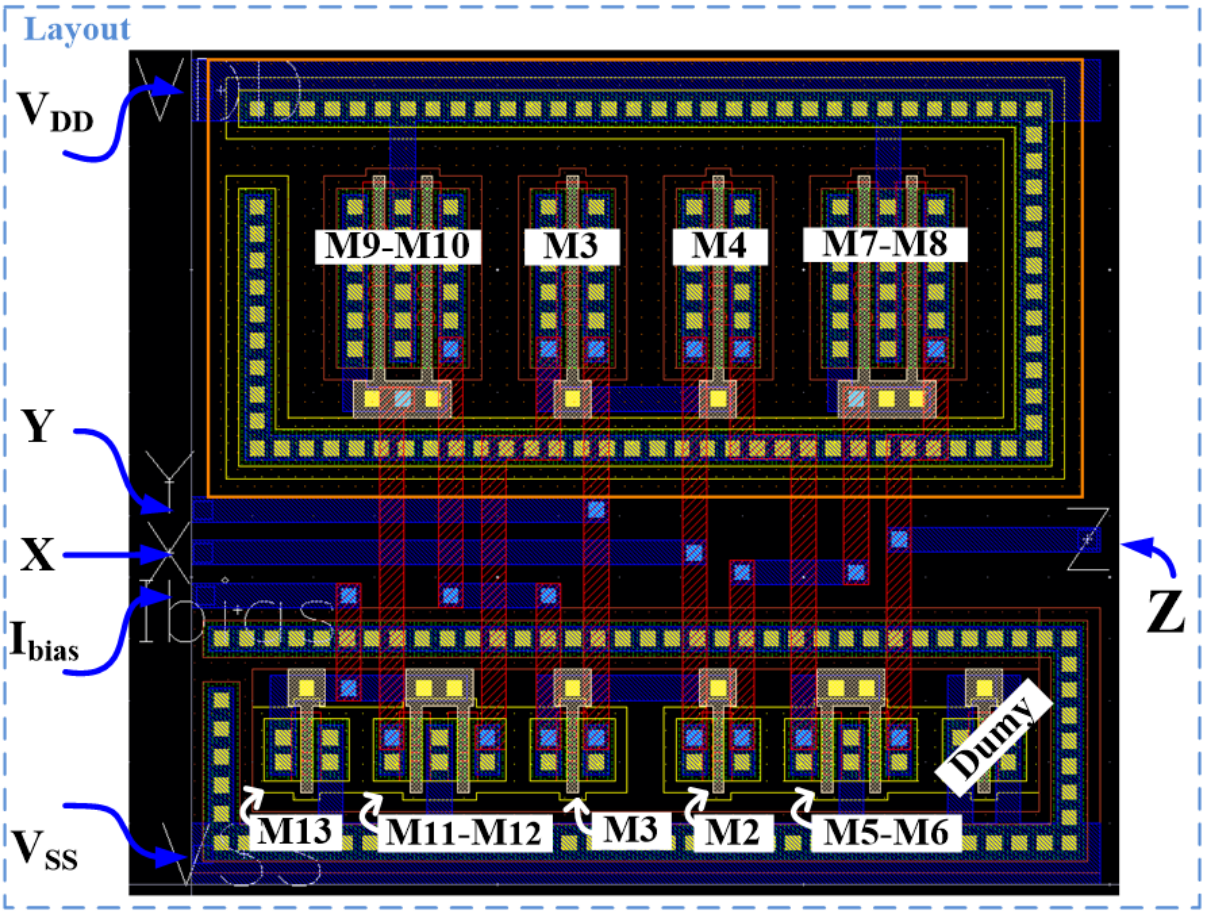


Figure 6.3: *Schéma du Courant Convoyeur CMOS optimisé.*

3 Simulation post-layout et discussions

Après avoir achevé la conception du *layout*, il est essentiel de vérifier s'il respecte les règles de conception. Cela se fait à travers une vérification *Design Rules Check (**DRC**)* pour s'assurer de la conformité géométrique, et une vérification *Layout Versus Schematic (**LVS**)* pour vérifier la correspondance avec le schéma. Ces deux types de vérifications sont cruciaux dans le processus de conception. Dans notre cas et après avoir effectué ces vérifications, le *layout* final du **CCII** est un "layout parfaitement ordonnée" (sans erreurs **DRC**, correcte **LVS** et conforme aux contraintes).

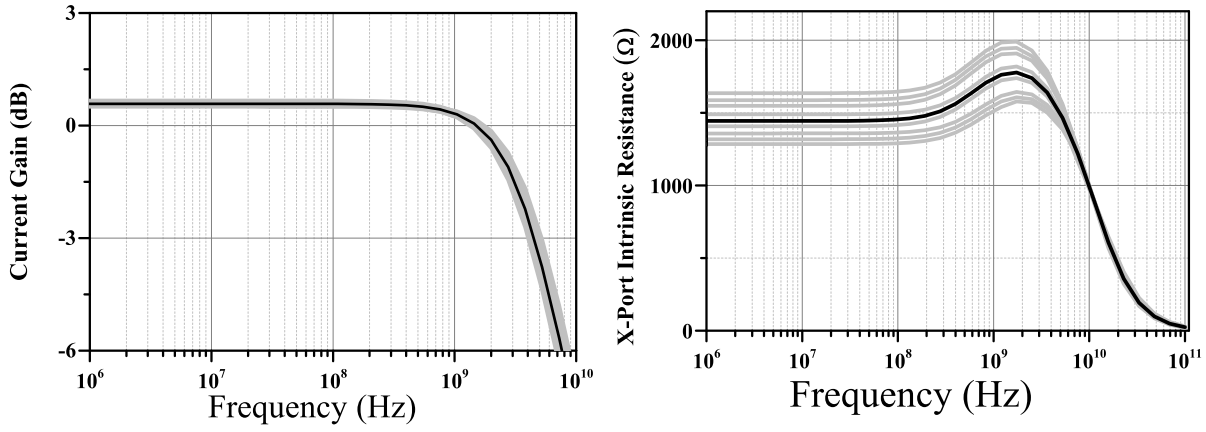
Ensuite, intervient la vérification finale, la vérification électrique, qui a pour objectif de s'assurer que le circuit conserve la fonctionnalité désirée après le *layout*. Pour ce faire, un modèle électrique est dérivé de la géométrie du *layout* et des informations technologiques.

La comparaison de la simulation post-layout et de la simulation pré-layout est effectuée, et les résultats sont representés dans le Tableau 6.2. Il faut garder à l'esprit que les effets parasites deviennent significatifs à des fréquences gigahertz. La différence entre les deux résultats est faible, avec un écart maximal de 4.37%, ce qui est bon à haute fréquence. Cette disparité est due au fait que les transistors, dans leur représentation schématique, modélisent uniquement les caractéristiques intrinsèques du dispositif, sans inclure les éléments parasites inhérents au dispositif réel. Il convient de noter qu'à des fréquences plus basses, on s'attend à ce que ces effets parasites aient un effet négligeable.

Table 6.2: *Résultats de comparaison entre la pré-layout et la post-layout du CCII.*

	Simulation re-layout	Simulation post-layout	Écart en pourcentage [%]
$R_X[\Omega]$	1444.44	1444.44	0
$f_{\text{-3dB}}$[GHz]	3.82	3.66	4.37

La réponse en fréquence de la simulation post-layout est présentée dans la Figure **??**. La courbe noire représente la simulation typique (processus typique, ±1.5V, 27°C). De



(a) Réponse de la fréquence de coupure.

(b) Résistance intrinsèque.

Les courbes noires résultent de la simulation typique, tandis que les courbes grises sont fournies par les simulations PVT.

Figure 6.4: *Analyse PVT pour le layout final du CCII.*

plus, des courbes grises sont incluses, résultant d'une analyse **PVT** avec des variations de tensions d'alimentation (±1.4V, ±1.5V et ±1.6V) et de températures (-55°C, +27°C, +100°C).

Les résultats de simulation pour la résistance R_X sont présentés dans la Figure **??**. Les mêmes annotations sont utilisées pour les courbes en gris et en noir. Pour le coin typique, la fréquence de coupure et la résistance intrinsèque au port X sont respectivement de 3.82GHz et 1444.44Ω. Ces valeurs présentent une variation maximale de 0.68GHz et 191.1Ω, respectivement.

De plus, afin d'étudier l'effet des variations de processus sur les performances du **CCII** après *layout*, des simulations pour différents coins de processus sont effectuées, et les résultats sont résumés dans le Tableau 6.3. Pour la simulation au coin typique, la fréquence de coupure f_{3dB} et R_X sont respectivement de 3.82GHz et 1444.44Ω, et présentent une variation maximale de 0.81GHz et 217.34Ω, respectivement.

Table 6.3: *Performances en fonction des variations de processus.*

T = 27°C	FF	FS	TT	SF	FF
$R_X(\Omega)$	1252.82	1484.94	1444.4	1440.31	1661.78
$f_{\text{-3dB}}$(GHz)	4.63	3.69	3.82	4.15	3.37

T signifie typique, S signifie lent et F signifie rapide. Par exemple, SF signifie NMOS lent et PMOS rapide..

4 Conclusion

Dans ce chapitre, le *layout* du **CCII** pré-dimensionné est présenté. Ensuite, l'extraction des parasites est abordée, suivie de simulations post-layout pour évaluer leur impact sur les performances du circuit. Des discussions approfondies ont permis d'analyser les résultats de simulation et de comprendre les avantages des méthodes proposées. Enfin, les éventuels défis rencontrés lors de la conception et de la simulation ont été traités pour enrichir la compréhension des conclusions.

Conclusion générale et perspectives

La conception automatisée d'algorithmes est un domaine de recherche relativement nouveau mais qui connaît une croissance rapide. Il s'est maintenant développé au point où des algorithmes efficaces sont devenus disponibles et sont effectivement utilisés dans la pratique. L'optimisation multi-objectifs est également une direction récente dans laquelle la conception automatisée se développe, avec un nouvel intérêt des deux communautés. En électronique, le dimensionnement des circuits analogiques peut être considéré comme un problème non linéaire complexe avec de multiples critères, de nombreuses variables et des contraintes. En général, les concepteurs doivent consacrer beaucoup de temps à acquérir suffisamment d'expérience sur un circuit donné pour répondre à toutes les exigences de conception. Par conséquent, nous croyons fermement que cette direction de travail apportera un grand bénéfice aux concepteurs de circuits analogiques. Elle leur permettra d'obtenir des performances encore meilleures et de mieux comprendre le comportement des circuits.

Ainsi, afin de répondre aux besoins de l'industrie en matière de méthodes numériques robustes et puissantes pour l'optimisation optimale des circuits analogiques, nous avons adapté et présenté plusieurs approches. Dans le troisième chapitre, nous avons introduit un nouvel algorithme d'optimisation inspiré des papillons. Nous l'avons adapté ici à un problème multi-objectif et ensuite nous l'avons testé sur le **CCII**. Nous avons discuté de l'impact de la longueur de canal et du courant de polarisation sur les performances du circuit. Nos résultats ont démontré que l'algorithme proposé est plus performant que les méthodes d'optimisation classiques, atteignant de bons compromis dans des paramètres tels que la fréquence de coupure, la superficie et la consommation de puissance. Ces résultats ont été publiés dans le "Microelectronics Journal" (https://doi.org/10.1016/j.mejo.2021.105078).

Dans le quatrième chapitre, nous avons introduit un outil efficace d'optimisation basée sur la simulation multi-objectif. Nous avons appliqué à quatre *Integrated Circuits (**ICs**)* analogiques différents, dont **CCII**, *Convoyeur de courant de deuxième génération à haute précision (**HCCII**)*, **OTA** et **CFOA**. Dans ce résumé, nous avons présenté que les résultats du premier circuit. Nous avons exploré deux types d'application : les applications biomédicales portables où les transistors sont polarisés en sous le seuil et les applications haute performance utilisant des transisteurs polarisés linéaires/saturation. Nous avons évalué la méthode proposée en utilisant des métriques de performance

telles que le temps de convergence, la métrique C, et les statistiques d'hyper-volume, démontrant sa supériorité par rapport à d'autre algorithme bien connus. Les résultats ont fourni des orientations précieuses pour les concepteurs de circuits dans différentes topologies, nombres de variables et modes de fonctionnement des transistors. Ces résultats ont été publiés dans "International Journal of Electronics and Communications" (https://doi.org/10.1016/j.aeue.2021.154023).

Dans le cinquième chapitre, nous avons introduit un nouveau environemment d'optimisation qui combine l'optimisation globale basée sur les algorithmes evolutionnaires avec l'optimisation locale basé sur les Deep Neural Networks(**DNN**). Cette approche a montré un faible coût de calcul, des performances élevées et la capacité à gérer des spécifications strictes en utilisant des modèles **DNN** pour la prédiction des performances. Quatre exemples de tests ont mis en évidence l'efficacité de l'environnement pour la conception de circuits à haute performance, tout en relevant les défis liés aux techniques d'optimisation assistées par l'intelligence artificielle. De même, nous avons présentons ici que les résultats du **Op-Amp** cascode Replié. Nos résultats ont montré des avantages significatifs par rapport aux approches classiques, en particulier en réduisant le temps de calcul et en minimisant les erreurs de entre es résultats de simulation et des modeles. Ces découvertes ont été publiées dans "Expert Systems With Applications" (https://doi.org/10.1016/j.eswa.2023.121480).

Cette étude ouvre de nombreuses perspectives de recherche :

- D'un point de vue pratique, il pourrait être intéressant d'appliquer le calcul parallèle. En comparaison avec le problème abordé au chapitre 1, le temps de convergence sera considérablement réduit, et le résultat de sortie pourrait toujours être efficace.
- Les techniques proposées dans ce manuscrit contribuent à l'optimisation des circuits analogiques et à une meilleure compréhension des compromis nécessaires pour atteindre les meilleures performances en tenant compte de nombreuses contraintes, ainsi que des relations d'appariement et des règles de conception. Cependant, elles ne tiennent pas compte la robustesse requise à long terme (dégradation des porteurs chauds) et à court terme (électrique et thermique) de la Zone de Fonctionnement Sûre (SOA). Par conséquent, une direction intéressante pour les recherches futures serait d'effectuer l'optimisation en tenant compte d'un critère de risque de la SOA.

Bibliography

References for Chapter : Introduction générale

Kumar, Priyan Malarvizhi et al. (2018). "Ant colony optimization algorithm with internet of vehicles for intelligent traffic control system". In: *Computer Networks* 144, pp. 154–162.

Ahmadyan, Seyed Nematollah (2016). "Validation and optimization of analog circuits using randomized search algorithms". PhD thesis. University of Illinois at Urbana-Champaign.

Boyd, Stephen et al. (2007). "A tutorial on geometric programming". In: *Optimization and engineering* 8.1, pp. 67–127.

Jafari, Ali et al. (2010). "Design optimization of analog integrated circuits by using artificial neural networks". In: *2010 international conference of soft computing and pattern recognition*. IEEE, pp. 385–388.

Lin, Honghuang and Peng Li (2015). "Circuit performance classification with active learning guided sampling for support vector machines". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34.9, pp. 1467–1480.

Sallem, Amin et al. (2013). "A Multi-objective simulation based tool: application to the design of high performance LC-VCOs". In: *Doctoral Conference on Computing, Electrical and Industrial Systems*. Springer, pp. 459–468.

Liu, Bo, Francisco V Fernández, et al. (2009). "A memetic approach to the automatic design of high-performance analog integrated circuits". In: *ACM Transactions on Design Automation of Electronic Systems (TODAES)* 14.3, pp. 1–24.

Tlelo-Cuautle, Esteban et al. (2020). "Sizing CMOS amplifiers by PSO and MOL to improve DC operating point conditions". In: *Electronics* 9.6, p. 1027.

Castañeda-Aviña, Perla Rubi et al. (2020). "Single-Objective Optimization of a CMOS VCO Considering PVT and Monte Carlo Simulations". In: *Mathematical and Computational Applications* 25.4, p. 76.

Boyd, Stephen P, Thomas H Lee, et al. (2001). "Optimal design of a CMOS op-amp via geometric programming". In: *IEEE Transactions on Computer-aided design of integrated circuits and systems* 20.1, pp. 1–21.

Fakhfakh, Mourad et al. (2010). "Analog circuit design optimization through the particle swarm optimization technique". In: *Analog integrated circuits and signal processing* 63.1, pp. 71–82.

Akbari, Meysam, Mohammad Shokouhifar, et al. (2016). "Systematic design of analog integrated circuits using ant colony algorithm based on noise optimization". In: *Analog Integrated Circuits and Signal Processing* 86.2, pp. 327–339.

Arora, Sankalap and Satvir Singh (2019). "Butterfly optimization algorithm: a novel approach for global optimization". In: *Soft Computing* 23.3, pp. 715–734.

Cao, Weidong et al. (2022). "Domain Knowledge-Based Automated Analog Circuit Design with Deep Reinforcement Learning". In: *arXiv preprint arXiv:2202.13185*.

Lyu, Wenlong et al. (2017). "An efficient bayesian optimization approach for automated optimization of analog circuits". In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 65.6, pp. 1954–1967.

Liu, Bo, Dixian Zhao, et al. (2014). "GASPAD: A general and efficient mm-wave integrated circuit synthesis method based on surrogate model assisted evolutionary algorithm". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 33.2, pp. 169–182.

Liu, Bo et al. (2013). "A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems". In: *IEEE Transactions on Evolutionary Computation* 18.2, pp. 180–192.

Kotti, Mouna et al. (2018). "Kriging Metamodeling-Assisted Multi-Objective Optimization of CMOS Current Conveyors". In: *2018 15th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*. IEEE, pp. 293–296.

Garbaya, Amel, Mouna Kotti, Mourad Fakhfakh, et al. (2020). "Surrogate Assisted Optimization for Low-Voltage Low-Power Circuit Design". In: *Journal of Low Power Electronics and Applications* 10.2, p. 20.

Garbaya, Amel, Mouna Kotti, Nawel Drira, et al. (2018). "An RBF-PSO technique for the rapid optimization of (CMOS) analog circuits". In: *2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST)*. IEEE, pp. 1–4.

Lourenço, Nuno et al. (2019). "Using polynomial regression and artificial neural networks for reusable analog ic sizing". In: *2019 16th International Conference on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD)*. IEEE, pp. 13–16.
Li, Yaping et al. (2019). "An artificial neural network assisted optimization system for analog design space exploration". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 39.10, pp. 2640–2653.
Shadab, Syed et al. (2022). "Gaussian process surrogate model for an effective life assessment of transformer considering model and measurement uncertainties". In: *International Journal of Electrical Power & Energy Systems* 134, p. 107401.
Zhang, Shuhan et al. (2019). "Bayesian optimization approach for analog circuit synthesis using neural network". In: *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, pp. 1463–1468.
He, Biao et al. (2020). "An efficient bayesian optimization approach for analog circuit synthesis via sparse gaussian process modeling". In: *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, pp. 67–72.
Liu, Bo, Noël Deferm, et al. (2012). "An efficient high-frequency linear RF amplifier synthesis method based on evolutionary computation and machine learning techniques". In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 31.7, pp. 981–993.
Wang, Hanrui et al. (2020). "GCN-RL circuit designer: Transferable transistor sizing with graph neural networks and reinforcement learning". In: *2020 57th ACM/IEEE Design Automation Conference (DAC)*. IEEE, pp. 1–6.
Settaluri, Keertana et al. (2020). "AutoCkt: deep reinforcement learning of analog circuit designs". In: *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, pp. 490–495.

References for Chapter 1: Optimisation et Métaheuristiques

Tlelo-Cuautle, Esteban et al. (2020). "Sizing CMOS amplifiers by PSO and MOL to improve DC operating point conditions". In: *Electronics* 9.6, p. 1027.
Kaveh, Ali (2014). *Advances in metaheuristic algorithms for optimal design of structures*. Springer.
Lberni, Abdelaziz et al. (2021). "Efficient butterfly inspired optimization algorithm for analog circuits design". In: *Microelectronics Journal* 113, p. 105078. ISSN: 0026-2692. DOI: https://doi.org/10.1016/j.mejo.2021.105078.
Ma, Xiaoliang et al. (2017). "On Tchebycheff decomposition approaches for multiobjective evolutionary optimization". In: *IEEE Transactions on Evolutionary Computation* 22.2, pp. 226–244.
Zhang, Qingfu and Hui Li (2007). "MOEA/D: A multiobjective evolutionary algorithm based on decomposition". In: *IEEE Transactions on evolutionary computation* 11.6, pp. 712–731.
Choi, Kyung et al. (2015). "Hybrid algorithm combing genetic algorithm with evolution strategy for antenna design". In: *IEEE transactions on magnetics* 52.3, pp. 1–4.
Zhao, Shuguang et al. (2005). "Evolutionary design of analog circuits with a uniform-design based multi-objective adaptive genetic algorithm". In: *2005 NASA/DoD Conference on Evolvable Hardware (EH'05)*. IEEE, pp. 26–29.
Kchaou, O Bellaaj et al. (2016). "Sensitivity aware NSGA-II based Pareto front generation for the optimal sizing of analog circuits". In: *Integration* 55, pp. 220–226.
Patel, Subhash J and Rajesh A Thakker (2018). "Parasitic aware automatic analog CMOS circuit design environment using ABC algorithm". In: *2018 31st International Conference on VLSI Design and 2018 17th International Conference on Embedded Systems (VLSID)*. IEEE, pp. 445–446.

References for Chapter 2: Réseaux de neurones et leur application

Gurney, Kevin (2018). *An introduction to neural networks*. CRC press.
Patterson, Josh and Adam Gibson (2017). *Deep learning: A practitioner's approach*. " O'Reilly Media, Inc.".

References for Chapter 3: Adaptation de BOA pour la Conception de Circuits Analogiques

Lberni, Abdelaziz et al. (2021). "Efficient butterfly inspired optimization algorithm for analog circuits design". In: *Microelectronics Journal* 113, p. 105078. ISSN: 0026-2692. DOI: https://doi.org/10.1016/j.mejo.2021.105078.
Soliman, Ahmed M and Ramy A Saad (2010). "Generation of second generation current conveyor (CCII) family from inverting second generation current conveyor (ICCII) family". In: *International journal of electronics* 97.4, pp. 405–414.
Lima Moreto, Rodrigo Alves de et al. (2019). "A customized genetic algorithm with in-loop robustness analyses to boost the optimization process of analog CMOS ICs". In: *Microelectronics Journal* 92, p. 104595.
Sibalija, Tatjana V (2019). "Particle swarm optimisation in designing parameters of manufacturing processes: A review (2008–2018)". In: *Applied Soft Computing* 84, p. 105743.
Zhang, Xin and Chunhua Wang (2019). "A novel multi-attractor period multi-scroll chaotic integrated circuit based on CMOS wide adjustable CCCII". In: *IEEE Access* 7, pp. 16336–16350.
El Dor, Abbas et al. (2016). "Multiobjective differential evolution algorithm using crowding distance for the optimal design of analog circuits". In: *Journal of Electrical Systems* 12.3, pp. 612–622.
— (2014). "Performance optimization of CMOS second generation current conveyors using a multi-swarm algorithm". In: *AEU-International Journal of Electronics and Communications* 68.6, pp. 496–503.
Khateb, Fabian et al. (2019). "0.3 V bulk-driven current conveyor". In: *IEEE Access* 7, pp. 65122–65128.

References for Chapter 4: Méthode multi-objectifs basée sur la simulation pour la conception des circuits analogiques

Azadeh, A et al. (2017). "Unique NSGA-II and MOPSO algorithms for improved dynamic cellular manufacturing systems considering human factors". In: *Applied Mathematical Modelling* 48, pp. 655–672.
Yucehan, Tolga and Erkan Yuce (2022). "CCII-based voltage-mode and current-mode high-order filters with gains and grounded passive elements only". In: *AEU-International Journal of Electronics and Communications* 155, p. 154346.
Sharan, Tripurari et al. (2020). "Low-Power Bulk-Driven Subthreshold OTA and CCII+ Cells Based Instrumentation Amplifier for LF Applications". In: *Proceedings of the 5th International Conference on Computers & Management Skills (ICCM 2019)| North Eastern Regional Institute of Science & Technology (NERIST), Nirjuli, Arunachal Pradesh, India.*
Ferri, Giuseppe et al. (2009). "A CCII-based low-voltage low-power read-out circuit for DC-excited resistive gas sensors". In: *IEEE sensors journal* 9.12, pp. 2035–2041.

References for Chapter 5: Optimisation assistée par les réseaux de neurones profonds

Tlelo-Cuautle, Esteban et al. (2020). "Sizing CMOS amplifiers by PSO and MOL to improve DC operating point conditions". In: *Electronics* 9.6, p. 1027.
Akbari, Meysam, Sadegh Biabanifard, et al. (2014). "Design and analysis of DC gain and transconductance boosted recycling folded cascode OTA". In: *AEU-International Journal of Electronics and Communications* 68.11, pp. 1047–1052.
Zhang, Qisheng et al. (2017). "Multipath recycling method for transconductance enhancement of folded cascade amplifier". In: *AEU-International Journal of Electronics and Communications* 72, pp. 1–7.