

**École Nationale Polytechnique**



**Département d'Automatique**  
Laboratoire de Commande des Processus

**Thèse de Doctorat**

Spécialité : Automatique

**Contribution à l'optimisation multiobjectif par  
métaheuristiques : application à la synthèse des PSS  
dans les réseaux électriques**

Présentée par

**Hilal LABDELAOUI**

Soutenue publiquement le 4 Décembre 2019 devant le jury composé de :

A. HELLAL	Professeur à l'ENP	Président
F. BOUDJEMA	Professeur à l'ENP	Directeur de thèse
D. BOUKHETALA	Professeur à l'ENP	Co-directeur de thèse
L. NEZLI	Professeur à l'ENP	Examineur
A. BENALIA	Professeur à l'UATL	Examineur
H. RADJELI	Professeur à l'UFAS	Examineur
S. BENAHDUGA	Maître de conférences à l'UBBA	Examineur



**École Nationale Polytechnique**



**Département d'Automatique**  
Laboratoire de Commande des Processus

**Thèse de Doctorat**

Spécialité : Automatique

**Contribution à l'optimisation multiobjectif par  
métaheuristiques : application à la synthèse des PSS  
dans les réseaux électriques**

Présentée par

**Hilal LABDELAOUI**

Soutenue publiquement le 4 Décembre 2019 devant le jury composé de :

A. HELLAL	Professeur à l'ENP	Président
F. BOUDJEMA	Professeur à l'ENP	Directeur de thèse
D. BOUKHETALA	Professeur à l'ENP	Co-directeur de thèse
L. NEZLI	Professeur à l'ENP	Examineur
A. BENALIA	Professeur à l'UATL	Examineur
H. RADJELI	Professeur à l'UFAS	Examineur
S. BENAHDUGA	Maître de conférences à l'UBBA	Examineur

# Remerciements

Je voudrai en premier lieu remercier chaleureusement Monsieur Farès Boudjema, Professeur à l'École Nationale Polytechnique, et Monsieur Djamel Boukhetala, Professeur à l'École Nationale Polytechnique, qui ont bien voulu prendre en charge la direction scientifique de ce travail. Leurs conseils judicieux, leurs disponibilités et leurs encouragements m'ont beaucoup aidé pour l'aboutissement de ce travail. Qu'ils trouvent ici l'expression de ma reconnaissance la plus sincère.

Je tiens ensuite à remercier Monsieur Abdelhafid Hellal, Professeur à l'École Nationale Polytechnique, pour l'honneur qu'il me fait en acceptant la présidence du jury de cette thèse. Mes remerciements s'adressent également aux membres du jury qui ont bien voulu me faire l'honneur d'examiner ce travail; Monsieur Lazhari Nezli, Professeur à l'École Nationale Polytechnique, Monsieur Atallah Benalia, Professeur à l'Université Amar Telidji de Laghouat, Monsieur Hammoud Radjeai, Professeur à l'Université Ferhat Abbas de Sétif et Monsieur Seddik Benahdouga, Maître de conférences à l'Université Mohamed El-Bachir El-Ibrahimi de Bordj Bou Arréridj.

Je remercie également tous mes collègues et mes amis qui m'ont encouragé et soutenu durant la préparation de cette thèse. Je cite en particulier K. Ghaoui, N. Bali, A. Nait Seghir, L. Barazane, R. Dermouche, F. Amrani, S. Labandji et A. Benhamouda.

Enfin, ce travail n'aurait pas pu être réalisé sans le soutien de ma famille, en particulier mes parents, mes frères, ma soeur et ma femme. Qu'ils trouvent en cette thèse le modeste témoignage de ma profonde reconnaissance.

## ملخص

يتناول هذا العمل طرق تحسين ثبات الأنظمة الكهربائية باستخدام الأدلة العليا. نقوم أولاً بتصميم خوارزمية جديدة متعددة الأهداف تعمل بتقنية التطور التفاضلي. ثم من أجل تقييم فعالية هذه الخوارزمية، نقوم بمقارنتها ببعض الخوارزميات الأخرى الفعالة عن طريق حل بعض المسائل النظرية. فيما بعد، نقترح تطبيق جميع هذه الخوارزميات على ثلاث مسائل متعددة الأهداف تتعلق بتعديل مثبتات الطاقة من أجل تحسين ثبات الأنظمة الكهربائية. في الأخير، نستعمل تجارب التحسين بالإضافة إلى المحاكاة الزمنية من أجل التأكد من فعالية طرق البحث المقترحة في هذا العمل.

**كلمات مفتاحية :** الأنظمة الكهربائية، مثبتات الطاقة، التحسين متعدد الأهداف، الأدلة العليا.

## Abstract

This work deals with the problem of stability enhancement of electric power systems. First, a new multiobjective optimization algorithm, based on differential evolution, is developed and compared to other well-known optimization methods using benchmark test functions. All these algorithms are then used to solve three multiobjective optimal tuning problems of power system stabilisers in order to enhance the stability of power systems. The efficiency of the proposed algorithm and the design approaches are validated through optimization tests and numerical simulations.

**Keywords :** Electric power systems, power system stabilizers, multiobjective optimization, metaheuristics.

## Résumé

Ce travail porte sur la résolution des problèmes d'optimisation de la stabilité des réseaux électriques au moyen des métaheuristiques. Dans un premier temps, nous développons un nouvel algorithme d'optimisation multiobjectif en se basant sur la technique de l'évolution différentielle. Afin de valider les performances de cet algorithme, une étude comparative avec d'autres algorithmes de référence est effectuée en considérant la résolution de plusieurs problèmes académiques. Ensuite, nous proposons la résolution de trois problèmes multiobjectifs qui concernent la synthèse optimale des stabilisateurs de puissance afin d'améliorer la stabilité des réseaux électriques. Les performances de l'algorithme ainsi que des approches de résolution proposés sont validés à travers des tests d'optimisation et des simulations numériques.

**Mots clés :** Réseaux électriques, stabilisateurs de puissance, optimisation multiobjectif, métaheuristiques.

# Table des matières

Table des figures

Liste des tableaux

<b>Chapitre 1 : Introduction</b>	11
1.1 L'optimisation multiobjectif . . . . .	11
1.1.1 Formulation mathématique . . . . .	11
1.1.2 Classification des méthodes de résolution . . . . .	13
1.2 Les métaheuristiques . . . . .	14
1.2.1 Classification des métaheuristiques . . . . .	15
1.2.2 Choix d'une métaheuristique . . . . .	15
1.3 Synthèse optimale des stabilisateurs de puissance dans les réseaux électriques	16
1.3.1 Les stabilisateurs de puissance . . . . .	17
1.3.2 Dimensionnement optimal des PSS . . . . .	17
1.4 Objectifs du travail . . . . .	19
1.5 Organisation du document . . . . .	19
<b>Chapitre 2 : Optimisation multiobjectif par métaheuristiques</b>	21
2.1 Introduction . . . . .	21
2.2 Non-dominated Sorting Genetic Algorithm II (NSGA-II) . . . . .	21
2.2.1 Les algorithmes génétiques . . . . .	21
2.2.2 Description de l'algorithme NSGA-II . . . . .	23
2.3 Generalized Differential Evolution 3 (GDE3) . . . . .	25
2.3.1 L'évolution différentielle . . . . .	25
2.3.2 Description de l'algorithme GDE3 . . . . .	27
2.4 Archived Multiobjective Simulated Annealing . . . . .	27
2.4.1 Le recuit simulé . . . . .	28
2.4.2 Description de l'algorithme AMOSA . . . . .	29
2.5 Speed-Constrained Particle Swarm Optimization (SMPSO) . . . . .	31
2.5.1 L'optimisation par essaims de particules . . . . .	31
2.5.2 Description de l'algorithme SMPSO . . . . .	33
2.6 Conclusion . . . . .	34
<b>Chapitre 3 : Développement d'un nouvel algorithme à évolution différentielle pour l'optimisation multiobjectif</b>	36
3.1 Introduction . . . . .	36

3.2	L'algorithme Multi-Objective Differential Evolution using two Archives (MODE-2A) . . . . .	36
3.2.1	Description de l'algorithme MODE-2A . . . . .	37
3.2.2	Discussion . . . . .	38
3.3	Evaluation de l'algorithme MODE-2A . . . . .	39
3.3.1	Problèmes de test . . . . .	39
3.3.2	Métriques de performance . . . . .	44
3.3.3	Résultats de simulation . . . . .	46
3.4	Conclusion . . . . .	50
 <b>Chapitre 4 : Applications aux problèmes de synthèse des PSS dans les réseaux électriques</b>		 58
4.1	Introduction . . . . .	58
4.2	Application à un réseau mono-machine . . . . .	58
4.2.1	Formulation du problème . . . . .	58
4.2.2	Résultats de l'optimisation . . . . .	59
4.2.3	Résultats de simulation . . . . .	61
4.2.4	Sélection d'une solution de compromis . . . . .	63
4.3	Application à un réseau à 3 machines . . . . .	65
4.3.1	Formulation du problème . . . . .	65
4.3.2	Résultats de l'optimisation . . . . .	68
4.3.3	Résultats de simulation . . . . .	69
4.3.4	Sélection d'une solution de compromis . . . . .	73
4.4	Application à un réseau à 10-machines . . . . .	79
4.4.1	Formulation du problème . . . . .	79
4.4.2	Résultats de l'optimisation . . . . .	79
4.4.3	Résultats de simulation . . . . .	81
4.4.4	Sélection d'une solution de compromis . . . . .	82
4.5	Conclusion . . . . .	86
 <b>Chapitre 5 : Conclusions et perspectives</b>		 91
5.1	Synthèse des travaux . . . . .	91
5.2	Perspectives . . . . .	93
 <b>Bibliographie</b>		 101
 <b>Annexe A : Modèle dynamique de la machine</b>		 102
 <b>Annexe B : Données du réseau mono-machine</b>		 104
 <b>Annexe C : Données du réseau à 3 machines</b>		 105
 <b>Annexe D : Données du réseau à 10 machines</b>		 107

# Table des figures

Figure 1.1 : Illustration des notions de dominance et d'optimalité de Pareto. . . . .	13
Figure 1.2 : Classification des métaheuristiques. . . . .	16
Figure 2.1 : Illustration de la procédure de ranking. . . . .	24
Figure 2.2 : Calcul de la distance de crowding. . . . .	25
Figure 2.3 : Opérateur de l'évolution différentielle. . . . .	26
Figure 2.4 : Illustration de la procédure de clustering ( $N = 5$ ). . . . .	31
Figure 2.5 : Déplacement d'une particule dans l'OEP. . . . .	32
Figure 3.1 : Fonctionnement de l'algorithme MODE-2A pour une itération. . . . .	38
Figure 3.2 : Réduction de la taille de l'archive ( $N = 5$ ). . . . .	39
Figure 3.3 : Aspects de la qualité des solutions non dominées. . . . .	45
Figure 3.4 : Illustration du calcul de l'hypervolume. . . . .	46
Figure 3.5 : Illustration du calcul de la métrique $\bar{d}_{\max}$ . . . . .	47
Figure 3.6 : Comparaison des résultats du problème ZDT1. . . . .	51
Figure 3.7 : Comparaison des résultats du problème ZDT2. . . . .	51
Figure 3.8 : Comparaison des résultats du problème ZDT3. . . . .	52
Figure 3.9 : Comparaison des résultats du problème ZDT4. . . . .	52
Figure 3.10 : Comparaison des résultats du problème ZDT6 (a). . . . .	53
Figure 3.11 : Comparaison des résultats du problème ZDT6 (b). . . . .	53
Figure 3.12 : Fronts non dominés du problème DTLZ1. . . . .	54
Figure 3.13 : Fronts non dominés du problème DTLZ2. . . . .	54
Figure 3.14 : Fronts non dominés du problème DTLZ3. . . . .	54
Figure 3.15 : Fronts non dominés du problème DTLZ4. . . . .	55
Figure 3.16 : Fronts non dominés du problème DTLZ5. . . . .	55
Figure 3.17 : Fronts non dominés du problème DTLZ6. . . . .	55
Figure 3.18 : Fronts non dominés du problème DTLZ7. . . . .	56
Figure 3.19 : Fronts non dominés du problème DTLZ8. . . . .	56
Figure 3.20 : Fronts non dominés du problème DTLZ9. . . . .	56
Figure 4.1 : Réseau mono-machine. . . . .	59
Figure 4.2 : Variation de l'hypervolume. . . . .	60
Figure 4.3 : Front non dominé. . . . .	61
Figure 4.4 : Réponses du système. . . . .	62
Figure 4.5 : Tensions d'excitation. . . . .	62
Figure 4.6 : Tensions de commande. . . . .	62
Figure 4.7 : Comparaison des performances des PSS (scénario n°1). . . . .	64
Figure 4.8 : Comparaison des performances des PSS (scénario n°2). . . . .	64
Figure 4.9 : Comparaison des performances des PSS (scénario n°3). . . . .	64



Figure 4.10 : Performances du PSS proposé (scénario n°1).	66
Figure 4.11 : Performances du PSS proposé (scénario n°2).	66
Figure 4.12 : Performances du PSS proposé (scénario n°3).	66
Figure 4.13 : Réponses du système (scénario n°1).	67
Figure 4.14 : Réponses du système (scénario n°2).	67
Figure 4.15 : Réponses du système (scénario n°3).	67
Figure 4.16 : Réseau à 3 machines.	68
Figure 4.17 : Variation de l'hypervolume.	70
Figure 4.18 : Front non dominé.	70
Figure 4.19 : Réponses du système avec la 1 <sup>ère</sup> solution.	71
Figure 4.20 : Réponses du système avec la 2 <sup>ème</sup> solution.	71
Figure 4.21 : Réponses du système avec la 3 <sup>ème</sup> solution.	71
Figure 4.22 : Comparaison des performances des PSS (scénario n°1).	72
Figure 4.23 : Comparaison des performances des PSS (scénario n°2).	72
Figure 4.24 : Comparaison des performances des PSS (scénario n°3).	72
Figure 4.25 : Comparaison des performances des PSS (scénario n°1).	74
Figure 4.26 : Comparaison des performances des PSS (scénario n°1).	74
Figure 4.27 : Comparaison des performances des PSS (scénario n°2).	74
Figure 4.28 : Comparaison des performances des PSS (scénario n°2).	75
Figure 4.29 : Comparaison des performances des PSS (scénario n°3).	75
Figure 4.30 : Comparaison des performances des PSS (scénario n°3).	75
Figure 4.31 : Performances du PSS proposé (scénario n°1).	76
Figure 4.32 : Performances du PSS proposé (scénario n°2).	76
Figure 4.33 : Performances du PSS proposé (scénario n°3).	76
Figure 4.34 : Réponses de la 2 <sup>ème</sup> machine (scénario n°1).	77
Figure 4.35 : Réponses de la 3 <sup>ème</sup> machine (scénario n°1).	77
Figure 4.36 : Réponses de la 2 <sup>ème</sup> machine (scénario n°2).	77
Figure 4.37 : Réponses de la 3 <sup>ème</sup> machine (scénario n°2).	78
Figure 4.38 : Réponses de la 2 <sup>ème</sup> machine (scénario n°3).	78
Figure 4.39 : Réponses de la 3 <sup>ème</sup> machine (scénario n°3).	78
Figure 4.40 : Réseau à 10 machines.	80
Figure 4.41 : Variation de l'hypervolume.	81
Figure 4.42 : Front non dominé.	82
Figure 4.43 : Réponses du système avec la 1 <sup>ère</sup> solution.	83
Figure 4.44 : Réponses du système avec la 2 <sup>ème</sup> solution.	83
Figure 4.45 : Tensions d'excitation avec la 1 <sup>ère</sup> solution.	83
Figure 4.46 : Tensions d'excitation avec la 2 <sup>ème</sup> solution.	84
Figure 4.47 : Déviations de vitesse de la 2 <sup>ème</sup> machine.	84
Figure 4.48 : Tensions d'excitation de la 2 <sup>ème</sup> machine.	84
Figure 4.49 : Comparaison des performances des PSS (scénario n°1).	85
Figure 4.50 : Comparaison des performances des PSS (scénario n°2).	85
Figure 4.51 : Comparaison des performances des PSS (scénario n°3).	85

Figure 4.52 : Performances du PSS proposé (scénario n°1). . . . .	87
Figure 4.53 : Performances du PSS proposé (scénario n°2). . . . .	87
Figure 4.54 : Performances du PSS proposé (scénario n°3). . . . .	87
Figure 4.55 : Réponses du système avec le PSS proposé (scénario n°1). . . . .	88
Figure 4.56 : Réponses du système avec le PSS conventionnel (scénario n°1). . . . .	88
Figure 4.57 : Réponses du système avec le PSS proposé (scénario n°2). . . . .	88
Figure 4.58 : Réponses du système avec le PSS conventionnel (scénario n°2). . . . .	89
Figure 4.59 : Réponses du système avec le PSS proposé (scénario n°3). . . . .	89
Figure 4.60 : Réponses du système avec le PSS conventionnel (scénario n°3). . . . .	89

# Liste des tableaux

Tableau 3.1 : Résultats de la métrique $GD$ . . . . .	48
Tableau 3.2 : Résultats de la métrique $H$ . . . . .	48
Tableau 3.3 : Résultats de la métrique $S$ . . . . .	49
Tableau 3.4 : Résultats de la métrique $\bar{d}_{\max}$ . . . . .	49
Tableau 4.1 : Espace de recherche pour le réseau mono-machine. . . . .	59
Tableau 4.2 : Résultats des métriques pour le réseau mono-machine. . . . .	60
Tableau 4.3 : Paramètres des PSS sélectionnés pour le réseau mono-machine. . . . .	63
Tableau 4.4 : Espace de recherche pour le à 3 machines. . . . .	69
Tableau 4.5 : Résultats des métriques pour le réseau à 3 machines. . . . .	69
Tableau 4.6 : Paramètres des PSS sélectionnés pour le réseau à 3 machines. . . . .	79
Tableau 4.7 : Espace de recherche pour le réseau à 10 machines. . . . .	80
Tableau 4.8 : Résultats des métriques pour le réseau à 10 machines. . . . .	81
Tableau 4.9 : Paramètres des PSS sélectionnés pour le réseau à 10 machines. . . . .	86
Tableau 5.1 : Classements des algorithmes pour les problèmes de test. . . . .	92
Tableau 5.2 : Classements des algorithmes pour les problèmes de dimensionnement des PSS. . . . .	93
Tableau B.1 : Paramètres du réseau [79]. . . . .	104
Tableau B.2 : Points de fonctionnement. . . . .	104
Tableau C.1 : Paramètres des machines [86]. . . . .	105
Tableau C.2 : Paramètres des lignes [86]. . . . .	105
Tableau C.3 : Points de fonctionnement [86]. . . . .	106
Tableau C.4 : Charges du réseau [86]. . . . .	106
Tableau D.1 : Paramètres des machines [86]. . . . .	107
Tableau D.2 : Paramètres des lignes [86]. . . . .	108
Tableau D.3 : Points de fonctionnement [86]. . . . .	109
Tableau D.4 : Charges du réseau [86]. . . . .	109

# Chapitre 1

## Introduction

# Chapitre 1

## Introduction

### 1.1 L'optimisation multiobjectif

La plupart des problèmes réels peuvent être vus comme des problèmes d'optimisation multiobjectif, c'est-à-dire des problèmes qui consistent à minimiser ou maximiser plusieurs critères de performance indépendamment les uns des autres. Ces critères sont définis selon les objectifs du problème et sont souvent contradictoires, ce qui signifie qu'ils ne peuvent être optimisés simultanément. Dans ce cas, les problèmes traités sont dits multiobjectifs, ou multicritères, et leur résolution nécessitent l'emploi de méthodes ou algorithmes dits d'optimisation multiobjectif.

En automatique, les critères de performance considérés dans les problèmes de commande concernent généralement la stabilité des systèmes (marges de stabilité et de robustesse, amortissement, etc.), les performances temporelles des réponses (temps de montée, temps de réponse, erreur statique, etc.) et certains aspects des signaux de commande (énergie, variance, etc.). Toutefois, les critères les plus courants sont le dépassement et le temps de réponse du système, ces derniers sont souvent traités d'une manière agrégée dans le but d'aboutir à une solution de compromis [1].

En général, la solution d'un problème d'optimisation multiobjectif n'est pas unique. Dans ce cas, il existe un ensemble de solutions optimales qui représentent un compromis entre les valeurs prises par les fonctions objectifs du problème. La résolution d'un tel problème consiste donc à rechercher toutes les solutions qui offrent le meilleur compromis possible, c'est-à-dire les solutions dont les performances sont les meilleures vis-à-vis de toutes les autres solutions potentielles du problème.

#### 1.1.1 Formulation mathématique

L'optimisation multiobjectif, dite aussi multicritère ou vectorielle, est un processus de recherche qui permet d'optimiser simultanément  $m$  composantes  $f_i$ ,  $i = 1, \dots, m$ , d'une certaine fonction vectorielle  $\mathbf{f}$ , dépendant d'une variable de décision générale  $\mathbf{x}$  dans un univers de recherche  $\mathcal{X}$ , et éventuellement en présence de  $k$  contraintes ou conditions impliquant  $\mathbf{x}$ . En supposant un problème de minimisation, ce dernier peut être traduit mathématiquement de la façon suivante [2] :

$$\begin{aligned}
\text{Minimiser } \mathbf{f}(\mathbf{x}) &= (f_1(\mathbf{x}), \dots, f_m(\mathbf{x})) \\
\text{tel que } \mathbf{g}(\mathbf{x}) &= (g_1(\mathbf{x}), \dots, g_k(\mathbf{x})) \leq \mathbf{0} \\
\text{où } \mathbf{x} &= (x_1, \dots, x_n) \in \mathcal{X}
\end{aligned} \tag{1.1}$$

En pratique, un tel problème admet plusieurs solutions optimales. L'ensemble de ces solutions, dites aussi non dominées, est appelé l'ensemble optimal ou Pareto-optimal, et les vecteurs objectifs correspondants forment ce qu'on appelle le front optimal ou le front de Pareto. Ainsi, nous présentons dans ce qui suit les notions de dominance et d'optimalité au sens de Pareto [2].

**Définition 1.1. (Dominance au sens de Pareto)** On dit qu'un vecteur  $\mathbf{u} = (u_1, \dots, u_m)$  domine  $\mathbf{v} = (v_1, \dots, v_m)$  si et seulement si  $\mathbf{u}$  est partiellement inférieur à  $\mathbf{v}$  ( $\mathbf{u} \prec \mathbf{v}$ ), c'est-à-dire :

$$\forall i \in \{1, \dots, m\}, u_i \leq v_i \quad \exists i \in \{1, \dots, m\}, u_i < v_i \tag{1.2}$$

Si aucun des deux vecteurs  $\mathbf{u}$  et  $\mathbf{v}$  ne domine l'autre, c'est-à-dire si  $\mathbf{u}$  ne domine pas  $\mathbf{v}$  et  $\mathbf{v}$  ne domine pas  $\mathbf{u}$ , alors  $\mathbf{u}$  et  $\mathbf{v}$  sont dits incomparables ou non dominés entre eux.

**Définition 1.2. (Optimalité au sens de Pareto)** Une solution  $\mathbf{x}^* \in \mathcal{X}$  est dite Pareto-optimale si, et seulement si, il n'existe aucune autre solution  $\mathbf{x} \in \mathcal{X}$  pour laquelle  $\mathbf{v} = \mathbf{f}(\mathbf{x}) = (v_1, \dots, v_m)$  domine  $\mathbf{u} = \mathbf{f}(\mathbf{x}^*) = (u_1, \dots, u_m)$ .

**Définition 1.3. (Ensemble Pareto-optimal)** L'ensemble Pareto-optimal d'un problème multiobjectif quelconque est défini par :

$$\mathcal{EP} = \{\mathbf{x}^* \in \mathcal{X} \mid \nexists \mathbf{x} \in \mathcal{X}, \mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{x}^*)\} \tag{1.3}$$

**Définition 1.4. (Front de Pareto)** On appelle front de Pareto, ou surface de compromis, l'ensemble des vecteurs objectifs correspondant aux solutions Pareto-optimales :

$$\mathcal{FP} = \{\mathbf{u} = \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x})) \mid \mathbf{x} \in \mathcal{EP}\} \tag{1.4}$$

Si on considère un sous-ensemble  $\mathcal{A}$  de  $\mathcal{X}$ , les solutions non dominées de  $\mathcal{A}$  constituent l'ensemble non dominé de ce dernier, et les vecteurs objectifs correspondants représentent le front non dominé de  $\mathcal{A}$ .

La figure 1.1 illustre les concepts précédents pour un problème de minimisation à deux fonctions objectifs. Dans ce cas, seules les trois solutions représentées par les points  $a$ ,  $c$  et  $e$  sont non dominées parmi l'ensemble des points  $\{a, b, c, d, e, f\}$  dans l'espace des objectifs. Les points  $a$ ,  $c$  et  $e$  forment ainsi le front non dominé de cet ensemble.

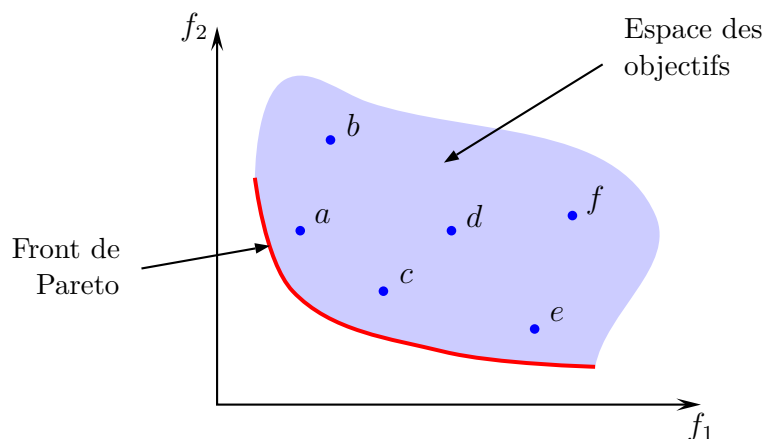


Figure 1.1 : Illustration des notions de dominance et d'optimalité de Pareto.

Par ailleurs, la surface de Pareto de ce problème est représentée par la courbe rouge à l'extrémité de l'espace des objectifs.

La comparaison de deux solutions, dont l'une au moins n'est pas réalisable, nécessite la prise en considération du degré de violation des contraintes. La règle de comparaison la plus simple est énoncée comme suit [3] :

**Définition 1.5. (Dominance avec contraintes)** Une solution  $\mathbf{x}$  domine  $\mathbf{y}$  si l'une des conditions suivantes est satisfaite :

- La solution  $\mathbf{x}$  est réalisable et la solution  $\mathbf{y}$  ne l'est pas.
- Les deux solutions ne sont pas réalisables mais  $\mathbf{x}$  possède un degré de violation des contraintes inférieur à celui de  $\mathbf{y}$ .
- Les deux solutions sont réalisables et  $\mathbf{x}$  domine  $\mathbf{y}$ .

### 1.1.2 Classification des méthodes de résolution

Les méthodes de résolution multiobjectif peuvent être classifiées du point de vue d'un concepteur ou celui d'un décideur. La première classification est établie selon la prise en considération ou non de la notion de dominance de Pareto durant le processus de recherche. Ainsi, nous distinguons les trois grandes familles suivantes [4] :

**Les approches agrégées :** dans lesquelles le problème multiobjectif est transformé en un problème mono-objectif. L'ensemble Pareto-optimal est obtenu en modifiant les paramètres d'influence des différentes fonctions objectifs du problème afin de guider la recherche vers différentes solutions optimales.

**Les approches Pareto :** se basent sur les notions de dominance et d'optimalité au sens de Pareto afin de comparer et évaluer la qualité des solutions potentielles du problème les unes par rapport aux autres.

**Les approches non agrégées et non Pareto :** où les fonctions objectifs sont traitées séparément. L'ensemble de Pareto est établi en considérant toutes les solutions optimales obtenues pour chacune des fonctions objectifs du problème.

D'un point de vue décideur, les approches de résolution sont classifiées selon la manière avec laquelle le décideur intervient sur le processus de recherche [4] :

**Les approches a priori :** le décideur définit ses préférences avant l'exécution du processus d'optimisation, généralement en transformant le problème multiobjectif en un problème mono-objectif. Ces approches permettent de simplifier la recherche en limitant le nombre de solutions optimales à rechercher, mais requiert toutefois des connaissances a priori sur les fonctions objectifs du problème. De plus, il est nécessaire de relancer l'optimisation à chaque fois que les préférences du décideur changent.

**Les approches interactives :** le décideur affine son choix de compromis pendant l'exécution de l'algorithme de l'optimisation, ce qui lui permet donc de guider la recherche vers les solutions souhaitées tout en diminuant la complexité et le temps d'exécution de l'algorithme. Toutefois, le décideur est contraint de suivre d'une façon continue les résultats de l'optimisation afin de pouvoir intervenir et imposer ses préférences sur le processus de recherche.

**Les approches a posteriori :** le décideur choisit la solution de son choix parmi l'ensemble des solutions obtenues à la fin de la recherche. Ces approches sont les plus courantes et permettent de trouver toutes les solutions optimales d'un problème quelconque en exécutant une seule fois le processus d'optimisation. Toutefois, la complexité et le temps d'exécution pour ces approches sont les plus élevés.

## 1.2 Les métaheuristiques

Une métaheuristique est un algorithme itératif de recherche qui permet de résoudre différents types de problèmes d'optimisation difficile sans modifications majeures. Les métaheuristiques sont utilisées pour résoudre tous les problèmes dont on ne connaît pas d'heuristique efficace pouvant fournir une solution optimale en un temps raisonnable [5]. Ainsi, les principales caractéristiques des métaheuristiques sont les suivantes :

- Le but des métaheuristiques est de guider le processus de recherche vers les solutions optimales, ou sous-optimales, en exploitant les solutions potentielles du problème et en explorant l'espace de recherche d'une manière efficace.
- Les métaheuristiques sont généralement des méthodes approchées, c'est-à-dire non déterministes, de ce fait elles ne donnent aucune garantie de trouver la solution optimale d'un problème quelconque.
- Elles sont généralement conçues sous forme d'algorithmes stochastiques ou probabilistes dans lesquels il y a un usage de l'aléatoire.



- Elles sont souvent inspirées de la nature.
- Elles partagent les mêmes inconvénients, à savoir la difficulté de réglage des paramètres utilisés, et le temps de calcul qui dépend notamment du problème traité.

### 1.2.1 Classification des métaheuristiques

Les métaheuristiques sont classifiées généralement selon la manière avec laquelle les solutions potentielles du problème sont recherchées et exploitées durant le déroulement de l'optimisation [5]. Ainsi, on distingue les méthodes manipulant un ensemble de solutions potentielles, appelé population, et celles qui ne manipulent qu'une seule solution à la fois. La figure 1.2 montre la classification des métaheuristiques parmi les autres méthodes de résolution multiobjectif.

**Les métaheuristiques à trajectoire :** dites aussi méthodes de recherche locale, elles permettent d'améliorer d'une façon itérative, une seule solution à la fois. La méthode de recherche tabou, le recuit simulé et la recherche à voisinages variables sont des exemples connus de ces méthodes. Ce type de métaheuristique explore l'espace de recherche en construisant des trajectoires qui permettent de diriger la recherche d'une façon efficace vers les solutions optimales.

**Les méthodes à population :** dites aussi méthodes globales, conduisent la recherche vers les solutions optimales en exploitant les informations fournies par plusieurs points dans l'espace de recherche. Les méthodes à population les plus connues sont les algorithmes évolutionnaires, l'optimisation par essaims de particules, l'optimisation par colonies de fourmis, et les algorithmes à estimation de distribution.

### 1.2.2 Choix d'une métaheuristique

L'objectif principal des métaheuristiques est de fournir des solutions optimales de qualité acceptable, en un temps raisonnable, et sans modifications majeures. Toutefois, le choix d'une méthode efficace parmi d'autres, pour un problème donné, n'est pas toujours évident. En effet, ce choix dépend surtout du type et des caractéristiques du problème traité. Dans ce qui suit, nous citons les principaux types de problèmes d'optimisation multiobjectif qu'on peut rencontrer en pratique.

**Les problèmes de dimensions élevées :** ce sont des problèmes qui impliquent un nombre élevé de variables de décision et/ou de fonctions objectifs.

**Les problèmes non linéaires :** dans lesquels les relations reliant les fonctions objectifs et les variables de décision sont non linéaires.

**Les problèmes multimodaux :** dont l'espace des objectifs présente plusieurs fronts locaux qui correspondent à des solutions sous optimales. Toutefois, il est parfois nécessaire

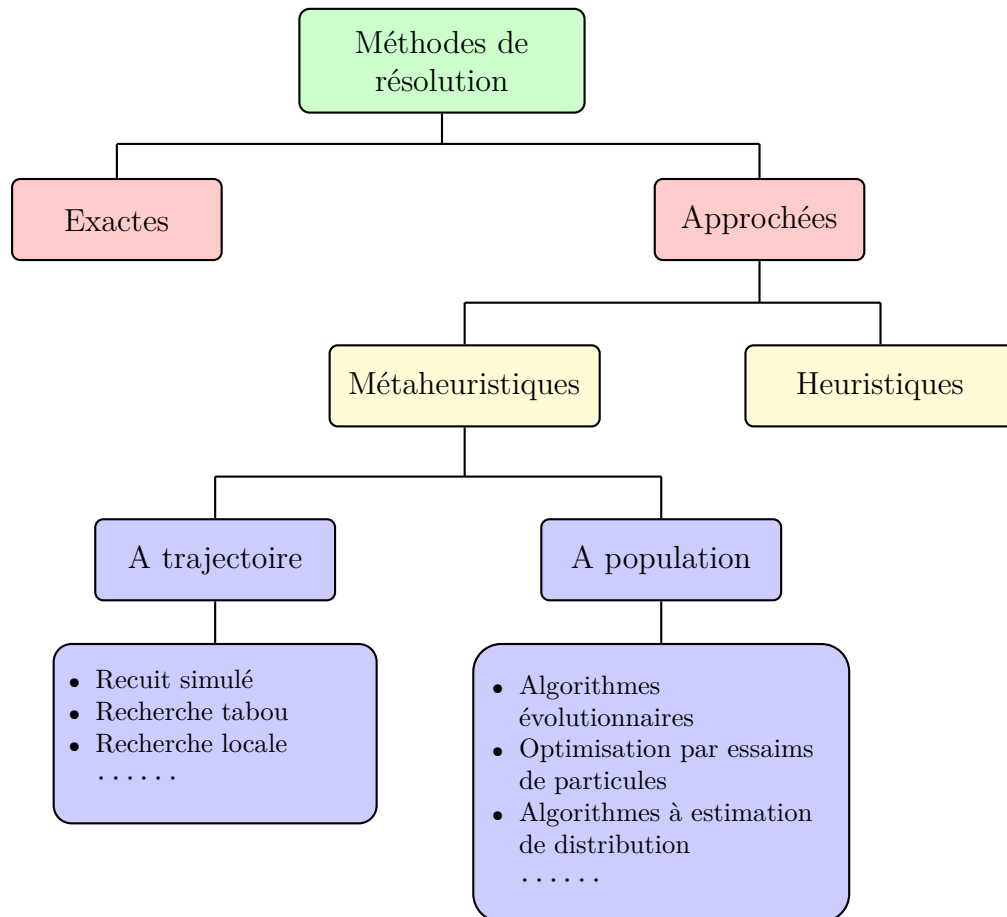


Figure 1.2 : Classification des métaheuristiques.

de rechercher les solutions sous optimales d'un problème donnée, on parle alors d'une optimisation multimodale.

**Les problèmes avec des fronts complexes :** dans ce cas, les fronts de Pareto présentent plusieurs caractéristiques à la fois (concavité, convexité, discontinuité, etc.).

**Les problèmes dynamiques :** dans lesquels les fonctions objectifs et/ou l'espace de recherche peuvent changer au cours de la recherche.

**Les problèmes incertains :** pour lesquels le calcul des fonctions objectifs est imprécis à cause de l'utilisation de paramètres incertains ou de la présence d'un bruit.

### 1.3 Synthèse optimale des stabilisateurs de puissance dans les réseaux électriques

Les réseaux électriques, ou systèmes de puissance, sont des systèmes complexes, non linéaires et fortement interconnectés. Ils fonctionnent souvent dans des conditions de stabilité extrêmes, et sont confrontés à plusieurs types de perturbations telles que les

courts-circuits et les variations paramétriques. Par conséquent, la stabilité de ces systèmes est une condition primordiale et un critère important pour garantir une qualité optimale de la puissance électrique fournie aux consommateurs [6].

### 1.3.1 Les stabilisateurs de puissance

La stabilité d'un réseau électrique peut être améliorée au moyen des systèmes de régulation appelés stabilisateurs de puissance ou PSS (Power System Stabilizers). Ces derniers sont installés au niveau des systèmes d'excitation des générateurs du réseau dans le but d'apporter un signal de commande supplémentaire à l'entrée du régulateur de tension (AVR). Ce signal est proportionnel à la variation de vitesse de rotor du générateur et permet d'amortir les oscillations électromécaniques causées par les différentes perturbations qui peuvent affecter le réseau.

Le type de PSS le plus employé possède la structure d'un régulateur à avance/retard de phase. Ce PSS, dont la fonction de transfert est donnée par l'équation 1.5, utilise la variation de la vitesse rotorique du générateur comme signal d'entrée. Il est constitué d'un bloc amplificateur, d'un filtre passe-haut et de deux blocs de compensation avance-retard de phase. L'amplificateur détermine le degré d'amortissement introduit par le PSS. Le filtre élimine les oscillations à très basse fréquence (inférieure à 0.2 Hz) du signal d'entrée et l'effet de la composante continue sur la tension terminale du générateur. Enfin, les deux blocs de compensation de phase permettent de compenser le retard de phase qui existe entre le couple électrique du générateur et l'entrée du système d'excitation [7].

$$V_{PSS} = K \left( \frac{sT_W}{1 + sT_W} \right) \left( \frac{1 + sT_1}{1 + sT_2} \right) \left( \frac{1 + sT_3}{1 + sT_4} \right) \cdot \Delta\omega \quad (1.5)$$

où  $V_{PSS}$  est la tension de sortie du PSS,  $\Delta\omega$  est la déviation de vitesse,  $K$  le gain de l'amplificateur,  $T_W$  la constante de temps du filtre passe-haut, et  $T_i$ ,  $i = 1, \dots, 4$  sont les constantes de temps des deux blocs de compensation de phase.

### 1.3.2 Dimensionnement optimal des PSS

Les performances des PSS dépendent fortement du choix de leurs paramètres. Ces derniers sont généralement réglés de façon à maximiser l'amortissement des oscillations électromécaniques causées par les perturbations. En général, les constantes de temps  $T_W$ ,  $T_2$  et  $T_4$  sont fixées a priori par le concepteur, ce qui veut dire que seuls les paramètres  $K$ ,  $T_1$  et  $T_3$  sont réglables.

Le dimensionnement optimal des PSS est généralement formulé comme un problème d'optimisation d'une ou de plusieurs fonctions objectifs. Ce problème peut être traité d'une façon efficace en utilisant des approches basées sur les métaheuristiques [8]. Dans la plupart des travaux de recherche proposés, les problèmes abordés consistent à optimiser

une seule fonction objectif. On distingue alors deux types de problème ; les problèmes dont les fonctions objectifs dépendent essentiellement de la variable du temps et les problèmes qui impliquent principalement les valeurs propres du système.

Dans le premier cas, plusieurs travaux ont été réalisés en utilisant des approches basées typiquement sur les algorithmes génétiques [9, 10], l'optimisation par essais de particules [11, 12], l'évolution différentielle [13] ainsi que d'autres types de métaheuristiques [14, 15, 16, 17]. Les fonctions objectifs proposées concernent principalement les réponses temporelles des vitesses rotoriques et des tensions terminales des générateurs. Ces approches permettent de caractériser d'une façon précise les objectifs du problème, mais nécessitent toutefois des simulations temporelles, ce qui rend le temps de calcul plus élevé. De plus, les résultats obtenus dépendent sensiblement des scénarios considérés dans les simulations.

Dans le cas où les fonctions objectifs sont définies au moyen des valeurs propres du système, le modèle dynamique du réseau électrique est linéarisé autour d'un ou de plusieurs points de fonctionnement. Les valeurs propres permettent de caractériser les performances du système indépendamment du temps et des scénarios considérés pour les simulations. Les fonctions objectifs proposées concernent principalement les coefficients d'amortissement et les parties réelles des valeurs propres. On distingue alors des approches utilisant les algorithmes génétiques [18, 19], l'optimisation par essais de particules [20, 21], l'évolution différentielle [22, 23] et le recuit simulé [26, 24]. D'autres métaheuristiques sont également utilisées telles que la recherche tabou [25], l'optimisation par colonies de fourmis [27], les systèmes immunitaires artificiels [28], ainsi que d'autres types de méthodes inspirées généralement de la nature [30, 29, 31, 32, 33, 34, 35, 36, 37]. Par ailleurs, il existe des travaux où les fonctions objectifs combinent des critères temporels avec ceux manipulant des valeurs propres [38, 39, 40]. Des études comparatives entre plusieurs métaheuristiques ont également été proposées [41, 42].

D'autre part, il existe des approches qui combinent plusieurs types de métaheuristiques dans le but d'améliorer les performances des méthodes standard et ainsi la qualité des solutions recherchées. Des exemples de ces approches sont la combinaison des algorithmes génétiques avec la méthode recherche locale [43], l'évolution différentielle avec l'optimisation par colonies de fourmis [44] et l'optimisation par essais de particules avec l'algorithme chaotique [45].

Le problème de dimensionnement multiobjectif des PSS a été également abordé dans la littérature. Dans ce cas, les fonctions objectifs du problème sont définies en utilisant principalement les valeurs propres du système [46, 47, 48, 49, 50, 51], des critères de performance temporelle [52, 53, 54, 55, 56], ou encore des critères combinés [57, 58, 59, 60]. Toutefois, la plupart de ces problèmes sont traités au moyen des algorithmes génétiques en considérant seulement deux fonctions objectifs. Dans la plupart de ces travaux, les deux fonctions objectifs sont formulées en utilisant les coefficients d'amortissement et les parties réelles des valeurs propres du système.

## 1.4 Objectifs du travail

Ce travail consiste à proposer des nouvelles solutions aux problèmes de synthèse des PSS dans les réseaux électriques au moyen des stabilisateurs de puissance, et en utilisant des méthodes de résolution multiobjectif basées sur des métaheuristiques. Ainsi, les principaux objectifs de ce travail sont les suivants :

- Proposition d'une nouvelle méthode de résolution multiobjectif pour des problèmes académiques et pour le dimensionnement optimal des PSS.
- Proposition de nouvelles approches de dimensionnement des PSS utilisant les normes  $H_\infty$  et  $H_2$ , ainsi que les valeurs propres du système en boucle fermée.
- Etude et application de quelques méthodes d'optimisation multiobjectif de référence sur des problèmes de dimensionnement optimal des PSS pour un réseau électrique mono-machine et deux réseaux multimachines.

## 1.5 Organisation du document

Ce document est organisé de la façon suivante ; dans le chapitre 2, nous présentons quatre algorithmes de référence dans le domaine de l'optimisation multiobjectif. Ces algorithmes sont basés sur les algorithmes génétiques, l'évolution différentielle, le recuit simulé et l'optimisation par essaims de particules.

Dans le chapitre 3, nous développons un nouvel algorithme d'optimisation multiobjectif en se basant sur la méthode de l'évolution différentielle. Nous décrivons, dans un premier temps, le fonctionnement de cet algorithme en détail. Ensuite, dans le but d'évaluer ses performances, nous réalisons une étude comparative avec les autres algorithmes de référence, en considérant la résolution de plusieurs problèmes académiques. Nous utilisons pour cela plusieurs métriques de performance dans le but de mesurer les performances des algorithmes utilisés.

Dans le chapitre 4, plusieurs approches de dimensionnement des PSS sont proposées en utilisant les différentes méthodes de résolution étudiées à cet effet. Ces approches concernent trois exemples de réseau électrique ; un réseau mono-machine, et deux autres réseaux à 3 et à 10 machines.

Le dernier chapitre conclut le travail, en résumant d'abord les différents résultats clés ainsi que les contributions de cette thèse, et en proposant ensuite quelques perspectives de recherche.

## Chapitre 2

# Optimisation multiobjectif par métaheuristiques

## Chapitre 2

# Optimisation multiobjectif par métaheuristiques

## 2.1 Introduction

Un nombre considérable de méthodes de résolution multiobjectif ont été proposées dans la littérature en utilisant les métaheuristiques. On distingue alors des méthodes développées pour résoudre des problèmes académiques, et d'autres qui sont conçues pour des problèmes spécifiques. La plupart de ces méthodes adoptent la notion de dominance de Pareto, et également le principe d'élitisme qui consiste à assurer la préservation des meilleures solutions durant les étapes de recherche [61]. Dans ce chapitre, nous présentons quatre algorithmes multiobjectifs de référence, dont les performances ont été prouvées tant sur des problèmes académiques que réels. Ces algorithmes sont connus sous les noms de NSGA-II, GDE3, AMOSA et SMPSO.

## 2.2 Non-dominated Sorting Genetic Algorithm II (NSGA-II)

L'algorithme NSGA-II a été développé par Deb et al. [62] et se base sur les algorithmes génétiques. Cet algorithme a été proposé comme une amélioration de l'algorithme NSGA, et se distingue de ce dernier par l'emploi de la procédure de ranking. De plus, NSGA-II utilise une nouvelle technique d'estimation de densité qui lui permet de comparer des solutions non dominées d'une manière plus efficace. L'algorithme NSGA-II est considéré actuellement comme une méthode de référence dans le domaine de l'optimisation multiobjectif, et il est constamment utilisé dans les études de comparaison avec d'autres algorithmes multiobjectifs.

### 2.2.1 Les algorithmes génétiques

Les algorithmes génétiques (AG) font partie de la famille des méthodes de recherche dites évolutionnaires qui sont inspirées principalement de la biologie. Le principe de fonctionnement de ces méthodes consiste à faire évoluer une population

---

**Algorithme 1** : Pseudo-code d'un algorithme génétique

---

**Entrées** :  $N, P_c, P_m$   
 $t \leftarrow 0$   
Initialiser  $P_t$   
Initialiser  $\mathbf{x}_{opt}$   
**Tant que** Critère d'arrêt non satisfait **faire**  
    Sélectionner  $Q_t$  à partir de  $P_t$   
    Appliquer le croisement et la mutation sur  $Q_t$   
    Etablir  $P_{t+1}$  à partir de  $P_t$  et  $Q_t$   
     $t \leftarrow t + 1$   
    Mettre à jour  $\mathbf{x}_{opt}$   
**Fin**  
**Sorties** :  $\mathbf{x}_{opt}$

---

d'individus, représentant des solutions potentielles au problème traité, à travers un certain nombre d'itérations, dites générations, dans le but d'obtenir des solutions de plus en plus efficaces. La qualité des solutions est améliorée en appliquant les opérateurs génétiques, à chaque génération, sur la population d'individus [63].

Un algorithme génétique standard est décrit par l'algorithme 1. Le fonctionnement de ce dernier commence par l'initialisation de la population courante, constituée de  $N$  individus, et de la solution optimale temporaire  $\mathbf{x}_{opt}$ . Chaque individu est vu comme étant une solution potentielle, et possède donc une mesure de qualité appelée fitness. La valeur de fitness est calculée en appliquant la fonction objectif du problème et en tenant compte de tous les autres individus de la population. A chaque génération  $t$ , une nouvelle population d'individus  $Q_t$ , appelée population descendante, est générée à partir de la population courante  $P_t$  en appliquant les opérateurs génétiques. Les deux populations  $Q_t$  et  $P_t$  sont alors combinées pour former une nouvelle population parente de taille  $N$ . Ce processus s'itère jusqu'à ce que le critère d'arrêt soit satisfait, généralement lorsqu'un certain nombre de générations est atteint.

Les opérateurs de variation utilisés dans les AG sont la sélection, le croisement et la mutation. La sélection est un processus qui permet de préserver les caractéristiques des meilleurs individus au fil des générations. La méthode de sélection la plus employée est la méthode dite par tournoi, qui consiste à sélectionner aléatoirement deux solutions et choisir celle qui possède la meilleure valeur de fitness [63].

Les deux opérateurs de croisement et de mutation permettent de générer de nouvelles solutions. Le croisement combine les caractéristiques de deux individus afin de donner lieu à deux nouveaux individus. Il est généralement appliqué avec une probabilité,  $P_c$ , proche de 1. Par ailleurs, la mutation est un opérateur de diversification, qui permet d'explorer l'espace de recherche en apportant de petites modifications aléatoires aux solutions existantes. La mutation est appliquée généralement avec un taux,  $P_m$ , très faible.



**Algorithme 2** : Pseudo-code de NSGA-II

---

```

Entrées :  $N, P_c, P_m$ 
 $t \leftarrow 0$ 
Initialiser  $P_t$ 
Générer  $Q_t$ 
Tant que Critère d'arrêt non satisfait faire
     $R_t \leftarrow P_t \cup Q_t$ 
    Appliquer le ranking sur  $R_t$ 
     $P_{t+1} \leftarrow \emptyset$ 
     $i \leftarrow 1$ 
    Tant que  $|P_{t+1}| < N$  faire
        Inclure dans  $P_{t+1}$  le  $i^{\text{ème}}$  front de  $R_t$ 
         $i \leftarrow i + 1$ 
    Fin
    Si  $|P_{t+1}| > N$  alors
        Retirer de  $P_{t+1}$  les solutions du dernier front qui possèdent les plus petites
        distances de crowding
    Fin
    Générer  $Q_{t+1}$  à partir de  $P_{t+1}$ 
     $t \leftarrow t + 1$ 
Fin
Sorties :  $P_t$ 

```

---

La dernière étape dans le fonctionnement d'un algorithme génétique consiste à remplacer la population courante par une nouvelle population en prenant en considération les solutions de la population descendante. Les types de remplacement les plus utilisés sont dits élitistes, dans lesquels on garantit le passage des meilleures solutions notamment  $\mathbf{x}_{opt}$ .

Les algorithmes génétiques sont des méthodes robustes, applicables à une large gamme de problèmes d'optimisation difficile, notamment aux problèmes multiobjectifs. Toutefois, la complexité et le temps de calcul de ces méthodes sont plus élevés par rapport aux autres métaheuristiques.

### 2.2.2 Description de l'algorithme NSGA-II

Le pseudo-code de NSGA-II est donné par l'algorithme 2. Dans un premier temps, une population initiale de  $N$  individus,  $P_0$ , est initialisée aléatoirement et une population descendante,  $Q_0$ , est générée à partir de  $P_0$  en appliquant les opérateurs de variation génétique (sélection, croisement et mutation). Ensuite, l'algorithme est exécuté pour un certain nombre de générations, où à chaque génération  $t$ , une nouvelle population  $P_{t+1}$  est formée à partir de  $P_t$  et  $Q_t$  en sélectionnant les meilleurs individus d'après leurs rangs et éventuellement une certaine mesure de densité, appelée distance de crowding. Enfin, une nouvelle population descendante,  $Q_{t+1}$ , est générée en appliquant les opérateurs de variation à  $P_{t+1}$ .

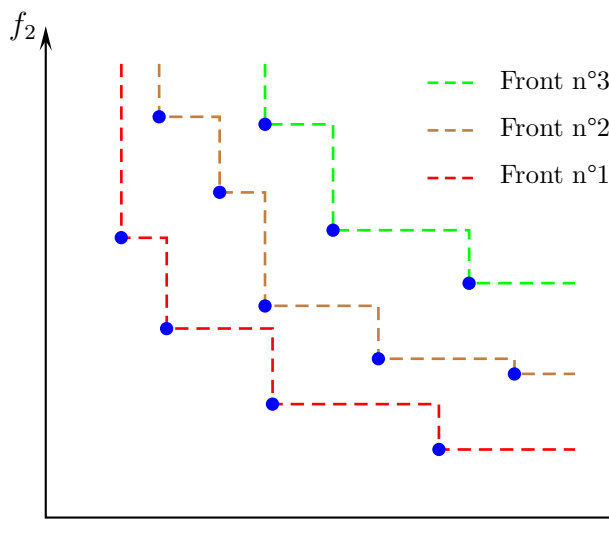


Figure 2.1 : Illustration de la procédure de ranking.

Le ranking permet de classer les individus d'une population selon la notion de front. Pour ce faire, tous les individus non dominés de la population reçoivent d'abord le rang 1 et sont retirés temporairement de la population. Ensuite, les individus non dominés de ce qui reste de la population reçoivent le rang 2 avant d'être retirés à leur tour. Cette procédure est répétée jusqu'à ce que tous les individus se voient attribués un rang. La figure 2.1 illustre la procédure de ranking appliquée à une population de 12 individus répartis en 3 fronts.

La distance de crowding est une mesure importante dans le fonctionnement de NSGA-II. En effet, elle est premièrement impliquée dans la mise à jour de la population parente, et deuxièmement pour le maintien de la diversité dans cette population afin de favoriser les individus les plus isolés sur leurs fronts. La distance de crowding d'un point quelconque se calcule en fonction du périmètre de l'hypercube ayant comme sommets les points les plus proches de ce point et appartenant au même front que celui-ci. Un exemple du calcul de cette distance est donné dans la figure 2.2 pour deux points situés dans des fronts différents.

Les opérateurs de variation utilisés par l'algorithme NSGA-II sont la sélection par tournoi, le croisement binaire simulé et la mutation polynomiale. La sélection par tournoi compare les rangs de deux individus sélectionnés aléatoirement et choisit l'individu avec le meilleur rang. Dans le cas où ces deux individus possèdent le même rang, l'individu ayant la plus grande distance de crowding est sélectionné.

Le croisement binaire simulé est un opérateur qui permet de combiner deux solutions à paramètres réels, sélectionnées aléatoirement, afin de générer deux autres solutions. Cet opérateur permet d'exploiter les solutions courantes afin de diriger la recherche. La mutation polynomiale permet de diversifier la recherche pour éviter les convergences prématurées vers les fronts locaux.

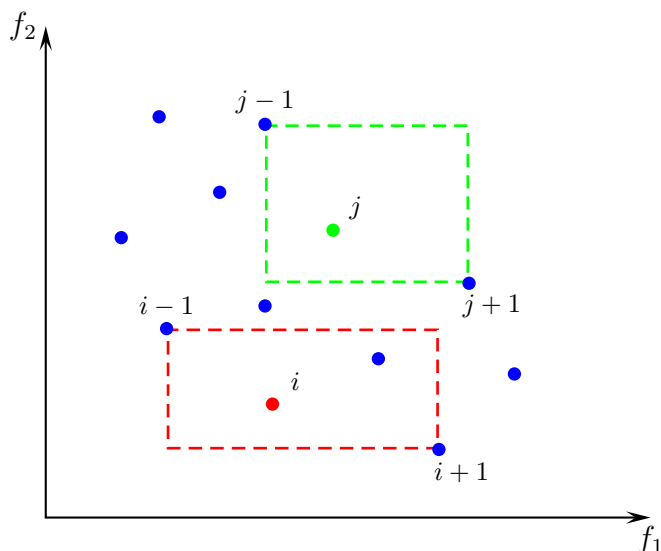


Figure 2.2 : Calcul de la distance de crowding.

## 2.3 Generalized Differential Evolution 3 (GDE3)

L'algorithme GDE3 a été proposé par Kukkonen et Lampinen [64] comme une troisième version de l'algorithme à évolution différentielle généralisé nommé Generalized Differential Evolution (GDE). Les performances de GDE3 sont améliorées par rapport aux versions précédentes vis-à-vis de la distribution des solutions non dominées grâce à l'utilisation de la distance de crowding.

### 2.3.1 L'évolution différentielle

L'évolution différentielle (ED), développée par Storn et Price [65], est une méthode qui est inspirée par deux autres méthodes évolutionnaires ; les algorithmes génétiques et les stratégies d'évolution [66]. L'algorithme à évolution différentielle se base sur l'utilisation d'un opérateur combiné de croisement et de mutation, qui génère une nouvelle solution à partir de trois autres solutions potentielles.

L'algorithme 3 décrit le fonctionnement d'un algorithme à évolution différentielle standard. Une population est d'abord initialisée d'une façon aléatoire. Ensuite, pour chaque solution de la population courante,  $P_t$ , une nouvelle solution est générée en appliquant l'opérateur de l'évolution différentielle. La nouvelle solution est acceptée dans la prochaine population si elle est meilleure que la première. Cette opération est répétée jusqu'à ce que le critère d'arrêt soit satisfait.

Pour générer une solution  $u_t^i$  à partir de  $x_t^i$ , on sélectionne d'abord trois autres solutions de la population d'une façon aléatoire mais mutuellement différentes. En supposant que ces solutions sont  $x_t^{r1}$ ,  $x_t^{r2}$  et  $x_t^{r3}$ , la nouvelle solution  $u_t^i$  est alors calculée comme suit [67] :

**Algorithme 3** : Pseudo-code d'un algorithme à évolution différentielle

---

**Entrées** :  $N, F, CR$   
 $t \leftarrow 0$   
 Initialiser  $P_t$   
 Initialiser  $\mathbf{x}_{opt}$   
**Tant que** Critère d'arrêt non satisfait **faire**  
    $P_{t+1} \leftarrow \emptyset$   
   **Pour**  $i = 1 : N$  **faire**  
     Calculer  $\mathbf{y}_t^i$   
     **Si**  $\mathbf{y}_t^i \prec \mathbf{x}_t^i$  **alors**  
        $\mathbf{x}_{t+1}^i \leftarrow \mathbf{y}_t^i$   
     **sinon**  
        $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i$   
     **Fin**  
    $t \leftarrow t + 1$   
**Fin**  
 Mettre à jour  $\mathbf{x}_{opt}$   
**Fin**  
**Sorties** :  $\mathbf{x}_{opt}$

---

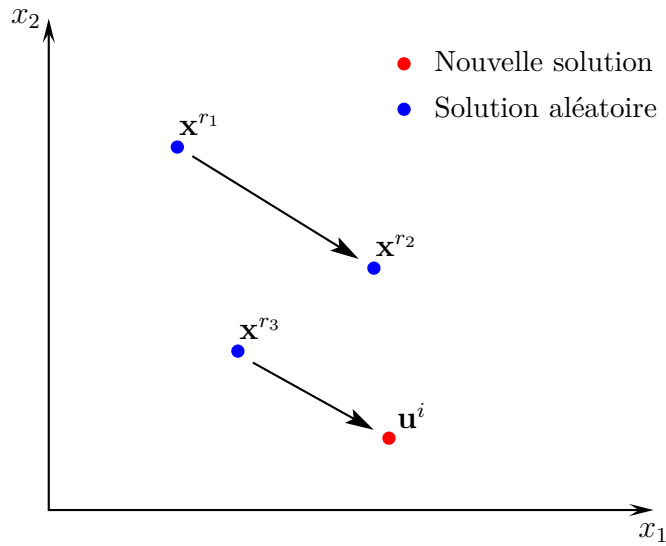


Figure 2.3 : Opérateur de l'évolution différentielle.

$$\forall j \leq n, u_{j,t}^i = \begin{cases} x_{j,t}^{r1} + F \cdot (x_{j,t}^{r2} - x_{j,t}^{r3}) & \text{si } r < CR \text{ ou } j = j_{rand} \\ x_{j,t}^i & \text{sinon} \end{cases} \quad (2.1)$$

où  $F$  et  $CR$  sont deux paramètres réels compris entre 0 et 1, désignant le poids différentiel et la probabilité de croisement, respectivement. Par ailleurs,  $j_{rand} \in \{1, \dots, n\}$  est un entier choisi d'une façon aléatoire, qui permet de garantir qu'au moins une variable de  $x_t^i$  soit différente de celle de  $u_t^i$ .

**Algorithme 4** : Pseudo-code de GDE3

---

```

Entrées :  $N, F, CR$ 
 $t \leftarrow 0$ 
Initialiser  $P_t$ 
Tant que Critère d'arrêt non satisfait faire
   $P_{t+1} \leftarrow \emptyset$ 
  Pour  $i = 1 : N$  faire
    Calculer  $\mathbf{y}_t^i$ 
    Si  $\mathbf{y}_t^i \prec_c \mathbf{x}_t^i$  alors
       $\mathbf{x}_{t+1}^i \leftarrow \mathbf{y}_t^i$ 
    sinon
       $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i$ 
    Fin
    Si  $(\forall j g_j(\mathbf{y}_t^i) \leq 0 \wedge \mathbf{x}_{t+1}^i == \mathbf{x}_t^i \wedge \mathbf{x}_t \not\prec \mathbf{y}_t^i)$  alors
       $P_{t+1} \leftarrow P_{t+1} \cup \mathbf{y}_t^i$ 
    Fin
    Si  $|P_{t+1}| > N$  alors
      Retirer de  $P_{t+1}$  les solutions qui possèdent les plus petites distances de crowding
    Fin
   $t \leftarrow t + 1$ 
Fin
Sorties :  $P_t$ 

```

---

L'évolution différentielle est une méthode qui est facile à implémenter et ne fait intervenir que deux paramètres. De plus, cette méthode s'est avérée très efficace pour la résolution des problèmes continus. Toutefois, la vitesse de convergence est relativement faible et le risque de stagnation est plus élevé.

### 2.3.2 Description de l'algorithme GDE3

Le fonctionnement de l'algorithme GDE3 est décrit par l'algorithme 4. Tout d'abord, une population  $P_0$  de  $N$  solutions potentielles est générée d'une façon aléatoire. Ensuite, à chaque itération  $t$ , une nouvelle solution  $\mathbf{u}_t^i$  est obtenue à partir de  $\mathbf{x}_t^i$  pour  $i = 1, \dots, N$ , d'après l'équation 2.1. Si la solution  $\mathbf{u}_t^i$  domine  $\mathbf{x}_t^i$ , alors  $\mathbf{u}_t^i$  remplace  $\mathbf{x}_t^i$  dans  $P_{t+1}$ , sinon on maintient  $\mathbf{x}_t^i$ . Dans le cas où ces deux solutions sont incomparables,  $P_{t+1}$  est augmentée avec  $\mathbf{u}_t^i$ . A la fin de la génération, la taille de  $P_{t+1}$  est réduite à  $N$  en éliminant séquentiellement les plus mauvaises solutions en termes de rang et éventuellement de distance de crowding.

## 2.4 Archived Multiobjective Simulated Annealing

L'algorithme AMOSA a été proposé par Bandyopadhyay et al. [68] en se basant sur la méthode du recuit simulé. Cet algorithme utilise une archive pour mémoriser les

**Algorithme 5** : Pseudo-code d'un algorithme de recuit simulé

---

```

Entrées :  $T_{max}$ ,  $T_{min}$ ,  $N_p$ ,  $\alpha$ 
 $T = T_{max}$ 
Générer une solution aléatoire  $\mathbf{x}$ 
Initialiser  $\mathbf{x}_{opt}$ 
Tant que  $T \geq T_{min}$  faire
    Pour  $i = 1 : N_p$  faire
        Générer  $\mathbf{y}$  à partir de  $\mathbf{x}$ 
        Si  $\mathbf{y}^i \prec \mathbf{x}^i$  alors
            |  $\mathbf{x}^i \leftarrow \mathbf{y}^i$ 
        sinon
            | Générer un nombre aléatoire  $r$ 
            | Si  $r < \exp(-\Delta f/T)$  alors
            | |  $\mathbf{x}^i \leftarrow \mathbf{y}^i$ 
            | Fin
        Fin
        Mettre à jour  $\mathbf{x}_{opt}$ 
    Fin
     $T \leftarrow \alpha T$ 
Fin
Sorties :  $\mathbf{x}_{opt}$ 

```

---

solutions non dominées, et une nouvelle mesure de dominance pour tester l'acceptation des nouvelles solutions durant le processus de recherche. Ainsi, le degré de dominance entre deux solutions quelconques  $\mathbf{x}$  et  $\mathbf{y}$  est défini comme suit :

$$\Delta_{\text{dom}_{\mathbf{x},\mathbf{y}}} = \prod_{i=1, f_i(\mathbf{x}) \neq f_i(\mathbf{y})}^m \frac{|f_i(\mathbf{x}) - f_i(\mathbf{y})|}{R_i} \quad (2.2)$$

où  $m$  est le nombre de fonctions objectifs du problème et  $R_i$  l'étendue de la  $i^{\text{ème}}$  fonction objectif du problème qui est calculée en prenant en considération les solutions  $\mathbf{x}$  et  $\mathbf{y}$  ainsi que celles de l'archive.

### 2.4.1 Le recuit simulé

Le recuit simulé (RS) est une technique de recherche locale développée par Kirkpatrick [69], en s'inspirant du processus de recuit utilisé dans la métallurgie. Ce dernier consiste à faire baisser progressivement la température d'un matériau dans le but de modifier sa structure métallique. Lorsque le matériau est à forte température, chaque particule possède une très grande énergie et peut effectuer de grands déplacements aléatoires dans la matière, le matériau peut donc prendre n'importe quelle structure. Lorsque la température est abaissée, chaque particule perd de l'énergie et sa capacité de déplacement est alors réduite, la structure du matériau devient de plus en plus difficile à modifier. Les différents états transitoires de refroidissement permettent ainsi d'obtenir des matériaux homogènes et de qualité optimale.

Dans l'algorithme du recuit simulé, la fonction objectif du problème représente l'énergie du matériau. L'algorithme commence par l'initialisation de la température et la génération d'une solution aléatoire  $\mathbf{x}$ . Cette solution est ensuite perturbée pour donner lieu à une nouvelle solution  $\mathbf{y}$ , si  $f(\mathbf{y}) < f(\mathbf{x})$  alors  $\mathbf{x}$  est remplacée par  $\mathbf{y}$ . Dans le cas contraire,  $\mathbf{y}$  est acceptée avec une certaine probabilité qui est fonction de la différence d'énergie  $\Delta f = f(\mathbf{y}) - f(\mathbf{x})$  et également de la valeur actuelle de la température. Cette procédure est répétée pendant plusieurs itérations en maintenant la température constante avant que celle-ci ne soit diminuée d'un palier. L'algorithme 5 montre le pseudo-code d'un algorithme de recuit simulé standard.

La température est un paramètre qui joue un rôle important dans le recuit simulé. Ce paramètre, désigné par  $T$ , intervient principalement dans le test d'acceptation des nouvelles solutions. Ainsi, la probabilité d'accepter une nouvelle solution qui est moins efficace que la solution courante est [2] :

$$p = \exp\left(-\frac{\Delta f}{T}\right) \quad (2.3)$$

La probabilité d'accepter une nouvelle solution est élevée lorsque la température est grande. Celle-ci doit donc être initialisée à une valeur suffisamment élevée afin d'explorer au maximum l'espace de recherche. Toutefois, si cette valeur est trop élevée, toutes les nouvelles solutions seront acceptées, ce qui allonge beaucoup le temps de recherche et diminue la rapidité de convergence. Par contre, si elle est trop basse, l'algorithme risque de converger rapidement vers un optimum local.

Le recuit simulé est une technique de recherche locale qui est facile à implémenter et en même temps très efficace pour la résolution des problèmes d'optimisation difficile. Toutefois, le choix des nombreux paramètres impliqués, en particulier la température initiale, peut devenir difficile pour certains problèmes.

## 2.4.2 Description de l'algorithme AMOSA

Le fonctionnement de AMOSA est décrit par l'algorithme 6. Dans un premier temps, l'archive est initialisée avec les solutions non dominées d'un ensemble de  $N$  solutions générées d'une façon aléatoire. Ensuite, la température initiale est calculée en exécutant l'algorithme pour un certain nombre d'itérations, où toutes les nouvelles solutions sont systématiquement acceptées, la température initiale est déduite en divisant la moyenne des différences positives de l'énergie par  $\log(2)$ .

La boucle principale de l'algorithme AMOSA commence en sélectionnant aléatoirement une solution  $\mathbf{x}$  de l'archive. Ensuite, à chaque itération une nouvelle solution  $\mathbf{y}$  est générée à partir de  $\mathbf{x}$ . Ainsi, en comparant  $\mathbf{x}$  et  $\mathbf{y}$ , trois cas sont envisageables :

**Algorithme 6** : Pseudo-code de AMOSA

---

**Entrées** :  $N, nt, T_{min}, N_p, \alpha$   
 Initialiser l'archive  $A$   
 Calculer  $T_{max}$   
 Sélectionner aléatoirement  $\mathbf{x} \in A$   
 $T \leftarrow T_{max}$   
**Tant que**  $T \geq T_{min}$  **faire**  
   **Pour**  $i = 1 : N_p$  **faire**  
     Générer une nouvelle solution  $\mathbf{y}$  à partir de  $\mathbf{x}$   
     Sélectionner une solution courante entre  $\mathbf{x}, \mathbf{y}$  et les solutions de l'archive  
     **Si**  $|A| > N$  **alors**  
       Appliquer le clustering sur  $A$   
     **Fin**  
   **Fin**  
    $T \leftarrow \alpha T$   
**Fin**  
**Sorties** :  $A$

---

**Cas 1** : la nouvelle solution  $\mathbf{y}$  est dominée par la solution courante  $\mathbf{x}$  et également par  $k$  solutions de l'archive. Dans ce cas,  $\mathbf{y}$  est acceptée avec la probabilité suivante :

$$\begin{cases} p = \frac{1}{1 + \exp(\Delta\text{dom}_{avg} \cdot T)} \\ \text{avec } \Delta\text{dom}_{avg} = \frac{(\sum_{i=1}^k \Delta\text{dom}_{i,\mathbf{y}}) + \Delta\text{dom}_{\mathbf{x},\mathbf{y}}}{k + 1} \end{cases} \quad (2.4)$$

**Cas 2** :  $\mathbf{x}$  et  $\mathbf{y}$  sont non dominées entre elles. Dans ce cas, si la solution  $\mathbf{y}$  est dominée par  $k$  ( $k \geq 1$ ) solutions de l'archive, la probabilité d'accepter  $\mathbf{y}$  est :

$$\begin{cases} p = \frac{1}{1 + \exp(\Delta\text{dom}_{avg} \cdot T)} \\ \text{avec } \Delta\text{dom}_{avg} = \frac{\sum_{i=1}^k (\Delta\text{dom}_{i,\mathbf{y}})}{k} \end{cases} \quad (2.5)$$

Dans le cas où la solution  $\mathbf{y}$  n'est dominée par aucune solution de l'archive, alors  $\mathbf{y}$  devient la solution courante.

**Cas 3** : La solution  $\mathbf{y}$  domine  $\mathbf{x}$ . Dans ce cas, si  $\mathbf{y}$  est dominée par  $k$  ( $k \geq 1$ ) solutions de l'archive, alors la solution de l'archive qui possède le plus petit degré de dominance par rapport à  $\mathbf{y}$ , noté  $\Delta\text{dom}_{min}$ , est sélectionnée avec la probabilité suivante :

$$p = \frac{1}{1 + \exp(-\Delta\text{dom}_{min} \cdot T)} \quad (2.6)$$

Dans le cas où  $\mathbf{y}$  est non dominée par rapport à toutes les solutions de l'archive, alors  $\mathbf{y}$  devient la solution courante.



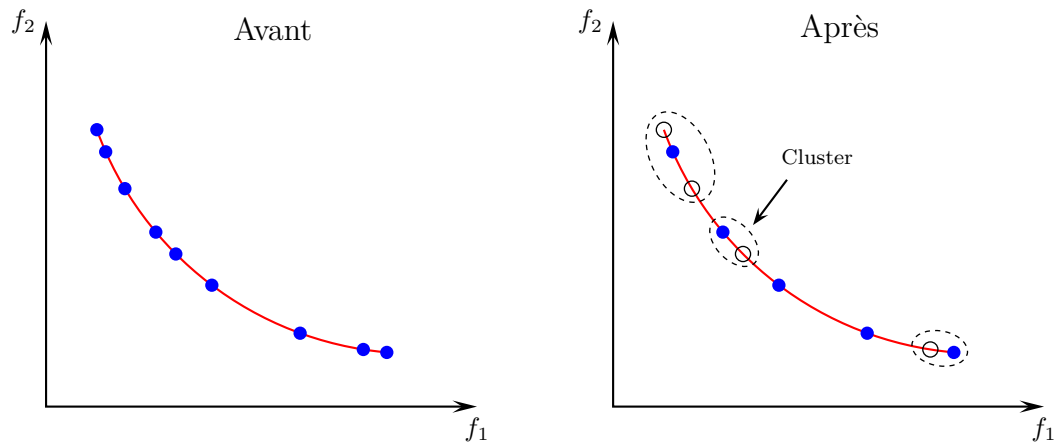


Figure 2.4 : Illustration de la procédure de clustering ( $N = 5$ ).

La procédure précédente est répétée un certain nombre de fois, ensuite la température actuelle est diminuée en la multipliant par un certain facteur  $\alpha$  ( $\alpha < 1$ ). Par ailleurs, si durant une itération quelconque la taille de l'archive dépasse  $N$ , la méthode de clustering, dite single-linkage clustering, est alors appliquée afin de réduire la taille de l'archive. Cette méthode consiste à répartir les solutions de l'archive en  $N$  groupes ou clusters, et en sélectionnant ensuite un représentant de chaque cluster. Les clusters sont identifiés un par un, en fusionnant à chaque fois les deux clusters les plus proches en termes de distance [70]. Cette procédure est illustrée par la figure 2.4.

## 2.5 Speed-Constrained Particle Swarm Optimization (SMPSO)

L'algorithme SMPSO a été proposé par Nebro et al. [71] en améliorant un autre algorithme multiobjectif basé sur l'optimisation par essaims de particules appelé Optimized Multiobjective Particle Swarm optimization (OMOPSO). La principale différence entre ces deux algorithmes réside dans le fait que SMPSO utilise un certain opérateur de constriction qui lui permet de mieux contrôler la convergence de l'algorithme et d'éviter ainsi les convergences prématurées.

### 2.5.1 L'optimisation par essaims de particules

Kennedy et Eberhart [72] ont proposé la technique d'optimisation par essaims de particules (OEP), dans le but de simuler le comportement socio-psychologique de certains groupements d'être vivants tels que les nuées d'oiseaux et les bancs de poissons. Cette technique s'appuie sur le concept de coopération entre plusieurs agents biologiques non développés, devant atteindre un objectif commun, dans un environnement donné. Les agents, appelés particules, se déplacent dans un espace à plusieurs dimensions en recherchant des sites intéressants, optimaux, par rapport à leurs objectifs. Chaque

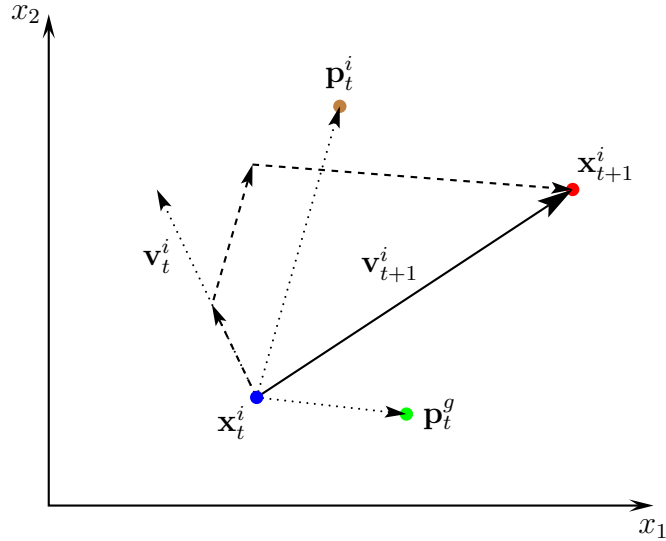


Figure 2.5 : Déplacement d'une particule dans l'OEP.

particule modifie sa position en fonction de sa vitesse actuelle, sa meilleure position mémorisée durant la recherche, ainsi que la meilleure position dans son voisinage ou dans l'ensemble des particules.

L'OEP peut être décrite par l'algorithme 7. Un ensemble de  $N$  particules, ou essaim, est d'abord initialisé aléatoirement. Chaque particule  $i$  représente une solution potentielle au problème et possède une valeur de position, définie par les variables de décisions du problème  $\mathbf{x}^i = (x_1^i, \dots, x_n^i)$ , ainsi qu'une vitesse de déplacement  $\mathbf{v}^i = (v_1^i, \dots, v_n^i)$ . A chaque itération, ces deux caractéristiques sont modifiées comme suit [73] :

$$\begin{cases} v_{j,t+1}^i = \omega v_{j,t}^i + c_1 R_1 (p_{j,t}^i - x_{j,t}^i) + c_2 R_2 (p_{j,t}^g - x_{j,t}^i) \\ x_{j,t+1}^i = x_{j,t}^i + v_{j,t+1}^i \end{cases} \quad (2.7)$$

où  $\omega$  est le facteur d'inertie,  $c_1$  et  $c_2$  sont les coefficients d'accélération et  $r_1, r_2 \in [0, 1]$  deux nombres aléatoires. Le facteur d'inertie  $\omega$  est un paramètre, choisi habituellement entre 0.4 et 0.9, qui permet de contrôler l'influence des allures antérieures d'une particule sur sa nouvelle vitesse. En diminuant la valeur de  $\omega$ , la recherche devient plus affinée mais moins diversifiée. Par ailleurs,  $c_1$  et  $c_2$  sont deux paramètres qui déterminent la tendance de la particule à s'orienter respectivement vers sa meilleure position personnelle  $\mathbf{p}^i$  et la meilleure position globale  $\mathbf{p}^g$  (on considère ici que le voisinage de chaque particule est l'ensemble entier des particules). Les paramètres  $c_1$  et  $c_2$  sont choisis typiquement autour de 2. Le principe de déplacement d'une particule dans l'espace de recherche à deux dimensions est illustré dans la figure 2.5.

L'OEP est utilisée pour résoudre différents types de problèmes réels, en particulier les problèmes à variables continues. Toutefois, elle présente un risque élevée de convergence prématurée et fait intervenir plusieurs paramètres de réglage.

**Algorithme 7** : Pseudo-code de l'optimisation par essaims de particules

---

**Entrées** :  $N, \omega, c_1, c_2$   
 $t \leftarrow 0$   
Initialiser  $P_t$   
Initialiser  $\mathbf{x}_{opt}$   
**Tant que** Critère d'arrêt non satisfait **faire**  
  **Pour**  $i = 1 : N$  **faire**  
    Calculer la vitesse  $\mathbf{v}_{t+1}^i$   
    Calculer la position  $\mathbf{x}_{t+1}^i$   
    Mettre à jour  $\mathbf{p}_{t+1}^i$   
  **Fin**  
  Mettre à jour  $\mathbf{p}_{t+1}^g$   
   $t \leftarrow t + 1$   
  Mettre à jour  $\mathbf{x}_{opt}$   
**Fin**  
**Sorties** :  $\mathbf{x}_{opt}$

---

## 2.5.2 Description de l'algorithme SMPSO

Le pseudo-code de SMPSO est donné par l'algorithme 8. Après avoir initialisé aléatoirement l'essaim de particules, une archive de solutions, appelées leaders, est alors formée avec les solutions non dominées de cet essaim. A chaque itération, les vitesses et les positions des particules sont calculées d'après l'équation 2.7. Ensuite, l'opérateur de mutation polynomiale, identique à celui qui est utilisé dans NSGA-II, est alors appliqué avec une certaine probabilité dans le but de diversifier la recherche. Une fois les particules évaluées, l'archive est mise à jour et le processus s'itère jusqu'à ce que le critère d'arrêt soit satisfait.

Pour pouvoir utiliser l'équation 2.7, chaque particule  $i$  de l'essaim doit choisir un leader  $\mathbf{p}_t^{i,g}$  de  $A_t$ . Ainsi, la distance de crowding de chaque particule dans  $A_t$  est d'abord calculée, ensuite un tournoi binaire sur la base de cette distance est appliqué pour choisir  $\mathbf{p}_t^{i,g}$ . Par ailleurs, afin de mettre à jour  $\mathbf{p}_t^i$ , la solution  $\mathbf{x}_t^i$  est comparée avec  $\mathbf{p}_t^i$  selon la dominance de Pareto. Ainsi, si la solution  $\mathbf{p}_t^i$  ne domine pas  $\mathbf{x}_t^i$ ,  $\mathbf{p}_t^i$  est alors remplacée par  $\mathbf{x}_t^i$ . Les vitesses des particules, définies dans l'équation 2.7, sont ensuite multipliées par le coefficient de constriction suivant [73] :

$$\chi = \frac{2}{\left| 2 - \phi - \sqrt{\phi(\phi - 4)} \right|} \quad (2.8)$$

où  $\phi = \phi_1 + \phi_2$ ,  $\phi_1 = c_1 r_1$  et  $\phi_2 = c_2 r_2$ . L'équation 2.8 n'est valide que si  $\phi > 4$ , ainsi on a  $0 < \chi < 1$ . En appliquant cet opérateur, l'oscillation des particules est diminuée durant la recherche, et la convergence vers un point stable est donc garantie. De plus, afin de mieux contrôler les déplacements aléatoires des particules, les vitesses de ces celles-ci sont limitées comme suit [73] :

**Algorithme 8** : Pseudo-code de SMPSO

---

**Entrées** :  $N, N_A, \omega, c_1, c_2$   
 $t \leftarrow 0$   
Initialiser  $P_t$   
Initialiser  $A_t$   
**Tant que** Critère d'arrêt non satisfait **faire**  
  **Pour**  $i = 1 : N$  **faire**  
    Calculer la vitesse  $\mathbf{v}_{t+1}^i$   
    Calculer la position  $\mathbf{x}_{t+1}^i$   
    Mettre à jour  $\mathbf{p}_{t+1}^i$   
  **Fin**  
  Mettre à jour  $\mathbf{p}_{t+1}^g$   
  Mettre à jour  $A_{t+1}$   
   $t \leftarrow t + 1$   
**Fin**  
**Sorties** :  $A_t$

---

$$v_{j,t}^i = \begin{cases} V_{max,j} & \text{si } v_{j,t}^i > V_{max,j} \\ -V_{max,j} & \text{si } v_{j,t}^i < -V_{max,j} \\ v_{j,t}^i & \text{sinon} \end{cases} \quad (2.9)$$

avec  $V_{max,j} = \frac{x_{max,j} - x_{min,j}}{2}$

A chaque itération de l'algorithme SMPSO, la nouvelle archive  $A_{t+1}$  est établie en sélectionnant les solutions non dominées de l'union des deux ensembles  $A_t$  et  $P_t$ . Si le nombre de ces solutions dépasse la taille limite de l'archive, les solutions avec les plus petites distances de crowding sont alors éliminées.

## 2.6 Conclusion

Dans ce chapitre, nous avons présenté quatre algorithmes multiobjectifs de référence. L'algorithme NSGA-II, qui est basé sur les algorithmes génétiques, est l'algorithme le plus connu et le plus utilisé actuellement pour résoudre des problèmes d'optimisation multiobjectif théoriques et réels. L'algorithme de l'évolution différentielle GDE3 est un algorithme qui est très simple à implémenter et ne fait intervenir que peu de paramètres. L'algorithme AMOSA, basé sur la méthode du recuit simulé, est l'une des méthodes de recherche locales les plus performantes. Enfin, l'algorithme d'optimisation par essaims de particules, SMPSO, est connu pour son efficacité à résoudre des problèmes d'optimisation à variables continues.

## Chapitre 3

# Développement d'un nouvel algorithme à évolution différentielle pour l'optimisation multiobjectif

## Chapitre 3

# Développement d'un nouvel algorithme à évolution différentielle pour l'optimisation multiobjectif

### 3.1 Introduction

Dans ce chapitre, nous développons un algorithme d'optimisation multiobjectif, nommé MODE-2A (Multi-Objective Differential Evolution using two Archives), en se basant sur le principe de l'évolution différentielle. Nous montrons que cet algorithme est capable de résoudre différents types de problèmes académiques d'une façon simple, efficace et robuste. Une étude comparative avec les autres méthodes de référence présentées dans le chapitre précédent est réalisée afin de valider les performances de l'algorithme proposé.

### 3.2 L'algorithme Multi-Objective Differential Evolution using two Archives (MODE-2A)

L'algorithme que nous proposons ici, MODE-2A, est basé essentiellement sur le principe de l'évolution différentielle. Cette métaheuristique a prouvé son efficacité à résoudre des problèmes d'optimisation continue avec seulement deux paramètres à régler ; le poids différentiel et la probabilité de croisement. Toutefois, le choix de ces derniers dépend sensiblement du problème traité. Ainsi, dans l'algorithme MODE-2A, ces deux paramètres ne sont pas fixés a priori par le concepteur mais varient aléatoirement durant la recherche, ce qui permet d'améliorer la flexibilité et la robustesse de l'algorithme. Par ailleurs, l'algorithme MODE-2A utilise trois ensembles de solutions dont l'un est l'ensemble principal de solutions potentielles, soit la population, et les deux autres ensembles sont des archives. Le rôle de chacun de ces ensembles est expliqué dans ce qui suit.

**La population :** contient les solutions potentielles du problème qui sont utilisées pour générer d'autres solutions en appliquant deux opérateurs de variation ; l'opérateur de l'évolution différentielle et un opérateur de diversification ou de mutation.

**Algorithme 9** : Pseudo-code de MODE-2A

---

**Entrées** :  $N, N_d, N_p$   
 Initialiser la population  $P$   
 Initialiser les archives  $A_d$  et  $A_p$   
 $t \leftarrow 0$   
**Tant que** *Critère d'arrêt non satisfait* **faire**  
     Générer une nouvelle solution  $\mathbf{y}$  à partir de  $P_t$   
     Appliquer la mutation sur  $\mathbf{y}$   
     Mettre à jour  $A_d$   
     Mettre à jour  $A_p$   
     Insérer des solutions de  $A_d$  et  $A_p$  dans  $P_t$   
      $t \leftarrow t + 1$   
**Fin**  
**Si**  $|A_p| \geq N$  **alors**  
     Réduire  $A_p$   
**Fin**  
**Sorties** :  $A_p$

---

**L'archive de diversité** : contient un nombre réduit de solutions non dominées afin de maintenir la diversité des solutions dans la population.

**L'archive principale** : utilisée pour mémoriser toutes les solutions non dominées découvertes durant la recherche. En pratique, la taille de cette archive est limitée, mais à une valeur beaucoup plus élevée que celle de la population.

### 3.2.1 Description de l'algorithme MODE-2A

Le pseudo-code de MODE-2A est donné par l'algorithme 9. La population initiale,  $P_0$ , est générée aléatoirement et les deux archives sont initialisées avec les solutions non dominées de  $P_0$ . Ensuite, à chaque itération, une nouvelle solution  $\mathbf{x}$  est générée en appliquant l'opérateur de l'évolution différentielle sur quatre solutions distinctes sélectionnées d'une façon aléatoire à partir de la population courante. Dans ce cas, les valeurs des deux paramètres  $F$  et  $CR$  sont choisies aléatoirement entre 0 et 1. De plus, l'opérateur de mutation polynomiale, identique à celui qui est utilisé dans l'algorithme NSGA-II, est également appliqué sur  $\mathbf{x}$ .

Les deux archives  $A_d$  et  $A_p$  sont mises à jour en prenant en considération la nouvelle solution. Ainsi, la solution  $\mathbf{x}$  est acceptée dans l'une des deux archives si elle est non dominée par toutes les solutions de cette archive. De plus, les solutions qui sont éventuellement dominées par  $\mathbf{x}$  sont retirées des archives. Dans le cas où le nombre de solutions dans  $A_d$  dépasse la taille limite de cet ensemble, la solution avec la plus petite distance de crowding est alors retirée. Par ailleurs, si la taille de l'archive principale est dépassée, une solution aléatoire est simplement éliminée. La figure 3.1 illustre les étapes de fonctionnement de MODE-2A pour une seule itération.

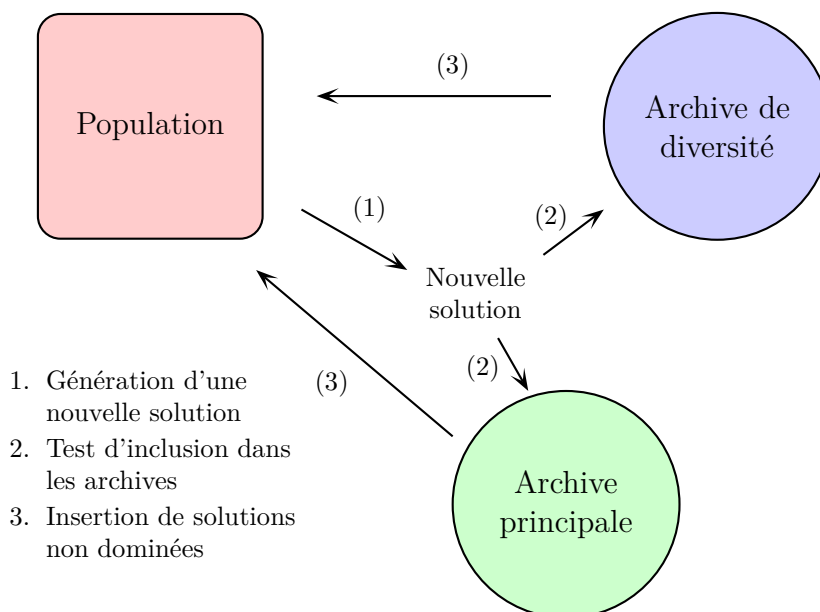


Figure 3.1 : Fonctionnement de l'algorithme MODE-2A pour une itération.

A la fin de chaque itération, la population principale est mise à jour en considérant les solutions des deux archives. Ainsi, un nombre aléatoire de solutions tirées de l'archive  $A_d$  sont insérées dans la population d'une façon aléatoire. Toutefois, ce nombre est limité à 30% de la taille de l'archive. De plus, les solutions de  $A_d$  qui sont choisies pour être insérées dans la population sont celles qui possèdent les plus grandes distances de crowding. Par ailleurs, une seule solution, tirée aléatoirement de l'archive  $A_p$ , est insérée dans la population.

A la fin de l'algorithme, les solutions des deux archives sont combinées pour former l'ensemble non dominé final. Si le nombre des solutions obtenues est supérieur à  $N$ , les solutions supplémentaires sont alors éliminées en appliquant la procédure de réduction proposée par Zitzler [74]. Cette procédure élimine les solutions supplémentaires, une par une, en sélectionnant à chaque fois la solution la plus encombrée dans l'espace des objectifs. Un exemple d'application de cette procédure est donné dans la figure 3.2.

### 3.2.2 Discussion

L'algorithme MODE-2A est une méthode d'optimisation globale qui s'inspire principalement de l'évolution différentielle. Afin d'accélérer la recherche, un certain nombre de solutions non dominées sont insérées, à chaque itération, dans la population courante. Ceci peut rapidement entraîner une perte de diversité dans la population et par conséquent provoquer une convergence prématurée. Afin de compenser ce risque, l'opérateur de mutation polynomiale est alors employé. Les principaux avantages de l'algorithme MODE-2A par rapport aux autres algorithmes de référence sont résumés dans ce qui suit.



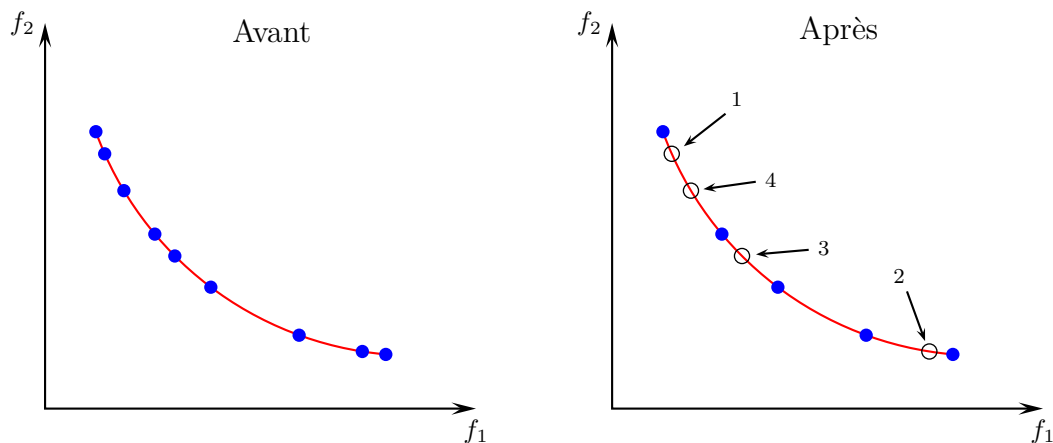


Figure 3.2 : Réduction de la taille de l'archive ( $N = 5$ ).

- L'algorithme MODE-2A ne fait intervenir que trois paramètres : les tailles des deux archives et un paramètre relatif à l'opérateur de mutation. Toutefois, le paramètre le plus important est celui qui fixe la taille de l'archive de diversité. En effet, si cette taille est trop grande, la diversité des solutions est améliorée et le risque de convergences prématurées est diminué. Par contre, la convergence de l'algorithme devient moins rapide vis-à-vis du nombre d'évaluations des fonctions objectifs.
- L'utilisation d'une archive de diversité dont la taille est réduite par rapport à celle de la population principale permet de simplifier le maintien de la diversité des solutions non dominées. En effet, le calcul de la distance de crowding devient dans ce cas moins compliqué et plus rapide.
- L'utilisation d'une deuxième archive d'une taille plus grande que celle de la population permet de diminuer le temps de calcul de l'algorithme et également d'améliorer la convergence et la distribution des solutions sur le front non dominé final.

### 3.3 Evaluation de l'algorithme MODE-2A

Dans le but d'évaluer les performances de l'algorithme MODE-2A, nous effectuons une étude comparative avec les algorithmes de référence NSGA-II, GDE3, AMOSA et SMPSO. Pour ce faire, nous nous proposons la résolution de certains problèmes académiques présentant différents types de difficultés pour les algorithmes d'optimisation. Par ailleurs, pour pouvoir évaluer les performances de ces algorithmes, nous employons plusieurs métriques de performance afin de mesurer la qualité des solutions non dominées obtenues par chaque algorithme.

#### 3.3.1 Problèmes de test

Les problèmes de test considérés dans cette étude sont les deux séries de problèmes de minimisation désignées par ZDT [70] et DTLZ [75]. La première série concerne des

problèmes à deux fonctions objectifs, tandis que dans la deuxième série le nombre de fonctions objectifs est variable. La série ZDT contient 5 problèmes continus à deux fonctions objectifs, présentant différentes caractéristiques telles que la convexité, la non-convexité, la discontinuité et la multimodalité. Les différents problèmes qui de cette série sont énoncés dans ce qui suit.

### Le problème ZDT1

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = x_1 \\ f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{x_1/g(\mathbf{x})}\right] \end{cases} \\ \text{avec } & g(\mathbf{x}) = 1 + 9 \cdot (\sum_{i=2}^n x_i) / (n - 1) \end{aligned} \quad (3.1)$$

où  $n = 30$  et  $x_i \in [0, 1]$  pour  $i = 1, \dots, n$ . Le front de Pareto est convexe et correspond à  $g = 1$ , c'est-à-dire  $0 \leq x_1 \leq 1$  et  $x_i = 0$  pour  $i = 2, \dots, n$ .

### Le problème ZDT2

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = x_1 \\ f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - (x_1/g(\mathbf{x}))^2\right] \end{cases} \\ \text{avec } & g(\mathbf{x}) = 1 + 9 \cdot (\sum_{i=2}^n x_i) / (n - 1) \end{aligned} \quad (3.2)$$

où  $n = 30$  et  $x_i \in [0, 1]$  pour  $i = 1, \dots, n$ . Le front de Pareto est non-convexe et correspond à  $g = 1$ , c'est-à-dire  $0 \leq x_1 \leq 1$  et  $x_i = 0$  pour  $i = 2, \dots, n$ .

### Le problème ZDT3

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = x_1 \\ f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{x_1/g(\mathbf{x})} - \frac{x_1}{g(\mathbf{x})} \sin(10\pi x_1)\right] \end{cases} \\ \text{avec } & g(\mathbf{x}) = 1 + 9 \cdot (\sum_{i=2}^n x_i) / (n - 1) \end{aligned} \quad (3.3)$$

où  $n = 30$  et  $x_i \in [0, 1]$  pour  $i = 1, \dots, n$ . Le front de Pareto est discontinu et correspond à  $g = 1$ , c'est-à-dire  $0 \leq x_1 \leq 1$  et  $x_i = 0$  pour  $i = 2, \dots, n$ .

### Le problème ZDT4

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = x_1 \\ f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - \sqrt{x_1/g(\mathbf{x})}\right] \end{cases} \\ \text{avec } & g(\mathbf{x}) = 1 + 10(n - 1) + \sum_{i=2}^n (x_i^2 - 10 \cos(4\pi x_i)) \end{aligned} \quad (3.4)$$

où  $n = 10$  et  $x_1 \in [0, 1]$  et  $x_i \in [-5, 5]$  pour  $i = 2, \dots, n$ . Le front de Pareto est convexe et correspond à  $g = 1$ , c'est-à-dire  $0 \leq x_1 \leq 1$  et  $x_i = 0$  pour  $i = 2, \dots, n$ . Ce problème est multimodal et contient  $21^9$  fronts locaux.

### Le problème ZDT6

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = 1 - \exp(-4x_1)\sin^6(6\pi x_1) \\ f_2(\mathbf{x}) = g(\mathbf{x}) \left[1 - (x_1/g(\mathbf{x}))^2\right] \end{cases} \\ \text{avec } & g(\mathbf{x}) = 1 + 9 \cdot \left[ (\sum_{i=2}^n x_i) / (n-1) \right]^{0.25} \end{aligned} \quad (3.5)$$

où  $n = 30$  et  $x_i \in [0, 1]$  pour  $i = 1, \dots, n$ . Le front de Pareto est non convexe et correspond à  $g = 1$ , c'est-à-dire  $0 \leq x_1 \leq 1$  et  $x_i = 0$  pour  $i = 2, \dots, n$ . Ce problème est caractérisé par une distribution non-uniforme des solutions Pareto-optimales.

La série DTLZ est constituée de 9 problèmes de minimisation dont deux sont des problèmes avec contraintes. Le nombre de fonctions objectifs  $M$  est variable, toutefois dans ce travail on se limite à  $M = 2$  et  $M = 3$ . Par ailleurs, le nombre de variables de décision est  $n = M + k - 1$ , où  $k$  est un paramètre entier. Dans tous les problèmes de cette série, les variables de décision sont telles que  $x_i \in [0, 1]$  et  $\mathbf{x}_M$  représente les  $k$  dernières variables de décision.

### Le problème DTLZ1

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = \frac{1}{2}x_1x_2 \cdots x_{M-1}(1 + g(\mathbf{x}_M)) \\ f_2(\mathbf{x}) = \frac{1}{2}x_1x_2 \cdots (1 - x_{M-1})(1 + g(\mathbf{x}_M)) \\ \vdots \\ f_{M-1}(\mathbf{x}) = \frac{1}{2}x_1(1 - x_2)(1 + g(\mathbf{x}_M)) \\ f_M(\mathbf{x}) = \frac{1}{2}(1 - x_1)(1 + g(\mathbf{x}_M)) \end{cases} \\ \text{avec } & g(\mathbf{x}_M) = 100 \left[ k + \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)) \right] \end{aligned} \quad (3.6)$$

Les solutions optimales de ce problème sont données par  $x_i = 0.5$ , pour  $x_i \in \mathbf{x}_M$ . Le front de Pareto est un hyperplan d'équation  $\sum_{m=1}^M f_m = 0.5$ . Ce problème est multimodal et présente  $(11^k - 1)$  fronts locaux. La valeur  $k = 5$  est considérée dans ce travail.

### Le problème DTLZ2

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1\pi/2) \cdots \cos(x_{M-2}\pi/2) \cos(x_{M-1}\pi/2) \\ f_2(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1\pi/2) \cdots \cos(x_{M-2}\pi/2) \sin(x_{M-1}\pi/2) \\ f_3(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1\pi/2) \cdots \sin(x_{M-2}\pi/2) \\ \vdots \\ f_M(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \sin(x_1\pi/2) \end{cases} \\ \text{avec } & g(\mathbf{x}_M) = \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2 \end{aligned} \quad (3.7)$$

Les solutions optimales sont données par  $x_i = 0.5$ , pour  $x_i \in \mathbf{x}_M$ . Le front de Pareto

est non-convexe, d'équation  $\sum_{m=1}^M f_m^2 = 1$ . La valeur  $k = 40$  est considérée dans ce travail.

### Le problème DTLZ3

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1\pi/2) \cdots \cos(x_{M-2}\pi/2) \cos(x_{M-1}\pi/2) \\ f_2(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1\pi/2) \cdots \cos(x_{M-2}\pi/2) \sin(x_{M-1}\pi/2) \\ f_3(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1\pi/2) \cdots \sin(x_{M-2}\pi/2) \\ \vdots \\ f_M(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \sin(x_1\pi/2) \end{cases} \\ \text{avec } & g(\mathbf{x}_M) = 100 \left[ k + \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2 - \cos(20\pi(x_i - 0.5)) \right] \end{aligned} \quad (3.8)$$

Les solutions optimales sont données par  $x_i = 0.5$ , pour  $x_i \in \mathbf{x}_M$ . Le front de Pareto est non-convexe, d'équation  $\sum_{m=1}^M f_m^2 = 1$ . Ce problème est multimodal et présente  $(3^k - 1)$  fronts locaux. La valeur  $k = 5$  est considérée dans ce travail.

### Le problème DTLZ4

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1^\alpha\pi/2) \cdots \cos(x_{M-2}^\alpha\pi/2) \cos(x_{M-1}^\alpha\pi/2) \\ f_2(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1^\alpha\pi/2) \cdots \cos(x_{M-2}^\alpha\pi/2) \sin(x_{M-1}^\alpha\pi/2) \\ f_3(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(x_1^\alpha\pi/2) \cdots \sin(x_{M-2}^\alpha\pi/2) \\ \vdots \\ f_M(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \sin(x_1^\alpha\pi/2) \end{cases} \\ \text{avec } & g(\mathbf{x}_M) = \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2 \end{aligned} \quad (3.9)$$

Les solutions optimales sont données par  $x_i = 0.5$ , pour  $x_i \in \mathbf{x}_M$ . Le front de Pareto est non-convexe, d'équation  $\sum_{m=1}^M f_m^2 = 1$ . Ce problème présente une densité non-uniforme des solutions Pareto-optimales. Les valeurs  $k = 20$  et  $\alpha = 100$  sont considérées ici.

### Le problème DTLZ5

$$\begin{aligned} \text{Minimiser } & \begin{cases} f_1(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(\theta_1\pi/2) \cdots \cos(\theta_{M-2}\pi/2) \cos(\theta_{M-1}\pi/2), \\ f_2(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(\theta_1\pi/2) \cdots \cos(\theta_{M-2}\pi/2) \sin(\theta_{M-1}\pi/2), \\ f_3(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(\theta_1\pi/2) \cdots \sin(\theta_{M-2}\pi/2), \\ \vdots \\ f_M(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \sin(\theta_1\pi/2), \end{cases} \\ \text{avec } & \theta_i = \frac{\pi}{4(1+g(\mathbf{x}_M))} (1 + 2g(\mathbf{x}_M)x_i), \quad i = 2, 3, \dots, M-1, \\ & g(\mathbf{x}_M) = \sum_{x_i \in \mathbf{x}_M} (x_i - 0.5)^2. \end{aligned} \quad (3.10)$$

Les solutions optimales sont données par  $x_i = 0.5$ , pour  $x_i \in \mathbf{x}_M$ . Le front de Pareto

est tel que  $f_1 = f_2 = \dots = f_{M-1}$  et  $\sum_{m=1}^M f_m^2 = 1$ . La valeur  $k = 50$  est considérée dans ce travail.

### Le problème DTLZ6

$$\begin{aligned} \text{Minimiser} \quad & \begin{cases} f_1(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(\theta_1\pi/2) \cdots \cos(\theta_{M-2}\pi/2) \cos(\theta_{M-1}\pi/2) \\ f_2(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(\theta_1\pi/2) \cdots \cos(\theta_{M-2}\pi/2) \sin(\theta_{M-1}\pi/2) \\ f_3(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \cos(\theta_1\pi/2) \cdots \sin(\theta_{M-2}\pi/2) \\ \vdots \\ f_M(\mathbf{x}) = (1 + g(\mathbf{x}_M)) \sin(\theta_1\pi/2) \end{cases} \\ \text{avec} \quad & \theta_i = \frac{\pi}{4(1+g(\mathbf{x}_M))} (1 + 2g(\mathbf{x}_M)x_i), \quad i = 2, 3, \dots, M-1, \\ & g(\mathbf{x}_M) = \sum_{x_i \in \mathbf{x}_M} x_i^{0.1} \end{aligned} \quad (3.11)$$

Les solutions optimales sont données par  $x_i = 0$ , pour  $x_i \in \mathbf{x}_M$ . Le front de Pareto est tel que  $f_1 = f_2 = \dots = f_{M-1}$  et  $\sum_{m=1}^M f_m^2 = 1$ . La valeur  $k = 50$  est considérée dans ce travail.

### Le problème DTLZ7

$$\begin{aligned} \text{Minimiser} \quad & \begin{cases} f_1(\mathbf{x}) = x_1 \\ f_2(\mathbf{x}) = x_2 \\ \vdots \\ f_{M-1}(\mathbf{x}) = x_{m-1} \\ f_M(\mathbf{x}) = (1 + g(\mathbf{x}_M))h(f_1, f_2, \dots, f_{M-1}, g) \end{cases} \\ \text{avec} \quad & g(\mathbf{x}_M) = 1 + \frac{9}{k} \sum_{x_i \in \mathbf{x}_M} x_i \\ & h(f_1, f_2, \dots, f_{M-1}, g) = M - \sum_{i=1}^{M-1} \left[ \frac{f_i}{1+g} (1 + \sin(3\pi f_i)) \right] \end{aligned} \quad (3.12)$$

Les solutions optimales de ce problème sont données par  $x_i = 0$ , pour  $x_i \in \mathbf{x}_M$ . Le front de Pareto est non linéaire et discontinu. Ce problème présente  $(2^M - 1)$  régions déconnectées dans l'espace de recherche. La valeur  $k = 50$  est considérée dans ce travail.

### Le problème DTLZ8

$$\begin{aligned} \text{Minimiser} \quad & f_j(\mathbf{x}) = \frac{1}{\lfloor \frac{n}{M} \rfloor} \sum_{i=\lfloor (j-1)\frac{n}{M} \rfloor}^{\lfloor j\frac{n}{M} \rfloor} x_i, \quad j = 1, \dots, M \\ \text{avec} \quad & g_j(\mathbf{x}) = f_M(\mathbf{x}) + 4f_j(\mathbf{x}) - 1 \geq 0, \quad j = 1, \dots, (M-1) \\ & g_M(\mathbf{x}) = 2f_M(\mathbf{x}) + \min_{\substack{i \neq j \\ i, j=1}}^{M-1} [f_i(\mathbf{x}) + f_j(\mathbf{x})] - 1 \geq 0 \end{aligned} \quad (3.13)$$

Le front de Pareto est une combinaison d'une ligne droite et d'un hyperplan. La droite est l'intersection des  $M-1$  premières contraintes et l'hyperplan est défini par la contrainte  $g_M$ . Dans notre cas, on prend  $n = 10M$ .

### Le problème DTLZ9

$$\begin{aligned} \text{Minimiser } f_j(\mathbf{x}) &= \sum_{i=\lfloor (j-1)\frac{n}{M} \rfloor}^{\lfloor j\frac{n}{M} \rfloor} x_i^0 \cdot 1, \quad j = 1, \dots, M \\ \text{avec } g_j(\mathbf{x}) &= f_M^2(\mathbf{x}) + f_j^2(\mathbf{x}) - 1 \geq 0, \quad j = 1, \dots, (M-1) \end{aligned} \quad (3.14)$$

Le front de Pareto est le même que celui du problème DTLZ5. La valeur  $n = 10M$  est considérée dans ce travail.

### 3.3.2 Métriques de performance

Les métriques de performance mesurent, selon plusieurs critères, la qualité des solutions non dominées obtenues par un algorithme d'optimisation quelconque. Les critères les plus importants sont ceux qui reflètent la convergence et la diversité des solutions non dominées. La convergence mesure le degré d'approche du front de Pareto dans l'espace des objectifs, tandis que la diversité regroupe deux aspects; la distribution et la répartition. Une bonne distribution est celle qui permet de découvrir toutes les parties du front de Pareto, et des solutions bien réparties signifie que les distances qui les séparent sont uniformes (figure 3.3).

Dans cette étude, nous utilisons quatre métriques de performance. La première est la métrique de convergence désignée par  $GD$  (Generational Distance) proposée par Van Veldhuizen [76]. Cette métrique calcule la moyenne des distances minimales entre chaque solution du front non dominé  $\mathcal{F}$  et celles du front de Pareto  $\mathcal{FP}$  :

$$GD = \frac{\sqrt{\sum_{i=1}^N d_i^2}}{N} \quad (3.15)$$

où  $d_i$  est la distance minimale entre la  $i^{\text{ème}}$  solution du front non dominé et les solutions du front de Pareto, et  $N$  est le nombre de solutions dans le front non dominé.

La deuxième métrique est l'hypervolume  $H$ , proposée par Zitzler [70], qui mesure simultanément les performances de convergence et de diversité. Cette métrique calcule le volume délimité par les solutions de  $\mathcal{F}$  et un certain point de référence :

$$H = \left\{ \bigcup_i a_i \mid \forall \mathbf{x}_i \in \mathcal{F} \right\} \quad (3.16)$$

où  $a_i$  est l'hypervolume formé par la solution  $\mathbf{x}_i$  et le point de référence. Un exemple de calcul de cette métrique est donné dans la figure 3.4.

La troisième métrique utilisée est la métrique d'espacement  $S$  (Spacing), proposée par Schott [77], qui mesure l'uniformité de la répartition des solutions du front non dominé dans l'espace des objectifs. Cette métrique est calculée de la façon suivante :

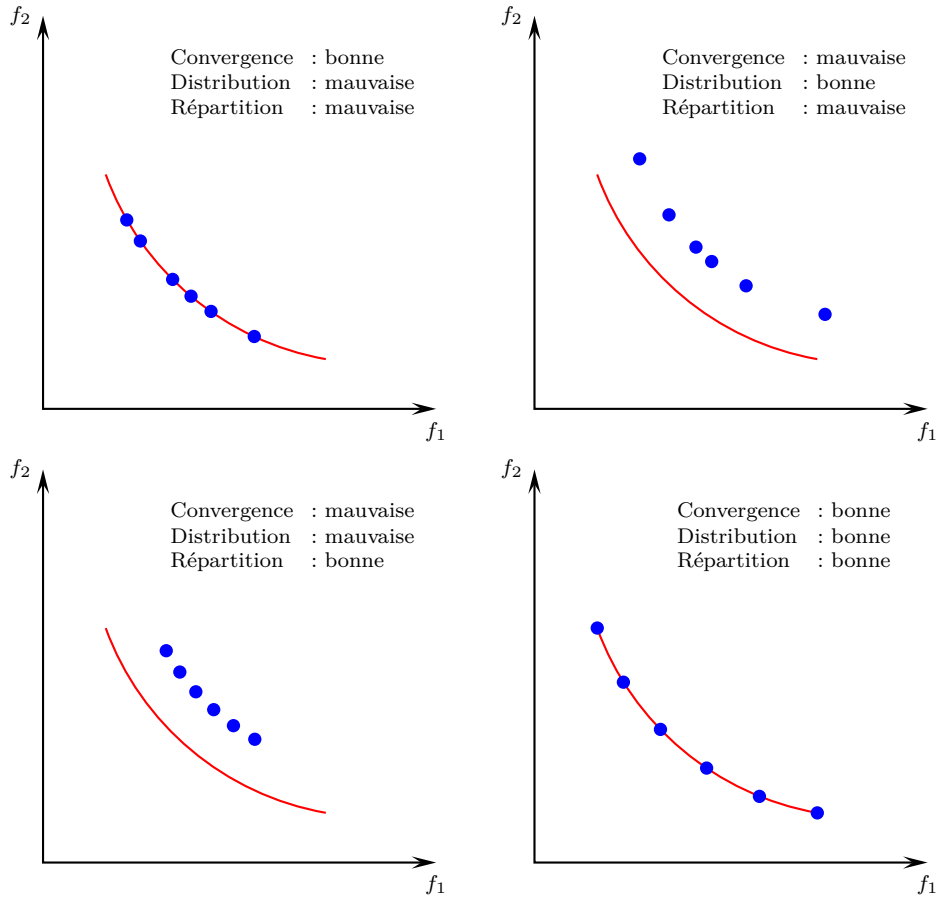


Figure 3.3 : Aspects de la qualité des solutions non dominées.

$$S = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (\bar{d} - d_i)^2} \quad (3.17)$$

où  $d_i$  est la distance minimale entre la  $i^{\text{ème}}$  solution du front non dominé et la solution la plus proche appartenant au même front, et  $\bar{d}$  est la moyenne de toutes les distances  $d_i$ .

La dernière métrique utilisée est une métrique qui combine les mesures de convergence, de distribution et de répartition à travers une seule valeur numérique. Une première définition de cette métrique, notée  $\bar{d}_{\max}$ , a été déjà proposée dans [78]. Dans le cadre de ce travail, nous proposons d'améliorer cette métrique afin d'avoir une mesure plus précise de la qualité globale des solutions non dominées.

Ainsi, pour calculer la valeur de  $\bar{d}_{\max}$ , nous devons d'abord calculer les distances minimales entre les solutions Pareto-optimales et celles du front non dominé, en attribuant à chaque solution Pareto-optimale  $\mathbf{x}^*$  un indice de répartition  $i$  qui identifie la solution de  $\mathcal{F}$  la plus proche de  $\mathbf{x}^*$ . Ensuite, l'ensemble des solutions Pareto-optimales est divisé en plusieurs groupes tels que les éléments de chaque groupe partagent le même indice  $i$ . La valeur de  $\bar{d}_{\max}$  est ensuite calculée comme suit :

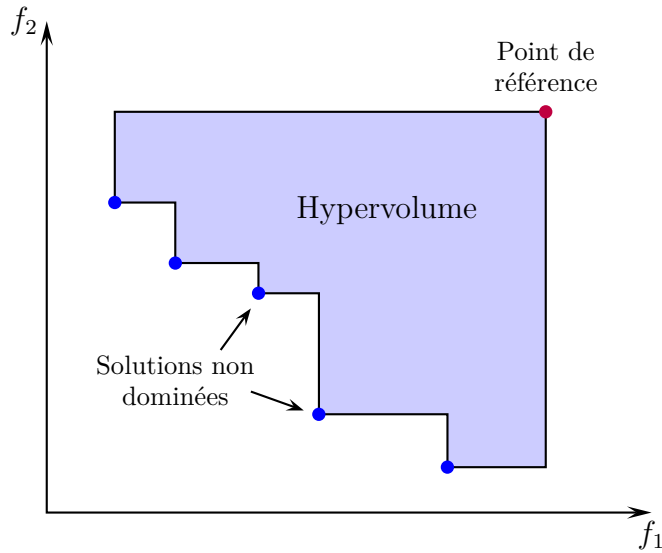


Figure 3.4 : Illustration du calcul de l'hypervolume.

$$\bar{d}_{\max} = \left( \frac{1}{L} \sum_{i=1}^L d_i \right) \cdot \left( 1 + \sum_{i=1}^M d_i^e \right) \quad (3.18)$$

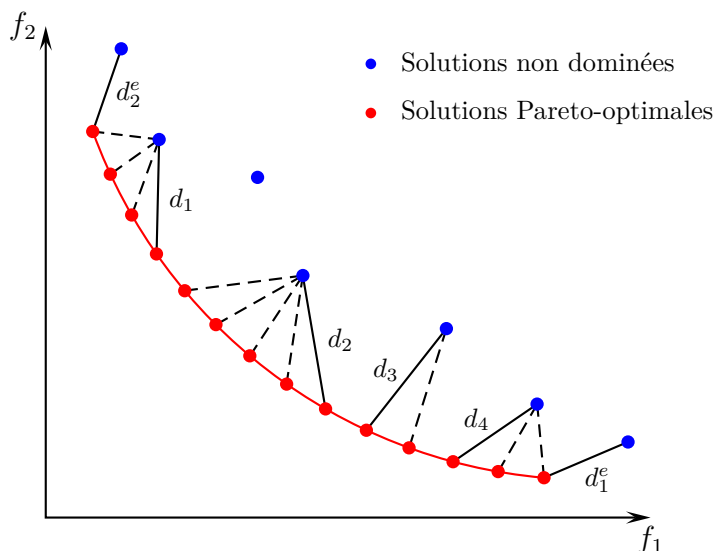
où  $d_i$  est la distance maximale dans le  $i^{\text{ème}}$  groupe, et  $d_i^e$  la distance entre les solutions extrêmes des fronts  $\mathcal{F}$  et celles de  $\mathcal{FP}$  selon la  $i^{\text{ème}}$  fonction objectif. Le deuxième facteur dans l'équation 3.18 est ajouté dans le but d'améliorer la mesure de la distribution des solutions non dominées par rapport au front de Pareto. En effet, une bonne distribution des solutions non dominées implique que les solutions extrêmes des deux fronts  $\mathcal{F}$  et  $\mathcal{FP}$  soient les plus proches possibles. Le calcul de la métrique  $\bar{d}_{\max}$  est illustré dans la figure 3.5. Dans ce cas, les solutions du front de Pareto sont divisées en cinq groupes.

### 3.3.3 Résultats de simulation

L'algorithme MODE-2A est comparé avec les quatre autres algorithmes de référence NSGA-II, GDE3, AMOSA et SMPSO, en considérant la résolution des deux séries de problèmes de test ZDT et DTLZ, et en utilisant les métriques  $GD$ ,  $H$ ,  $\bar{d}_{\max}$  et  $S$ . Par ailleurs, pour comparer équitablement les performances de ces algorithmes, chacun de ces derniers est exécuté 30 fois et les valeurs moyennes des métriques sont calculées pour être utilisées dans la comparaison. De plus, les mêmes ensembles de solutions initiales sont utilisés par tous les algorithmes.

Les algorithmes NSGA-II, GDE3, AMOSA et SMPSO, sont utilisés de la même façon que dans [62, 64, 68, 71] en considérant des populations et/ou des archives de taille 100. Par ailleurs, dans l'algorithme MODE-2A, la taille de la population est 100, celle de l'archive principale est 1000, tandis que la taille de l'archive de diversité est égale à 5 pour les problèmes à deux objectifs et à 15 pour les problèmes à trois objectifs.



Figure 3.5 : Illustration du calcul de la métrique  $\bar{d}_{\max}$ .

Les résultats des tests effectués sont résumés dans les tableaux 3.1 à 3.4, concernant les métriques  $GD$ ,  $H$ ,  $S$  et  $\bar{d}_{\max}$ , respectivement. Afin de comparer les performances des cinq algorithmes utilisés, le classement de chaque algorithme vis-à-vis des autres algorithmes est donné pour chaque problème de test ainsi que pour l'ensemble des problèmes.

Les résultats obtenus pour la métrique de convergence  $GD$  montrent que l'algorithme proposé, MODE-2A, offre les meilleures performances pour la plupart des problèmes de test, tandis que le classement général des cinq algorithmes est le suivant : 1-MODE-2A, 2-AMOSA, 3-GDE3, 4-SMPSO, 5-NSGA-II. Nous remarquons aussi que les meilleurs résultats pour les problèmes à deux objectifs sont obtenus par MODE-2A, et pour les problèmes à trois objectifs les meilleurs résultats sont ceux de l'algorithme AMOSA.

Concernant la métrique  $H$ , les meilleurs résultats sont également obtenus avec MODE-2A et le classement des cinq algorithmes est : 1-MODE-2A, 2-GDE3, 3-SMPSO, 4-AMOSA 5-NSGA-II. Ce résultat montre que l'algorithme MODE-2A est capable de fournir des solutions plus efficaces et plus diversifiées par rapport aux autres algorithmes. Nous constatons également que les performances des trois algorithmes GDE3, SMPSO et AMOSA sont similaires pour cette métrique.

Pour la métrique de répartition  $S$ , le classement général est le suivant : 1-MODE-2A, 2-GDE3, 3-NSGA-II, 4-SMPSO et 5-AMOSA. Dans ce cas, la différence de performance entre MODE-2A et les autres algorithmes est plus significative. Ces résultats montrent que les solutions obtenues avec MODE-2A sont mieux réparties sur les fronts non dominés par rapport aux solutions des autres algorithmes. Ceci est dû principalement à l'emploi d'une archive principale de grande taille, ce qui a permis d'augmenter le choix des solutions non dominées à retenir à la fin de la recherche.

Tableau 3.1 : Résultats de la métrique  $GD$ .

	<b>NSGAI</b>		<b>GDE3</b>		<b>AMOS</b>		<b>SMPSO</b>		<b>MODE-2A</b>	
ZDT1	1.03e-001	4	1.03e-001	3	2.78e-003	2	2.42e-001	5	7.40e-004	1
ZDT2	2.16e-001	4	1.76e-001	3	8.87e-003	2	2.66e-001	5	6.15e-004	1
ZDT3	5.43e-002	3	5.90e-002	4	3.82e-003	2	2.20e-001	5	6.41e-004	1
ZDT4	4.27e+000	4	1.31e+001	5	6.46e-002	2	2.43e-002	1	2.43e-001	3
ZDT6	3.80e+000	3	4.21e+000	4	1.07e-001	2	4.96e+000	5	2.60e-002	1
DTLZ12	4.19e-001	4	2.12e-003	1	4.09e-003	2	5.19e-001	5	5.10e-002	3
DTLZ22	9.29e-003	4	5.92e-002	5	2.36e-003	3	1.89e-003	2	1.20e-003	1
DTLZ32	5.32e-001	5	2.66e-003	2	2.88e-003	3	1.81e-003	1	4.15e-002	4
DTLZ42	1.42e-003	3	3.72e-003	4	1.67e-001	5	1.35e-003	2	5.51e-004	1
DTLZ52	1.82e-002	4	1.50e-001	5	6.38e-003	3	1.89e-003	1	2.24e-003	2
DTLZ62	1.84e+001	5	4.45e-002	2	7.94e-002	3	1.32e+001	4	3.91e-004	1
DTLZ72	1.05e-001	4	7.73e-002	3	1.41e-003	2	9.82e-001	5	4.26e-004	1
DTLZ82	1.63e-003	2	1.99e-003	3	1.57e-003	1	2.40e-003	5	2.03e-003	4
DTLZ92	1.37e-003	1	2.02e-003	2	8.79e-003	5	4.53e-003	4	2.42e-003	3
DTLZ13	1.68e+000	5	1.20e-002	1	1.79e-001	3	6.85e-002	2	4.82e-001	4
DTLZ23	7.49e-002	3	1.12e-001	5	3.47e-002	2	9.74e-002	4	2.43e-002	1
DTLZ33	1.29e+000	5	1.08e-002	1	5.03e-002	2	8.39e-001	4	4.30e-001	3
DTLZ43	1.81e-002	2	2.11e-002	3	1.39e-001	5	3.19e-002	4	1.36e-002	1
DTLZ53	7.08e-002	4	2.60e-001	5	5.83e-003	2	4.79e-003	1	3.12e-002	3
DTLZ63	4.37e+001	5	2.11e+000	3	1.94e-001	1	3.07e+001	4	2.13e-001	2
DTLZ73	3.59e-001	4	9.67e-002	3	1.89e-002	1	1.11e+000	5	2.80e-002	2
DTLZ83	6.13e-002	2	6.27e-002	3	1.01e-002	1	3.04e-001	5	1.58e-001	4
DTLZ93	4.98e-002	2	6.92e-002	3	3.03e-002	1	1.04e-001	4	1.05e-001	5
Total		85		69		57		82		52

Tableau 3.2 : Résultats de la métrique  $H$ .

	<b>NSGAI</b>		<b>GDE3</b>		<b>AMOS</b>		<b>SMPSO</b>		<b>MODE-2A</b>	
ZDT1	9.19e-001	4	9.23e-001	3	9.57e-001	2	8.73e-001	5	9.62e-001	1
ZDT2	7.37e-001	5	8.27e-001	3	9.20e-001	2	7.45e-001	4	9.25e-001	1
ZDT3	9.12e-001	3	9.05e-001	4	9.17e-001	2	8.23e-001	5	9.46e-001	1
ZDT4	6.43e-002	4	0.00e+000	5	9.33e-001	2	9.47e-001	1	8.70e-001	3
ZDT6	0.00e+000	4	0.00e+000	5	8.91e-001	2	6.13e-002	3	9.21e-001	1
DTLZ12	5.47e-001	5	7.74e-001	1	7.72e-001	2	7.58e-001	3	7.37e-001	4
DTLZ22	6.43e-001	4	6.06e-001	5	6.44e-001	3	6.48e-001	2	6.48e-001	1
DTLZ32	3.59e-001	5	6.47e-001	2	6.47e-001	3	6.48e-001	1	6.21e-001	4
DTLZ42	5.53e-001	4	6.47e-001	2	4.40e-001	5	6.48e-001	1	6.07e-001	3
DTLZ52	6.36e-001	3	5.36e-001	5	6.34e-001	4	6.48e-001	1	6.47e-001	2
DTLZ62	0.00e+000	4	6.07e-001	3	6.33e-001	2	0.00e+000	5	6.49e-001	1
DTLZ72	5.66e-001	4	5.96e-001	3	6.43e-001	2	1.19e-001	5	6.75e-001	1
DTLZ82	7.97e-001	4	8.05e-001	3	7.16e-001	5	8.55e-001	2	8.60e-001	1
DTLZ92	3.89e-001	4	4.85e-001	3	3.47e-001	5	5.44e-001	2	6.28e-001	1
DTLZ13	3.38e-001	5	9.30e-001	2	9.34e-001	1	9.11e-001	3	7.52e-001	4
DTLZ23	7.76e-001	3	7.58e-001	5	7.95e-001	2	7.68e-001	4	8.16e-001	1
DTLZ33	2.91e-001	5	8.15e-001	2	8.18e-001	1	7.60e-001	3	5.80e-001	4
DTLZ43	7.28e-001	4	8.09e-001	2	5.89e-001	5	7.91e-001	3	8.21e-001	1
DTLZ53	4.67e-001	4	3.85e-001	5	4.87e-001	2	4.95e-001	1	4.86e-001	3
DTLZ63	0.00e+000	4	1.15e-001	3	4.78e-001	1	0.00e+000	5	4.29e-001	2
DTLZ73	3.48e-001	4	5.92e-001	3	6.28e-001	2	6.58e-002	5	6.67e-001	1
DTLZ83	7.50e-001	4	7.65e-001	3	6.11e-001	5	7.74e-001	2	8.32e-001	1
DTLZ93	2.44e-001	4	3.37e-001	3	2.15e-001	5	3.65e-001	2	4.72e-001	1
Total		95		68		70		69		43

Tableau 3.3 : Résultats de la métrique  $S$ .

	<b>NSGAI</b>		<b>GDE3</b>		<b>AMOS</b>		<b>SMPSO</b>		<b>MODE-2A</b>	
ZDT1	1.06e-002	2	2.00e-002	5	1.51e-002	3	1.57e-002	4	2.96e-003	1
ZDT2	3.40e+000	4	4.83e-002	3	1.60e-002	2	1.34e+001	5	2.67e-003	1
ZDT3	6.67e-003	2	2.11e-002	3	4.02e-002	5	3.17e-002	4	3.19e-003	1
ZDT4	3.55e+000	4	3.34e+001	5	1.71e-002	1	2.23e-002	3	2.10e-002	2
ZDT6	1.44e-001	4	1.81e-001	5	6.48e-002	2	7.96e-002	3	1.11e-002	1
DTLZ12	2.39e-002	5	2.90e-003	1	3.19e-003	2	1.76e-002	4	5.32e-003	3
DTLZ22	6.96e-003	3	2.61e-002	5	1.32e-002	4	4.68e-003	2	3.33e-003	1
DTLZ32	3.58e-002	5	9.85e-003	4	4.06e-003	1	8.46e-003	3	5.38e-003	2
DTLZ42	1.01e+001	3	8.53e-003	2	3.69e+001	5	3.91e-003	1	1.33e+001	4
DTLZ52	8.00e-003	3	4.22e-002	5	2.25e-002	4	4.66e-003	2	4.20e-003	1
DTLZ62	2.88e-002	2	6.48e-002	4	3.12e-002	3	1.00e+001	5	2.85e-003	1
DTLZ72	9.35e-003	2	1.93e-002	3	2.25e-002	4	2.38e-002	5	2.29e-003	1
DTLZ82	5.31e-003	3	1.01e-002	4	2.65e-002	5	4.52e-003	2	3.24e-003	1
DTLZ92	5.23e-003	1	5.52e-003	2	2.34e+001	5	1.09e-002	4	6.50e-003	3
DTLZ13	3.92e-002	4	2.42e-002	2	2.85e-002	3	4.01e-002	5	1.86e-002	1
DTLZ23	3.81e-002	4	3.66e-002	2	4.05e-002	5	3.79e-002	3	1.41e-002	1
DTLZ33	4.94e-002	4	3.64e-002	3	2.97e-002	2	8.51e-002	5	2.75e-002	1
DTLZ43	3.41e+000	4	3.60e-002	2	3.64e+000	5	4.51e-002	3	1.43e-002	1
DTLZ53	1.37e-002	3	2.16e-002	5	1.66e-002	4	7.07e-003	1	1.09e-002	2
DTLZ63	3.75e-002	4	1.20e-001	5	2.18e-002	2	3.64e-002	3	1.13e-002	1
DTLZ73	2.86e-002	2	2.89e-002	3	3.27e-002	4	4.96e-002	5	1.20e-002	1
DTLZ83	3.47e-002	3	3.39e-002	2	3.92e-002	5	3.62e-002	4	1.82e-002	1
DTLZ93	2.86e-002	2	3.33e-002	3	3.38e+000	5	3.39e-002	4	1.51e-002	1
Total		77		70		86		78		34

Tableau 3.4 : Résultats de la métrique  $\bar{d}_{\max}$ .

	<b>NSGAI</b>		<b>GDE3</b>		<b>AMOS</b>		<b>SMPSO</b>		<b>MODE-2A</b>	
ZDT1	1.36e-001	3	1.37e-001	4	1.51e-002	2	4.65e-001	5	7.88e-003	1
ZDT2	1.17e+000	4	2.99e-001	3	1.80e-002	2	1.20e+000	5	7.85e-003	1
ZDT3	6.43e-002	3	9.56e-002	4	3.01e-002	2	5.01e-001	5	6.16e-003	1
ZDT4	5.36e+001	4	3.41e+002	5	9.76e-002	2	5.18e-002	1	4.46e-001	3
ZDT6	4.25e+001	3	4.82e+001	4	1.36e-001	2	6.57e+001	5	5.78e-002	1
DTLZ12	1.24e+000	5	8.15e-003	1	9.66e-003	2	5.25e-001	4	1.27e-001	3
DTLZ22	1.41e-002	4	7.43e-002	5	1.34e-002	3	8.62e-003	2	8.47e-003	1
DTLZ32	1.85e+000	5	1.04e-002	3	9.30e-003	1	9.84e-003	2	1.18e-001	4
DTLZ42	1.03e+000	4	1.01e-002	2	2.04e+000	5	8.36e-003	1	4.62e-001	3
DTLZ52	2.28e-002	4	2.05e-001	5	2.26e-002	3	8.60e-003	1	8.82e-003	2
DTLZ62	6.86e+002	5	2.20e-001	3	9.61e-002	2	3.51e+002	4	8.30e-003	1
DTLZ72	1.27e-001	4	9.99e-002	3	1.76e-002	2	4.15e+000	5	8.77e-003	1
DTLZ82	2.11e-002	4	1.96e-002	3	3.76e-002	5	1.15e-002	2	1.02e-002	1
DTLZ92	2.97e-002	4	2.13e-002	2	6.05e-001	5	2.17e-002	3	1.09e-002	1
DTLZ13	1.69e+001	5	8.43e-002	1	1.28e+000	3	2.13e-001	2	4.24e+000	4
DTLZ23	1.91e-001	2	2.29e-001	3	2.57e-001	4	2.61e-001	5	1.11e-001	1
DTLZ33	1.21e+001	5	1.20e-001	1	4.20e-001	2	5.58e+000	4	2.25e+000	3
DTLZ43	1.03e+000	4	1.26e-001	2	2.93e+000	5	3.87e-001	3	1.07e-001	1
DTLZ53	9.44e-002	4	4.64e-001	5	2.23e-002	2	1.33e-002	1	4.17e-002	3
DTLZ63	4.66e+003	5	1.53e+001	3	1.61e-001	1	2.11e+003	4	1.13e+000	2
DTLZ73	1.12e+000	4	2.91e-001	3	1.89e-001	2	6.88e+000	5	1.83e-001	1
DTLZ83	1.50e-001	3	1.30e-001	2	1.90e-001	4	3.76e-001	5	1.25e-001	1
DTLZ93	2.43e-001	4	1.58e-001	2	3.44e-001	5	2.02e-001	3	7.91e-002	1
Total		96		62		70		76		41

Enfin, pour la métrique  $\bar{d}_{\max}$ , le classement est le suivant : 1-MODE-2A, 2-GDE3, 3-AMOSA, 4-SMPSO, 5-NSGA-II. Ce résultat montre que les meilleures performances combinées de convergence, de distribution et de répartition des solutions, sont obtenues en utilisant l'algorithme MODE-2A. Nous remarquons aussi que les résultats de cette métrique sont semblables à ceux de la métrique  $H$ , ce qui prouve que la métrique  $\bar{d}_{\max}$  est capable de fournir des mesures fiables et robustes vis-à-vis des différents aspects de la qualité des solutions non dominées.

Afin d'illustrer graphiquement les performances des cinq algorithmes utilisés, nous montrons dans les figures 3.6 à 3.11 les fronts non dominés des problèmes ZDT, correspondant aux valeurs médianes des 30 mesures de la métrique  $\bar{d}_{\max}$  calculées pour chaque algorithme. Nous remarquons que les meilleurs fronts, en termes de convergence et de distribution des solutions, sont ceux de l'algorithme MODE-2A à l'exception du problème ZDT4. Nous constatons également que les deuxièmes meilleurs fronts sont ceux de l'algorithme AMOSA, ce qui reflète bien les résultats du tableau 3.4.

Par ailleurs, les meilleurs fronts des problèmes DTLZ obtenus avec MODE-2A, d'après la métrique  $H$ , sont illustrés sur les figures 3.12 à 3.20. Nous remarquons que les fronts obtenus pour les problèmes à trois objectifs ne représentent pas fidèlement les fronts réels, tandis que les fronts des problèmes à deux objectifs approchent les fronts de Pareto d'une manière plus précise.

## 3.4 Conclusion

Dans ce chapitre, un nouvel algorithme d'optimisation multiobjectif à évolution différentielle, nommé MODE-2A, est proposé et comparé avec les algorithmes de référence NSGA-II, GDE3, AMOSA et SMPSO. Les résultats de cette comparaison montrent que l'algorithme MODE-2A est capable de résoudre différents types de problèmes académiques d'une manière plus efficace que les autres algorithmes. Les performances de ces algorithmes sont comparées en termes de convergence et de diversité des solutions non dominées obtenues à la fin de la recherche. Dans le prochain chapitre, nous proposons l'application de toutes ces méthodes sur des problèmes d'optimisation réels qui concernent le dimensionnement des PSS dans les réseaux électriques.

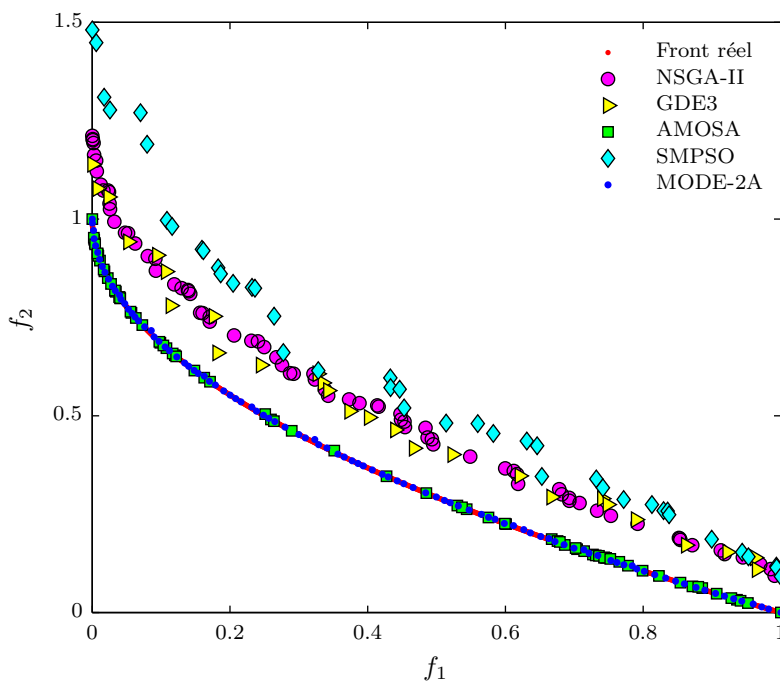


Figure 3.6 : Comparaison des résultats du problème ZDT1.

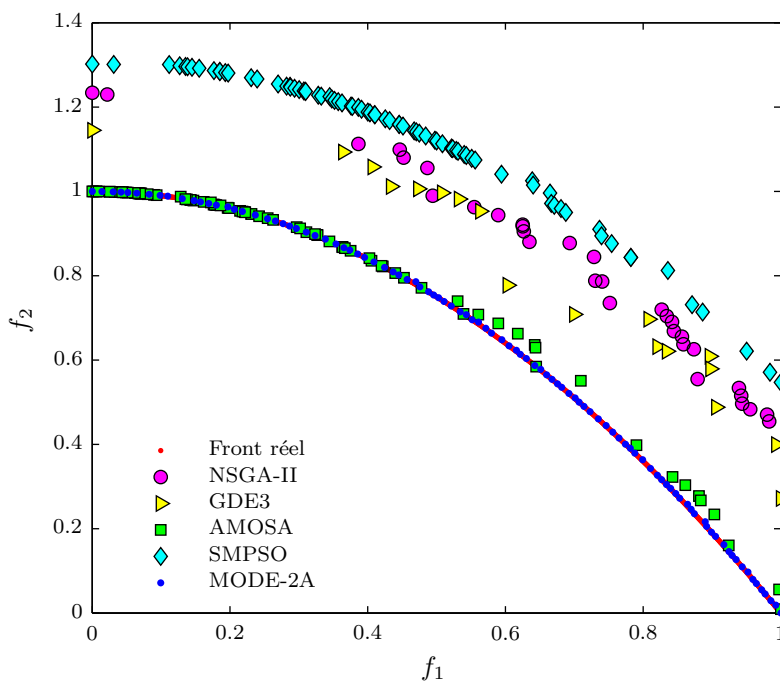


Figure 3.7 : Comparaison des résultats du problème ZDT2.

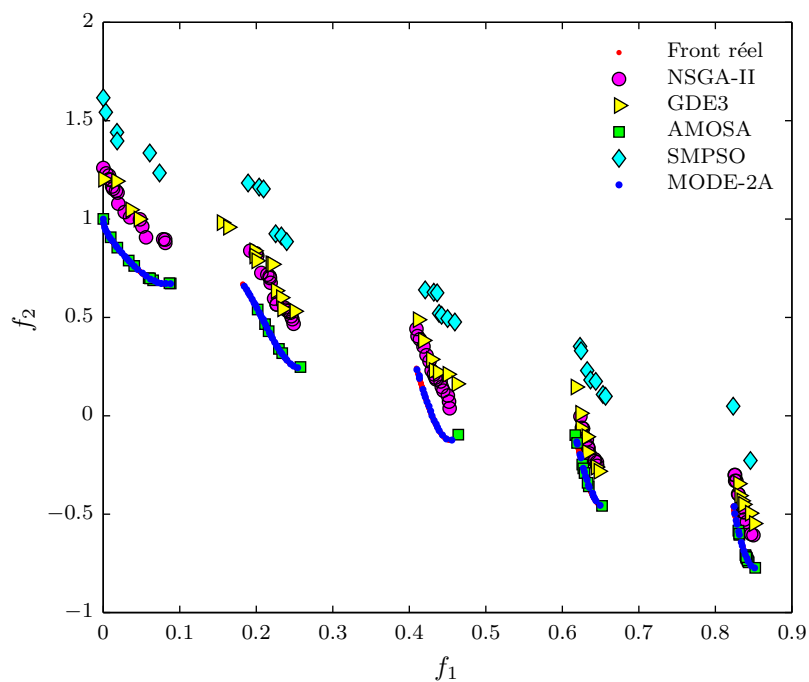


Figure 3.8 : Comparaison des résultats du problème ZDT3.

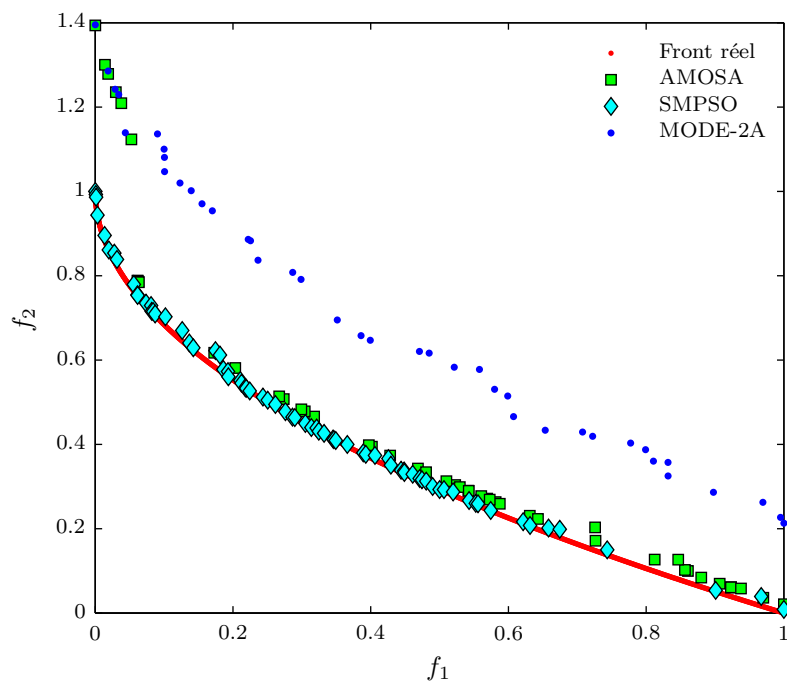


Figure 3.9 : Comparaison des résultats du problème ZDT4.

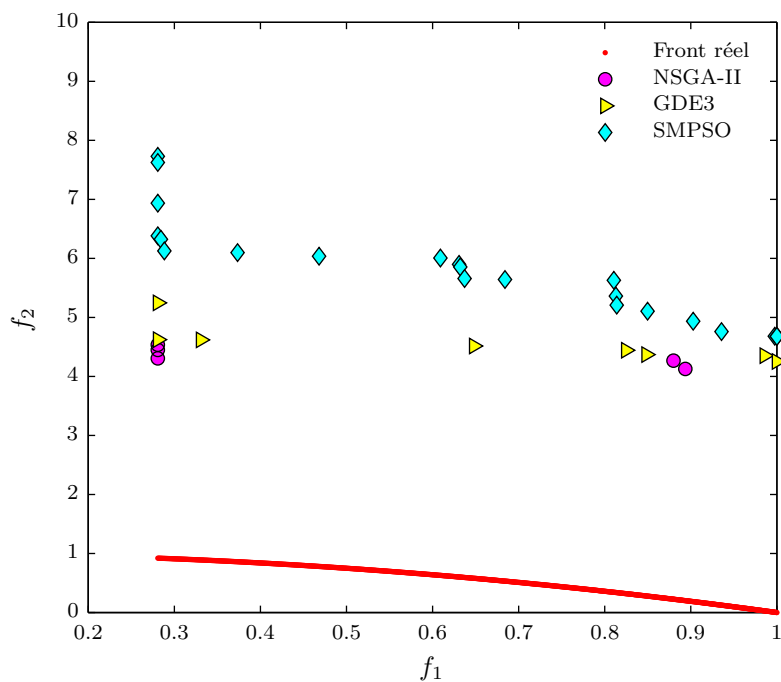


Figure 3.10 : Comparaison des résultats du problème ZDT6 (a).

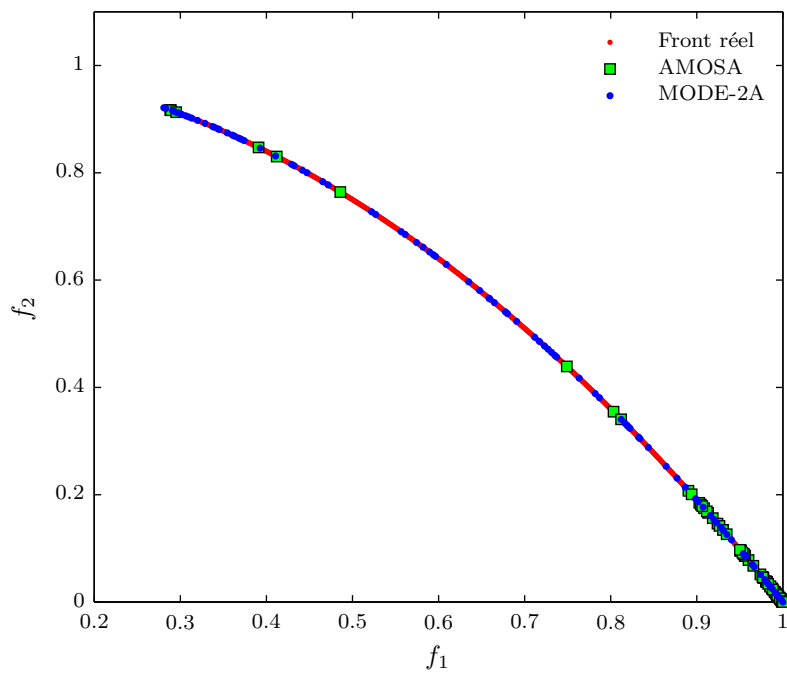


Figure 3.11 : Comparaison des résultats du problème ZDT6 (b).

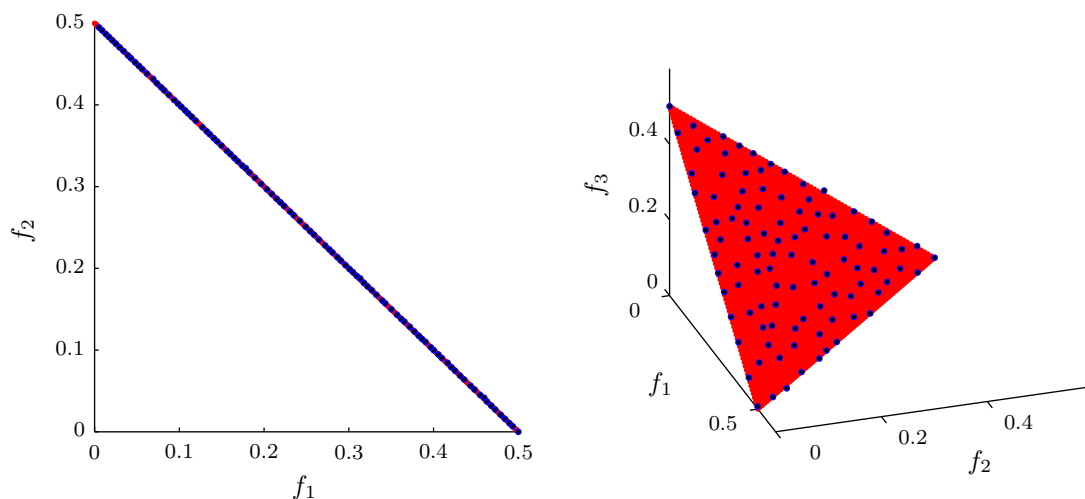


Figure 3.12 : Fronts non dominés du problème DTLZ1.

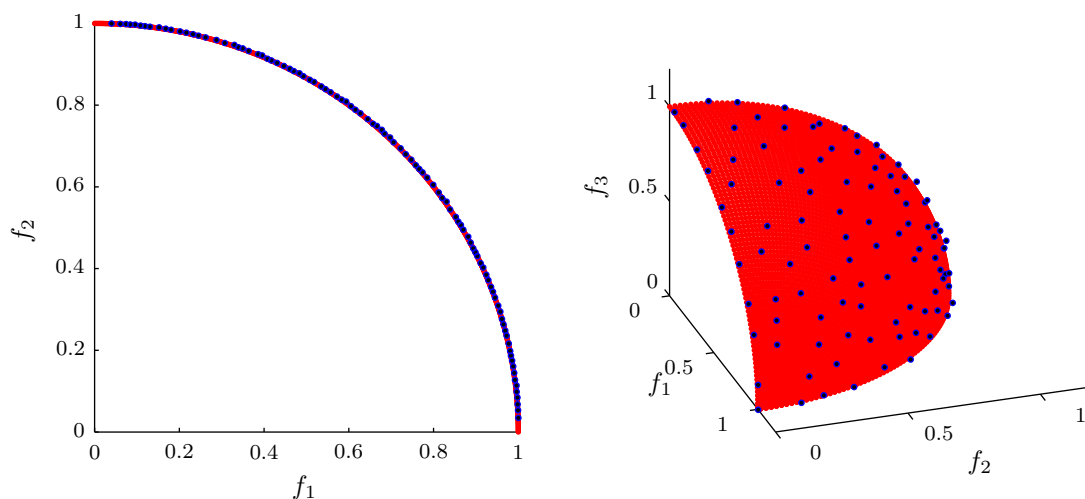


Figure 3.13 : Fronts non dominés du problème DTLZ2.

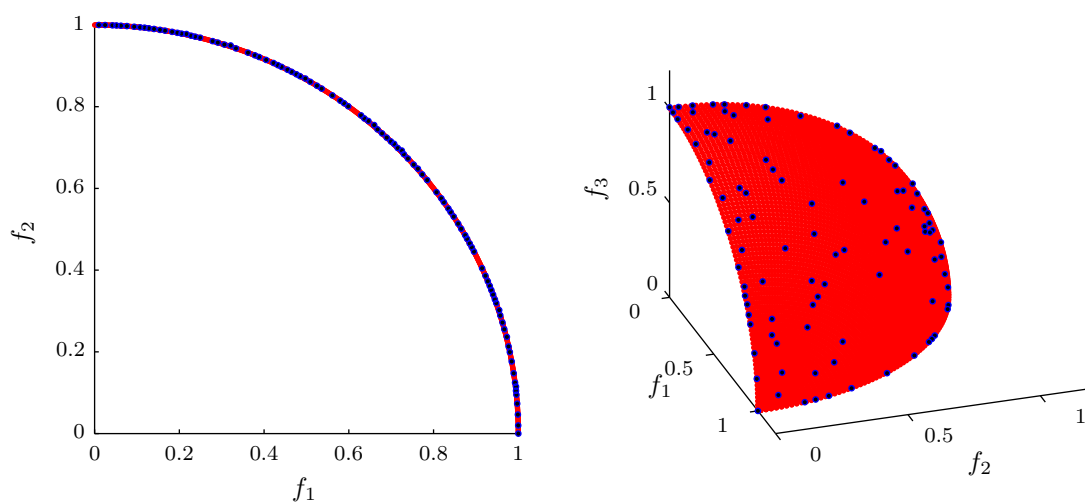


Figure 3.14 : Fronts non dominés du problème DTLZ3.



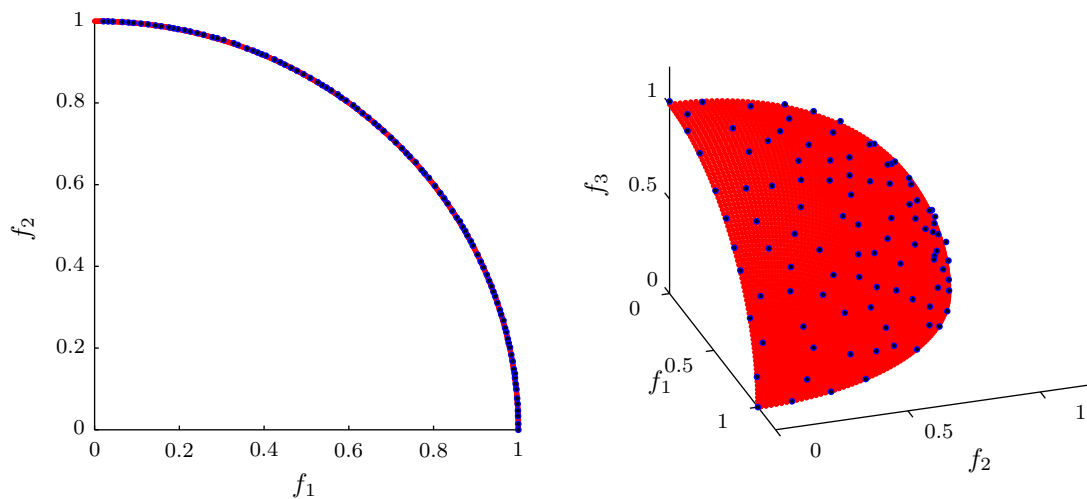


Figure 3.15 : Fronts non dominés du problème DTLZ4.

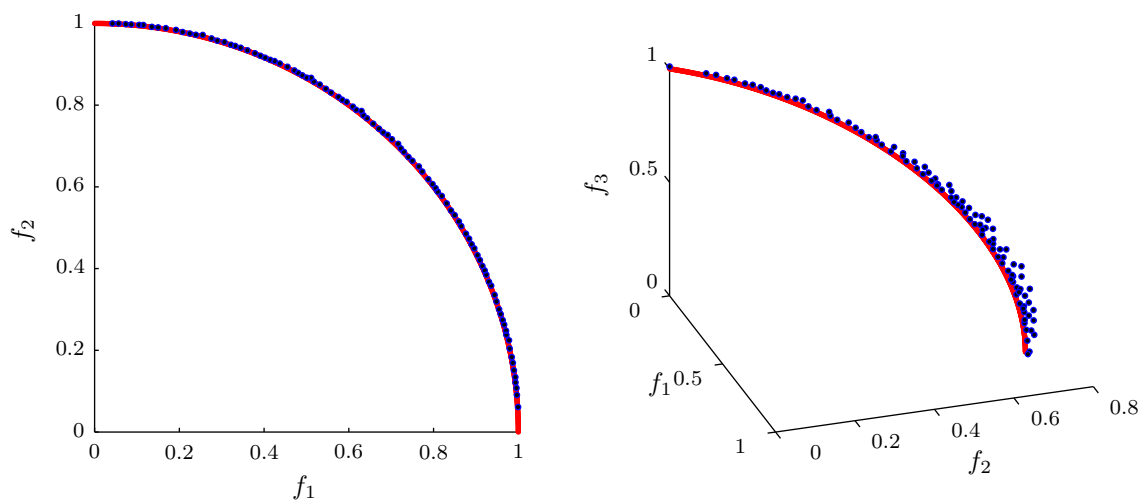


Figure 3.16 : Fronts non dominés du problème DTLZ5.

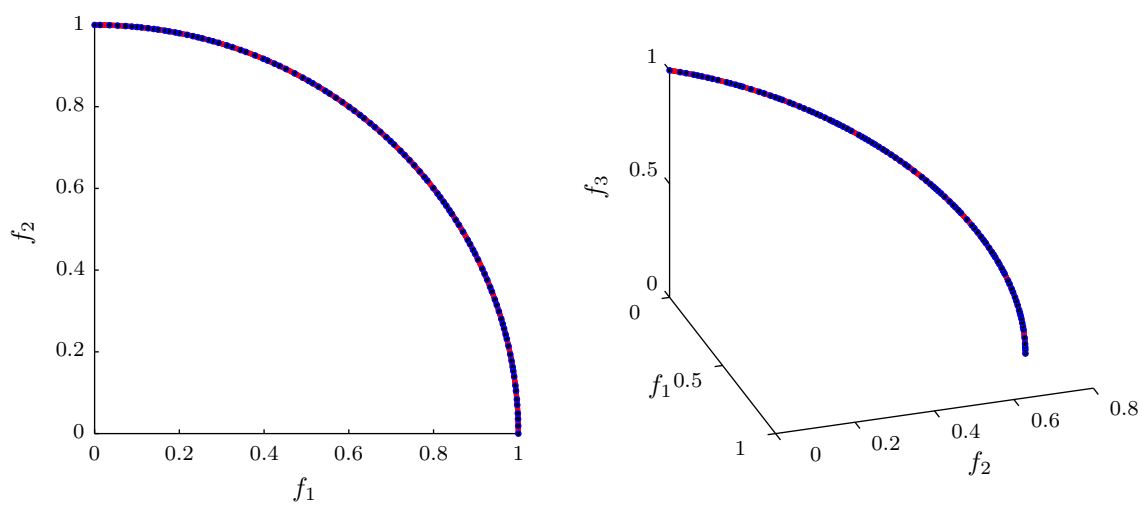


Figure 3.17 : Fronts non dominés du problème DTLZ6.

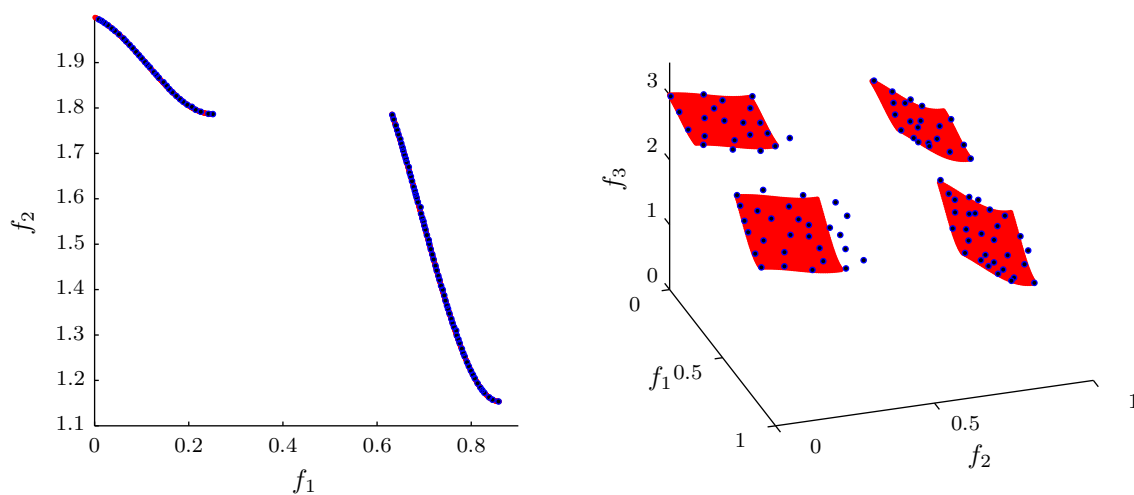


Figure 3.18 : Fronts non dominés du problème DTLZ7.

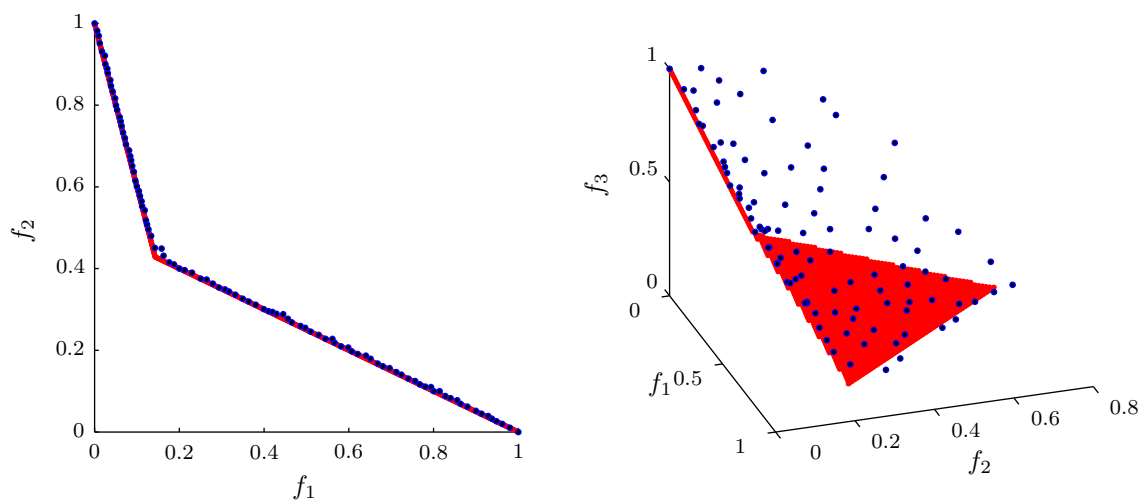


Figure 3.19 : Fronts non dominés du problème DTLZ8.

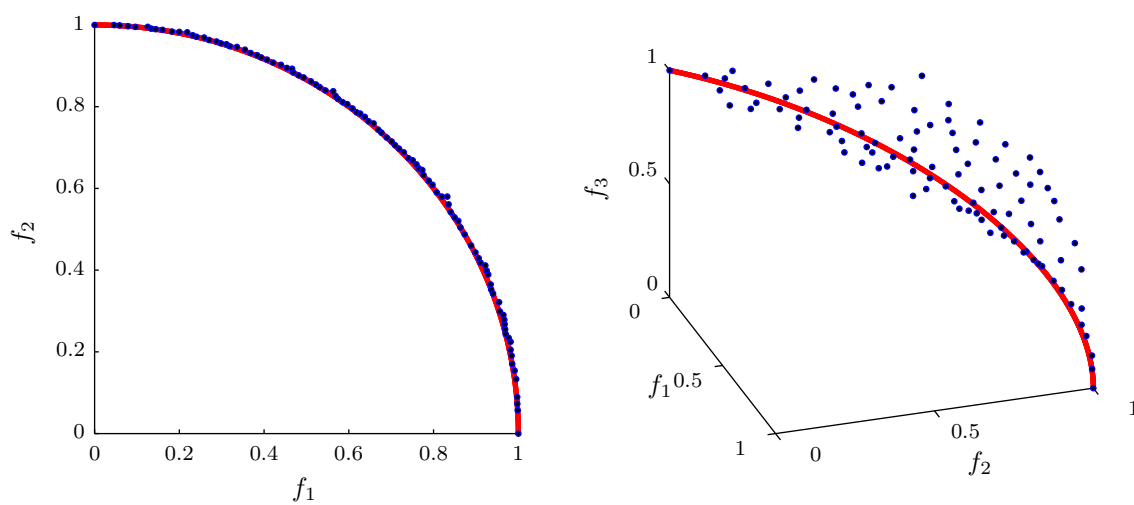


Figure 3.20 : Fronts non dominés du problème DTLZ9.

## Chapitre 4

# Applications aux problèmes de synthèse des PSS dans les réseaux électriques

## Chapitre 4

# Applications aux problèmes de synthèse des PSS dans les réseaux électriques

### 4.1 Introduction

Dans ce chapitre, nous proposons de nouvelles approches pour la résolution de plusieurs problèmes de dimensionnement optimal des PSS dans les réseaux électriques. Pour cela, nous utilisons les différents algorithmes d'optimisation multiobjectifs étudiés dans les chapitres précédents. Les problèmes considérés sont formulés sous forme de problèmes de minimisation à deux et à trois fonctions objectifs, en considérant plusieurs exemples de réseaux électriques. Des tests d'optimisation et de simulation sont effectués afin d'illustrer l'efficacité des approches proposées.

### 4.2 Application à un réseau mono-machine

Le premier réseau électrique étudié est illustré sur la figure 4.1 [79]. Ce réseau est constitué d'un seul générateur connecté à un nœud infini par l'intermédiaire d'une ligne de transport. Le générateur est équipé d'un régulateur de tension (AVR), et d'un stabilisateur de puissance (PSS). Les équations différentielles qui régissent la dynamique du générateur sont données dans l'annexe A, tandis que les paramètres de ce réseau sont donnés dans l'annexe B.

#### 4.2.1 Formulation du problème

Le problème considéré pour le réseau étudié est celui d'optimiser la robustesse et la rapidité de convergence des réponses du système en recherchant les paramètres optimaux du PSS. Ce problème peut être formulé mathématiquement comme suit [80, 81] :

$$\min F(\mathbf{x}) = \begin{pmatrix} \max_i \|T_i\|_\infty \\ \max_i \|T_i\|_2 \end{pmatrix}, \quad i = 1, \dots, 8 \quad (4.1)$$

où  $\mathbf{x}$  représente le vecteur des variables de décision du problème, c'est-à-dire les paramètres du PSS, et  $T$  la fonction de transfert du système en boucle fermée représentée sous la forme

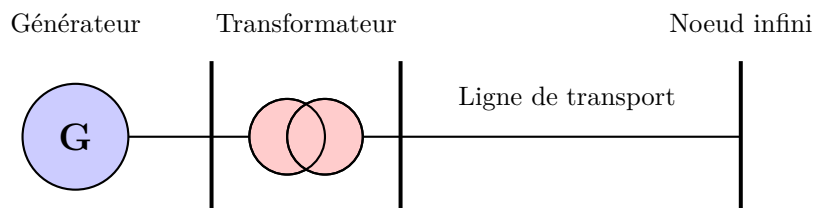


Figure 4.1 : Réseau mono-machine.

zéros-pôles-gain. La fonction  $T$  est calculée en linéarisant le système autour de 8 points de fonctionnement en considérant les valeurs limites (0.4, 1.2) p.u pour la puissance active du générateur, (0.1, 0.5) p.u pour la puissance réactive et (0.1, 0.7) p.u pour la réactance de la ligne de transport. Les limites de recherche des paramètres du PSS sont données dans le tableau 4.1.

Les normes  $H_\infty$  et  $H_2$  sont utilisées pour mesurer la robustesse et la rapidité de convergence du système en boucle fermée vis-à-vis de certaines variables d'entrée et d'autres variables de sortie [82]. Ces normes sont utilisées notamment pour le dimensionnement des PSS [83].

Tableau 4.1 : Espace de recherche pour le réseau mono-machine.

Paramètres	Min	Max
Zéros	-100	0
Gain	0.1	$10^6$

## 4.2.2 Résultats de l'optimisation

Le problème défini dans l'équation 4.1 est résolu en utilisant les algorithmes NSGA-II, GDE3, AMOSA, SMPSO et MODE-2A. Ces derniers sont appliqués de la même façon que dans le chapitre précédent, sauf que le nombre des solutions finales est dans ce cas fixé à 50 et que le nombre total d'évaluations des fonctions objectifs est 750. Par ailleurs, seuls l'hypervolume et la métrique d'espacement sont utilisés ici comme métriques de performances étant donné que le front de Pareto est inconnu. Ces métriques sont calculées en exécutant 30 fois chacun des algorithmes utilisés en considérant les mêmes ensembles de solutions initiales. Par ailleurs, afin de prendre en considération les performances des algorithmes durant le processus de recherche, nous prenons comme valeur finale des métriques de performance la somme des mesures de  $H$  et  $S$  après cinq étapes de recherche.

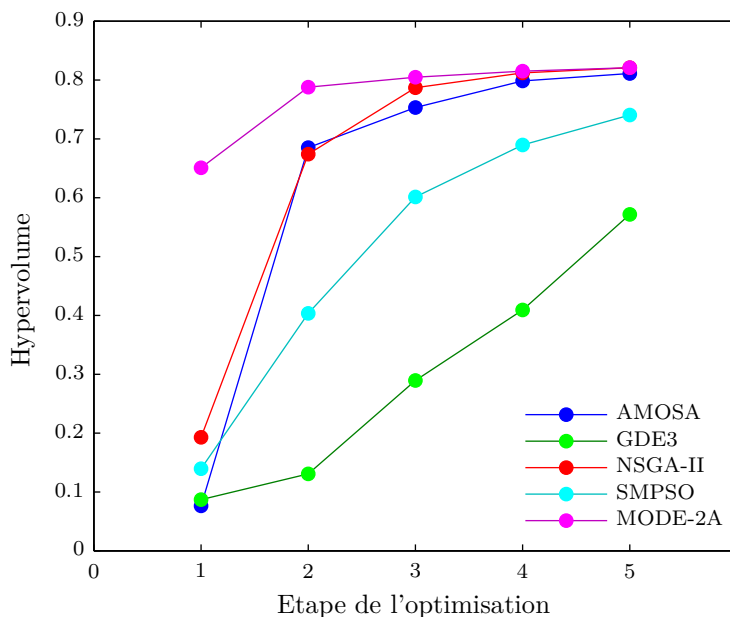


Figure 4.2 : Variation de l'hypervolume.

Les mesures finales des métriques  $H$  et  $S$  sont données dans le tableau 4.2. Nous constatons que les meilleurs résultats pour ces deux métriques sont obtenus avec l'algorithme MODE-2A, ce qui prouve que ce dernier est capable de fournir des solutions plus efficaces et plus diversifiées que les autres algorithmes. Nous remarquons également que les deuxièmes meilleures performances sont celles de NSGA-II et AMOSA, respectivement, tandis que l'algorithme GDE3 est classé dernier.

Tableau 4.2 : Résultats des métriques pour le réseau mono-machine.

	NSGAI	GDE3	AMOSA	SMPSO	MODE-2A
H	3.29e+000	2	1.49e+000	5	3.12e+000
S	1.14e+002	3	3.67e+002	5	1.04e+002

Afin d'illustrer les résultats précédents, nous montrons dans la figure 4.2 les variations des mesures de l'hypervolume en fonction des étapes de l'optimisation. Les courbes obtenues montrent que l'algorithme MODE-2A converge plus rapidement vers le front final, et ce dès la première étape de l'optimisation. Nous remarquons également que les deux algorithmes MODE-2A, NSGA-II convergent pratiquement vers le même front au stade final de l'optimisation.

Le front non dominé final du problème traité est montré sur la figure 4.3. Ce front est établi en sélectionnant les 50 solutions les mieux réparties parmi toutes les solutions non dominées trouvées par les cinq algorithmes utilisés. Le front obtenu est convexe, et illustre clairement le compromis qui existe entre les deux critères du problème, c'est-à-dire entre la robustesse et la rapidité de convergence du système.

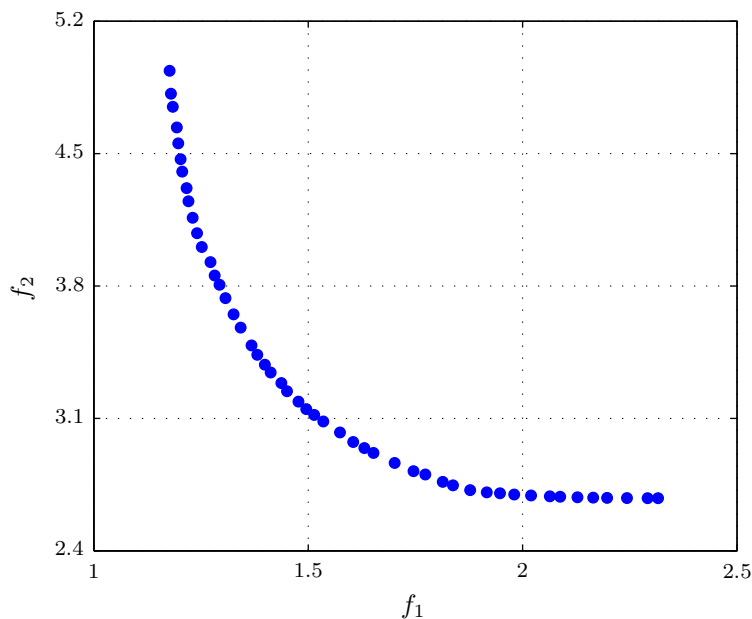


Figure 4.3 : Front non dominé.

### 4.2.3 Résultats de simulation

Dans le but d'examiner les performances des solutions de compromis, nous présentons dans ce qui suit quelques résultats de simulation obtenus en utilisant le modèle non linéaire du réseau et en considérant le point de fonctionnement nominal. Dans ce cas, la sortie du système est la variation de la vitesse rotorique du générateur par rapport à la valeur de synchronisme. Le scénario considéré est tel que la valeur de la tension de référence du générateur est diminuée de 0.05 p.u à  $t = 0.1$  s.

Pour des raisons de simplification, seuls les deux solutions correspondant aux points extrêmes du front non dominé de la figure 4.3 sont utilisées dans les simulations. Ces deux solutions sont celles qui permettent d'obtenir les meilleures performances vis-à-vis des deux fonctions objectifs du problème.

Les résultats de simulation sont montrés dans les figures 4.4 à 4.6. Nous constatons que la réponse du système présente un dépassement plus faible en utilisant la première solution, c'est-à-dire celle qui minimise la première fonction objectif. Par contre, la réponse la plus rapide est obtenue avec la deuxième solution. Ce résultat montre que les performances du système ne peuvent être améliorées simultanément en termes de robustesse et de rapidité de convergence.

D'autre part, quand on utilise la première solution, l'amplitude du signal de commande (tension du PSS) augmente d'une façon rapide, tandis qu'avec la deuxième solution le signal de commande converge plus rapidement. Néanmoins, dans ce cas particulier, la variation de la tension d'excitation est plus faible avec la première solution étant donné qu'elle est compensée avec la tension de sortie du PSS.

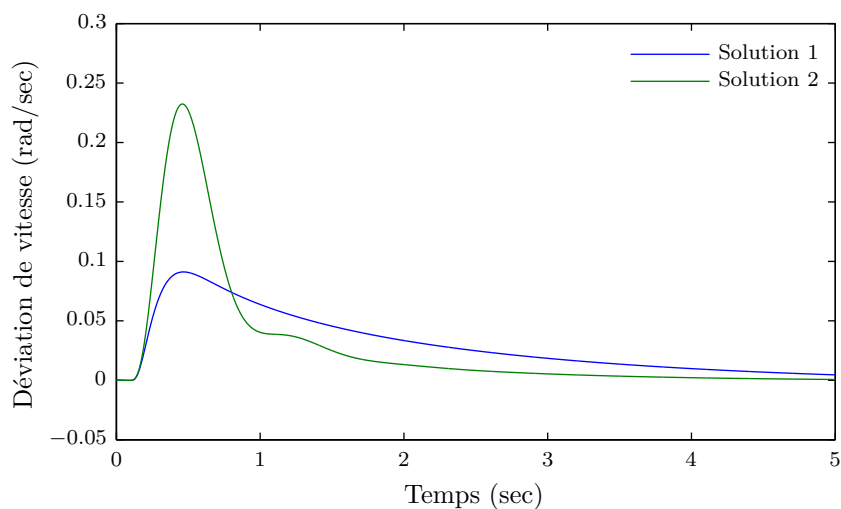


Figure 4.4 : Réponses du système.

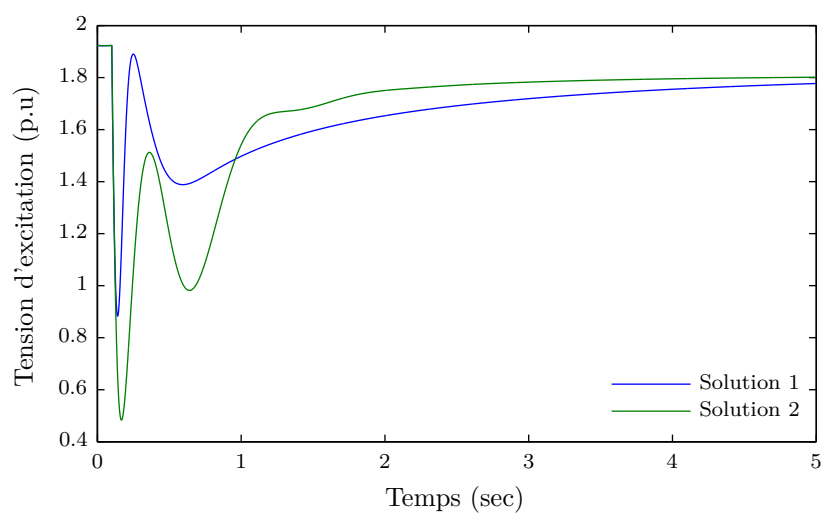


Figure 4.5 : Tensions d'excitation.

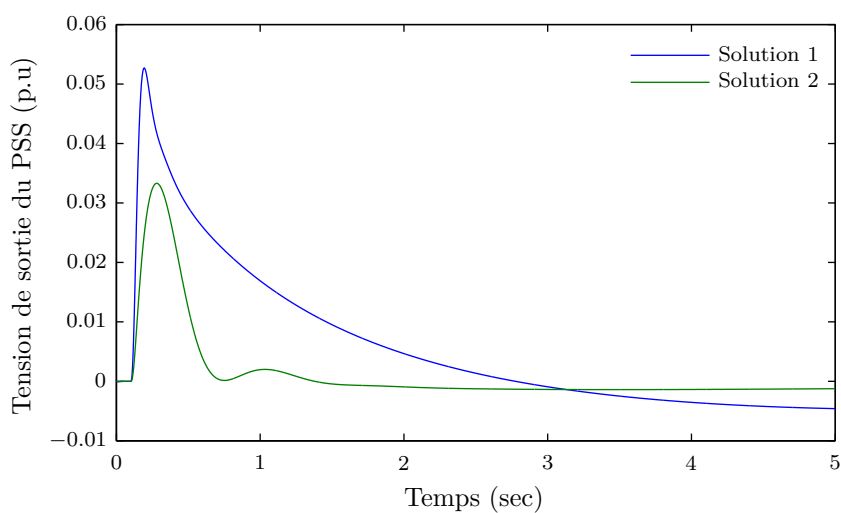


Figure 4.6 : Tensions de commande.



#### 4.2.4 Sélection d'une solution de compromis

Pour sélectionner une solution de compromis, nous effectuons des simulations dans le but de comparer les performances de toutes les solutions du front non dominé de la figure 4.3. Ainsi, deux autres scénarios sont considérés à cet effet, le deuxième scénario correspond à des augmentations simultanées de 0.05 p.u de la tension de référence du générateur et de 0.2 p.u de la réactance de la ligne de transport. Le troisième scénario est un court-circuit d'une durée de 0.05 s survenu au niveau du bus terminal du générateur. Le point de fonctionnement considéré est le point nominal.

Pour évaluer les performances des PSS, nous utilisons deux indicateurs de performance temporelle. Le premier est l'indicateur dit Integral of Time-weighted Absolute Error (ITAE) et le deuxième est le dépassement maximal de la réponse du système. Ces deux indicateurs sont évalués en réalisant des simulations jusqu'à l'instant  $t = 20$  s et en utilisant le modèle non linéaire du réseau. L'indicateur ITAE est utilisé pour mesurer la rapidité de convergence des réponses, tandis que le dépassement permet d'évaluer la robustesse du système.

Les résultats des indicateurs de performance sont montrés dans les figures 4.7 à 4.9, dans lesquelles les solutions sont classées selon l'ordre croissant de la première fonction objectif, c'est-à-dire selon les performances de robustesse. Nous constatons que pour les trois scénarios, l'indicateur ITAE augmente lorsque le dépassement diminue, c'est-à-dire quand le système devient plus robuste. Ces résultats reflètent clairement le compromis qui existe entre les deux fonctions objectifs du problème.

Pour sélectionner une solution de compromis, nous devons choisir entre des systèmes robustes et des systèmes rapides. Toutefois, étant donné que plusieurs points de fonctionnement ont été considérés pour évaluer les performances des PSS, la robustesse des systèmes vis-à-vis du changement du point de fonctionnement est donc assurée. De ce fait, notre choix se porte sur la solution qui minimise la deuxième fonction objectif, c'est-à-dire celle qui permet d'avoir la meilleure rapidité de convergence.

Les performances du PSS correspondant à la solution sélectionnée sont comparées avec celles d'un PSS conventionnel. Ce dernier est dimensionné en utilisant une approche déterministe, qui consiste à rechercher les paramètres du PSS permettant de maximiser l'amortissement du système en boucle fermée linéarisé autour du point de fonctionnement nominal. Les paramètres de ces deux PSS sont donnés dans le tableau 4.3.

Tableau 4.3 : Paramètres des PSS sélectionnés pour le réseau mono-machine.

PSS	$K$	$T_1$	$T_2$	$T_3$	$T_4$
Conventionnel	17	0.17	0.05	0.17	0.05
Proposé	38.58	0.10	0.01	0.10	0.01

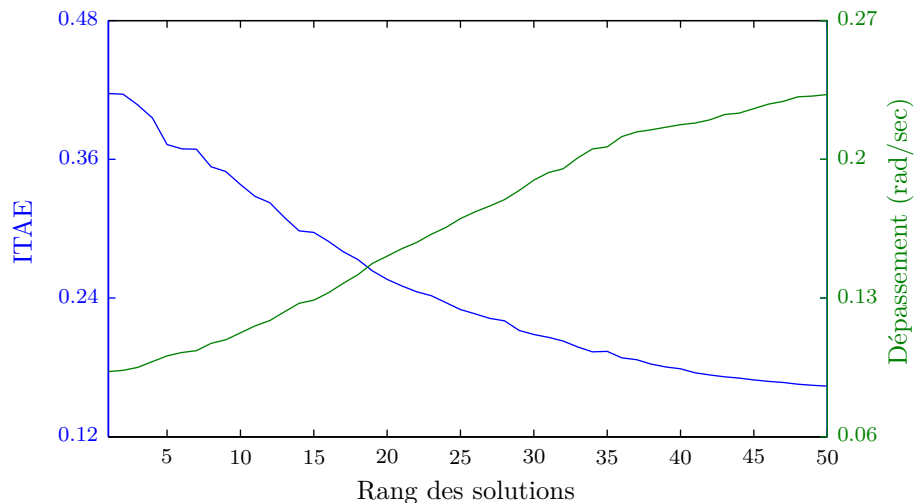


Figure 4.7 : Comparaison des performances des PSS (scénario n°1).

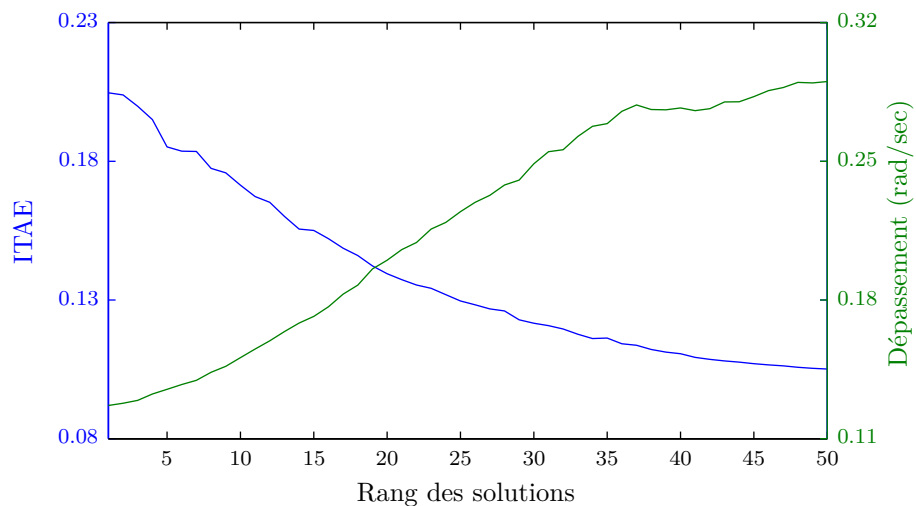


Figure 4.8 : Comparaison des performances des PSS (scénario n°2).

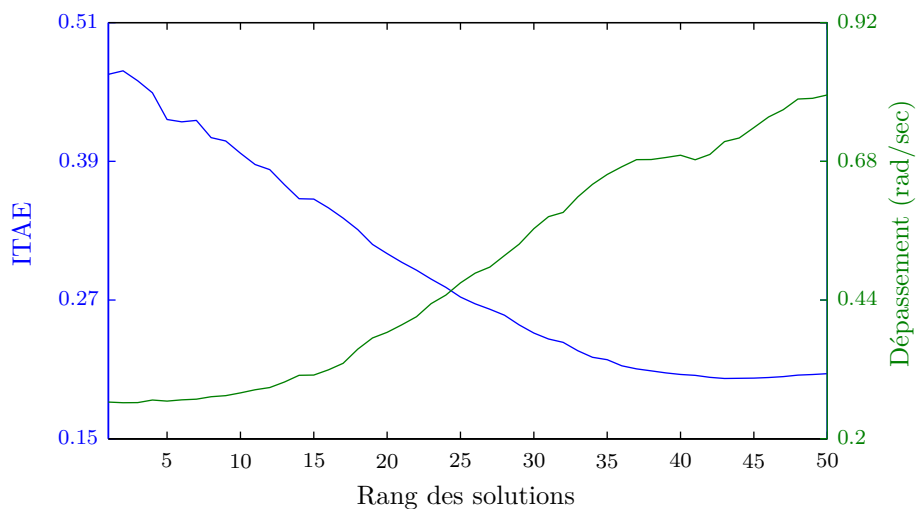


Figure 4.9 : Comparaison des performances des PSS (scénario n°3).

Plusieurs simulations sont effectuées afin de comparer le PSS proposé et le PSS conventionnel. Pour cela, nous considérons les trois scénarios précédents ainsi que trois différents points de fonctionnement qui dépendent de la charge du réseau (nominale, forte et faible). Nous utilisons également deux autres indicateurs de performance temporelle : l'indicateur Integral of Time Multiply Squared Error (ITSE) et l'indicateur Integral of Square Error (ISE).

Le premier indicateur, ITSE, mesure la rapidité de convergence en donnant plus d'importance aux erreurs finales de la réponse, tandis que le deuxième permet d'évaluer la robustesse du système en donnant plus d'importance aux grandes erreurs. Les résultats de ces deux indicateurs sont montrés dans les figures 4.10 à 4.12. Nous remarquons que le PSS proposé est plus performant que le PSS conventionnel pour tous les scénarios et points de fonctionnement considérés.

Afin d'illustrer les résultats de comparaison précédents, nous présentons dans les figures 4.13 à 4.15 quelques simulations effectuées en considérant le point de fonctionnement qui correspond à une forte charge. Ces simulations montrent que les réponses utilisant le PSS proposé présentent un meilleur dépassement et sont plus amorties par rapport à celles du PSS conventionnel. Nous pouvons donc conclure que l'approche proposée permet d'obtenir des solutions efficaces au problème de dimensionnement optimal du PSS pour le réseau mono-machine considéré.

## 4.3 Application à un réseau à 3 machines

Le deuxième exemple de réseau étudié dans ce travail est le réseau multimachine dit Western Systems Coordinating Council (WSCC) [79]. Ce réseau, dont le schéma unifilaire est représenté dans la figure 4.16, est composé de 3 machines alimentant 3 charges, de 3 transformateurs, 6 lignes de transport et 9 noeuds. Chaque machine est équipée d'un régulateur de tension (AVR), tandis que seulement les machines n°2 et n°3 sont équipées en plus d'un PSS. Dans ce cas, la machine n°1 est considérée comme un noeud de référence. Les données de ce réseau sont détaillées dans l'annexe C.

### 4.3.1 Formulation du problème

Comme pour le cas du réseau mono-machine, le problème considéré est de rechercher les paramètres optimaux des PSS qui permettent d'améliorer la stabilité et les performances du réseau. Dans ce cas, nous définissons un problème à trois fonctions objectifs. Les deux premières fonctions sont conçues pour optimiser la robustesse des deux machines n°2 et n°3 du réseau, indépendamment l'une de l'autre, tandis que la troisième fonction objectif permet de maximiser l'amortissement global du système. Le problème est donc formulé de la façon suivante :

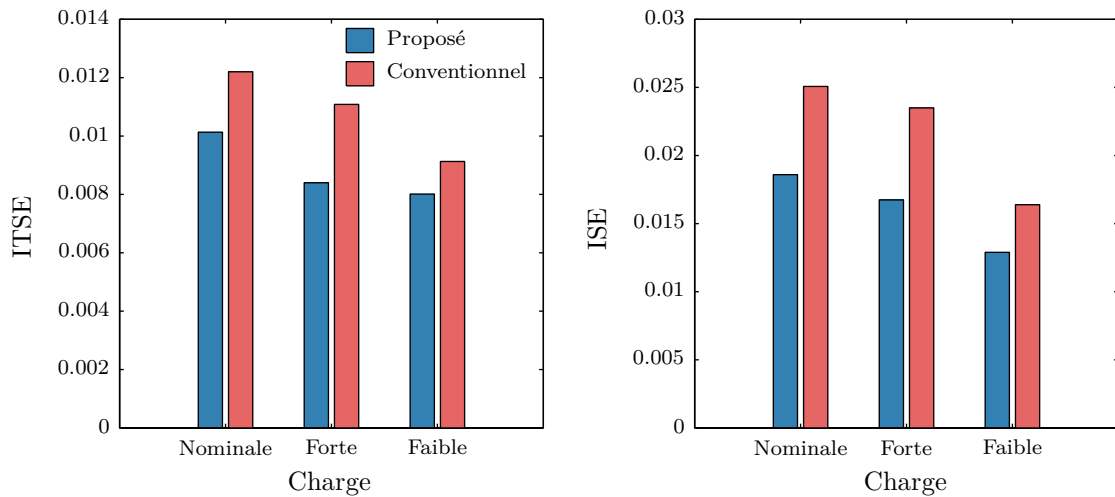


Figure 4.10 : Performances du PSS proposé (scénario n°1).

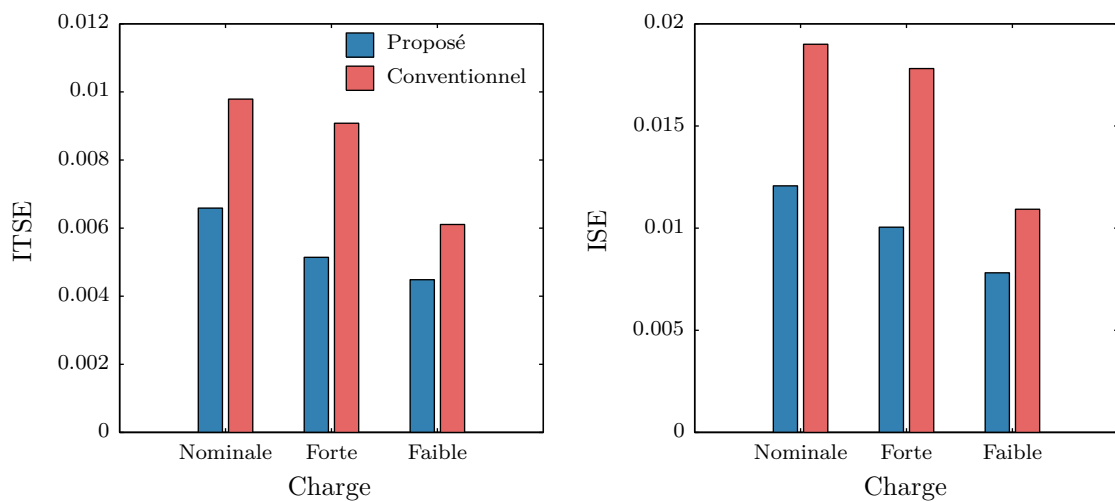


Figure 4.11 : Performances du PSS proposé (scénario n°2).

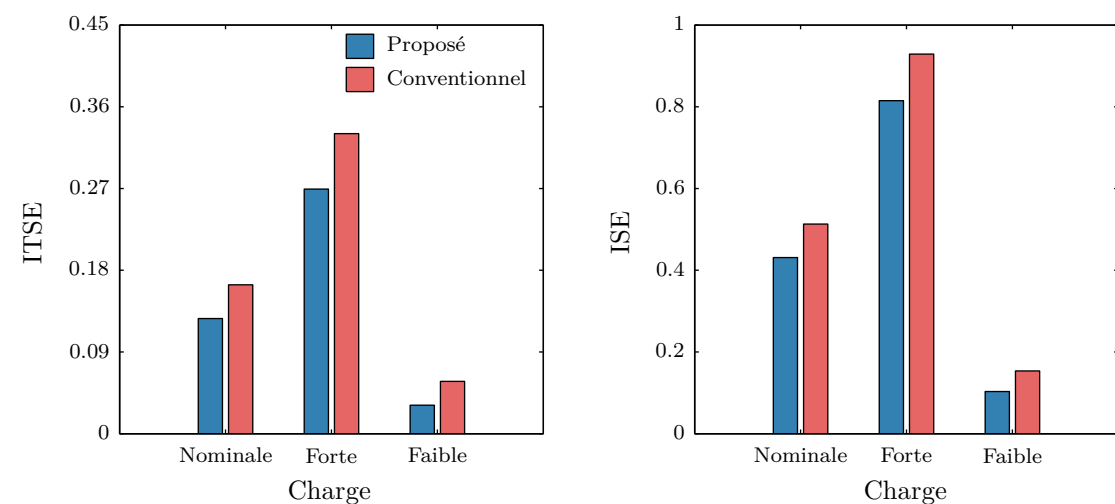


Figure 4.12 : Performances du PSS proposé (scénario n°3).

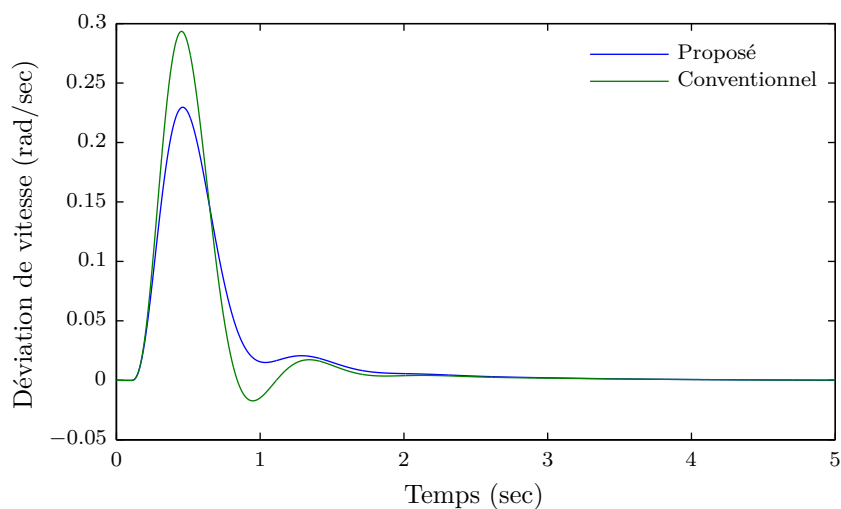


Figure 4.13 : Réponses du système (scénario n°1).

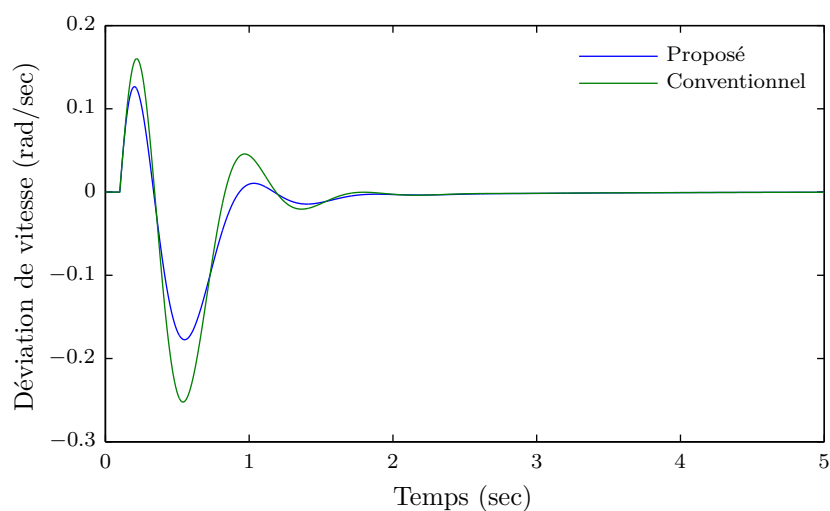


Figure 4.14 : Réponses du système (scénario n°2).

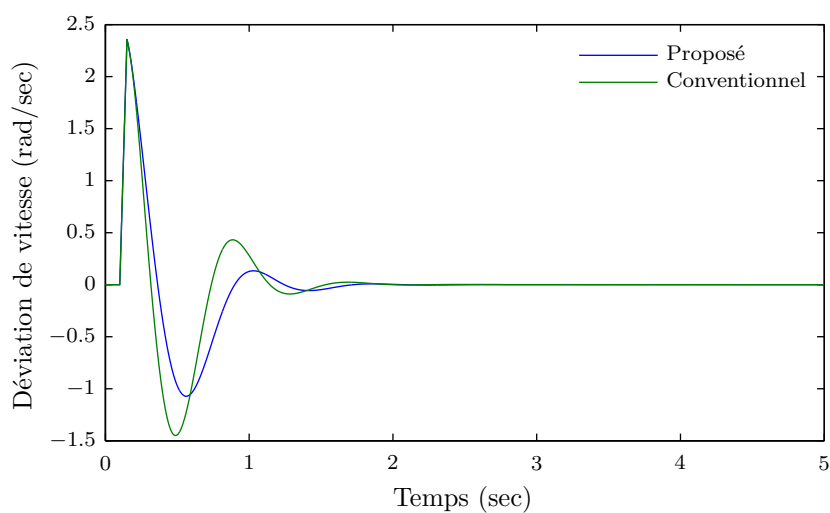


Figure 4.15 : Réponses du système (scénario n°3).

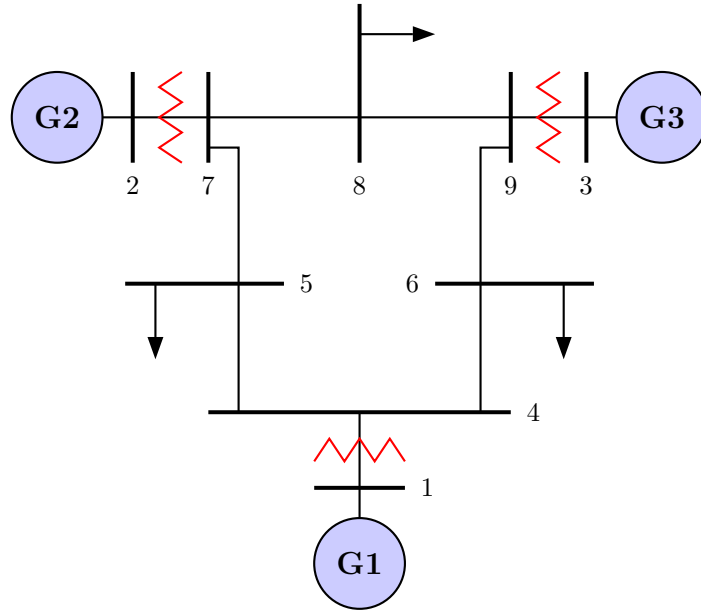


Figure 4.16 : Réseau à 3 machines.

$$\min F(\mathbf{x}) = \begin{pmatrix} \|T_{wz_1}\|_{\infty} \\ \|T_{wz_2}\|_{\infty} \\ -\min_i(\xi_i) \end{pmatrix} \quad (4.2)$$

où  $\mathbf{x}$  est le vecteur des variables de décision,  $T_{wz_1}$  et  $T_{wz_2}$  représentent les fonctions de transfert des systèmes linéarisés autour du point de fonctionnement nominal et reliant l'entrée exogène  $w$  et les variables cibles  $z_1$  et  $z_2$ , respectivement. Dans ce cas, la variable  $w$  est considérée comme étant une perturbation additive à l'entrée de l'AVR de la machine n°1, tandis que  $z_1$  et  $z_2$  représentent respectivement les déviations des vitesses rotoriques des machines n°2 et n°3 par rapport à la première machine. Par ailleurs,  $\xi_i$  représente le coefficient d'amortissement associé à la  $i^{\text{ème}}$  valeur propre du système. Dans ce cas, la contrainte  $\xi_i \geq 0.3$  est ajoutée au problème afin d'assurer un amortissement acceptable des oscillations électromécaniques. Les limites de l'espace de recherche pour ce problème sont données dans le tableau 4.4.

### 4.3.2 Résultats de l'optimisation

Le problème défini dans l'équation 4.2 est résolu en utilisant les algorithmes NSGA-II, GDE3, AMOSA, SMPPO et MODE-2A de la même façon que pour le réseau précédent. Toutefois, la taille des populations et/ou des archives est fixée ici à 100 et le nombre total d'évaluations des fonctions objectifs à  $4 \cdot 10^3$ .

Les résultats des métriques  $H$  et  $S$  sont rapportés dans le tableau 4.5. Nous constatons que les meilleurs résultats sont également obtenus en utilisant l'algorithme

Tableau 4.4 : Espace de recherche pour le à 3 machines.

Paramètres	Min	Max
Zéros	-100	0
Gains	0.1	3500

MODE-2A, et que les autres résultats de comparaison sont semblables à ceux du problème précédent. Ainsi, nous pouvons conclure que MODE-2A est un algorithme robuste qui permet de résoudre des problèmes d'optimisation réels d'une manière plus efficace que les autres algorithmes.

Par ailleurs, afin d'illustrer avec plus de détails les résultats de la métrique  $H$ , nous présentons dans la figure 4.17 les variations de cette mesure en fonction des étapes de l'optimisation. Les courbes obtenues montrent que l'algorithme MODE-2A converge plus rapidement que les autres algorithmes, même si ses performances sont proches de celles de NSGA-II. D'autre part, bien qu'il présente de mauvais résultats durant les premières étapes de l'optimisation, nous remarquons que l'algorithme GDE3 se classe troisième au stade final de l'optimisation. Ce résultat montre que les performances d'un algorithme de résolution dépendent notamment du nombre d'évaluations des fonctions objectifs effectuées durant la recherche.

Le front non dominé final du problème traité est illustré sur la figure 4.18. Nous constatons que ce front est convexe et que les trois fonctions objectifs du problème ne peuvent être minimisées d'une façon simultanée, ce qui signifie que les performances de robustesse des deux machines n°2 et n°3 ainsi que l'amortissement global du système sont des critères contradictoires.

Tableau 4.5 : Résultats des métriques pour le réseau à 3 machines.

	NSGAI	GDE3	AMOSA	SMPSO	MODE-2A
H	2.95e+000	2	2.49e+000	5	2.87e+000
S	2.65e-001	3	2.71e+001	5	2.06e-001

### 4.3.3 Résultats de simulation

Dans ce qui suit, nous présentons quelques résultats de simulation effectuées en utilisant le modèle non linéaire du réseau étudié. Le scénario considéré est celui d'une diminution simultanée de 0.05 p.u de la tension de référence des générateurs n°2 et n°3 du réseau. Les trois solutions utilisées dans les simulation sont celles qui correspondent aux points extrêmes du front non dominé de la figure 4.18, c'est-à-dire celles qui permettent d'avoir les meilleures performances de robustesse pour les deux machines n°2 et n°3 et également le meilleur amortissement global du système.

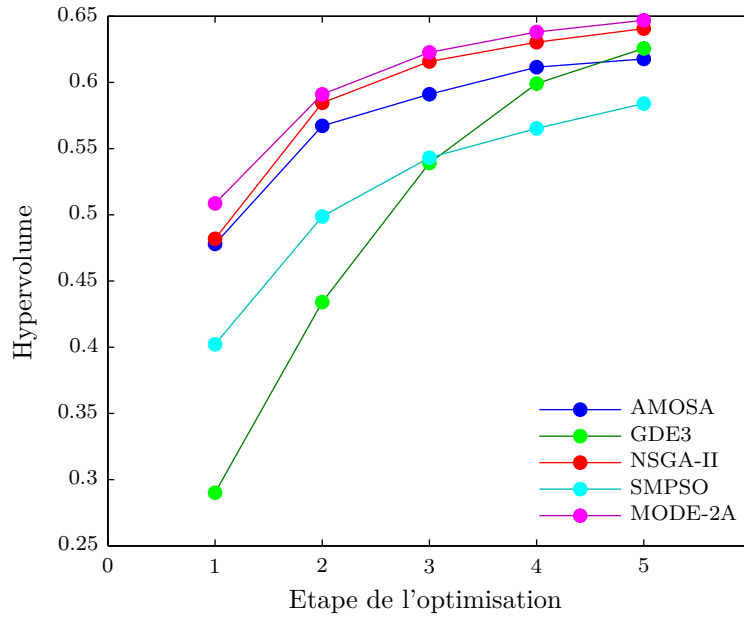


Figure 4.17 : Variation de l'hypervolume.

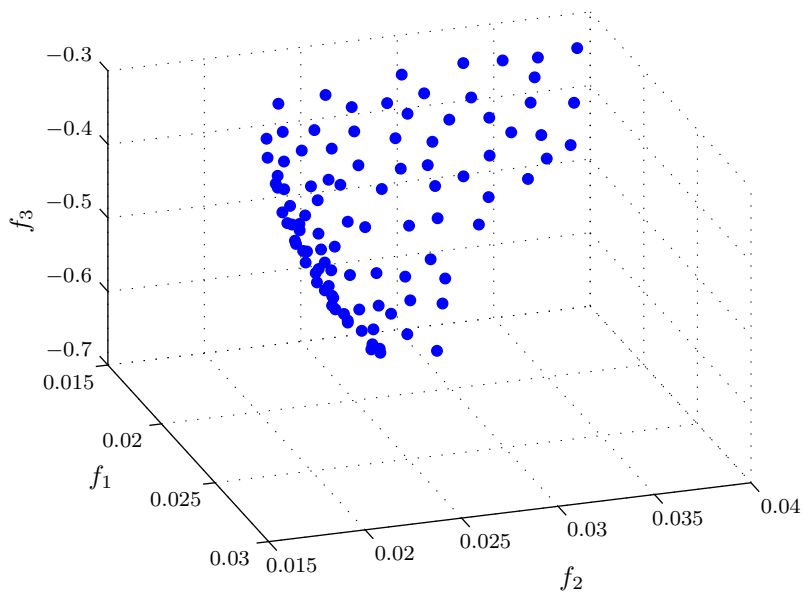


Figure 4.18 : Front non dominé.

Les résultats de simulation sont montrés dans les figures 4.19 à 4.21. Nous remarquons qu'en utilisant la première solution, la réponse de la machine n°2 est plus stable par rapport à celle de la machine n°3, tandis que la deuxième solution permet d'améliorer la réponse de la machine n°3, ceci signifie que les performances de ces deux machines du réseau ne peuvent pas être améliorées simultanément. Par ailleurs, en utilisant la troisième solution, les réponses de ces deux machines deviennent plus amorties et plus stables. Les mêmes remarques peuvent être faites concernant les signaux de commande qui sont illustrés sur les figures 4.22 à 4.24.



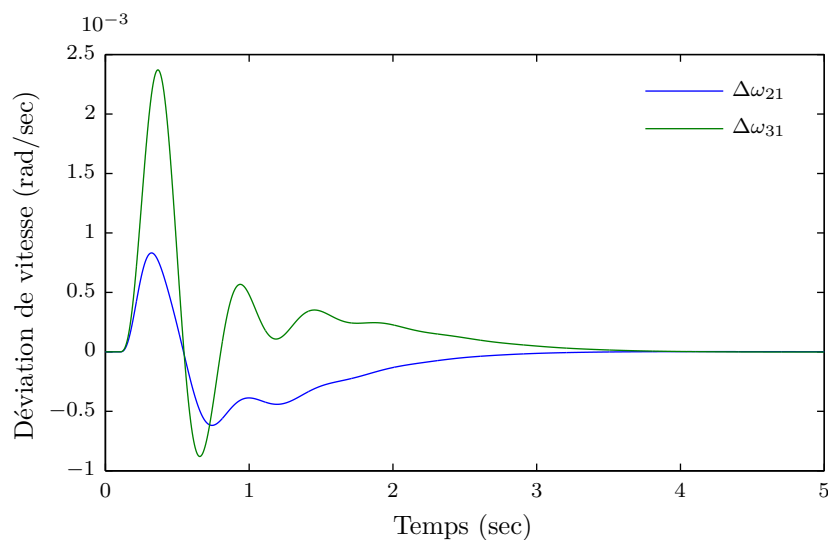


Figure 4.19 : Réponses du système avec la 1<sup>ère</sup> solution.

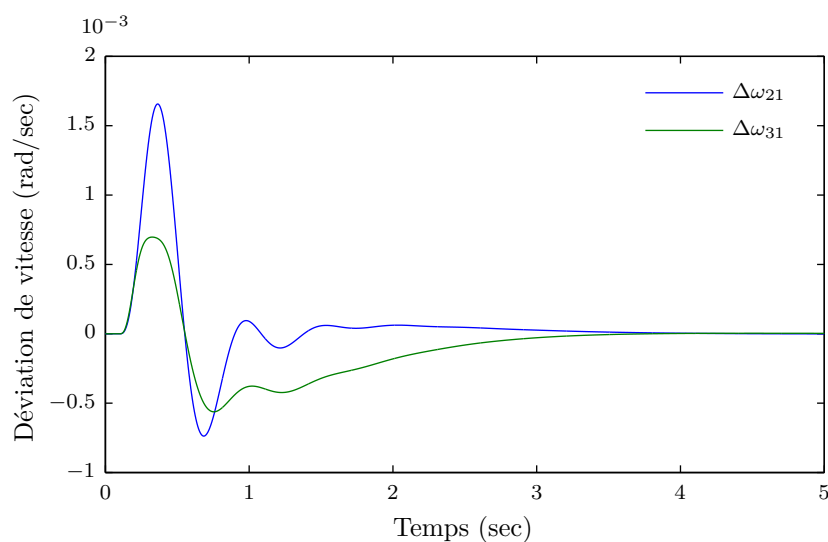


Figure 4.20 : Réponses du système avec la 2<sup>ème</sup> solution.

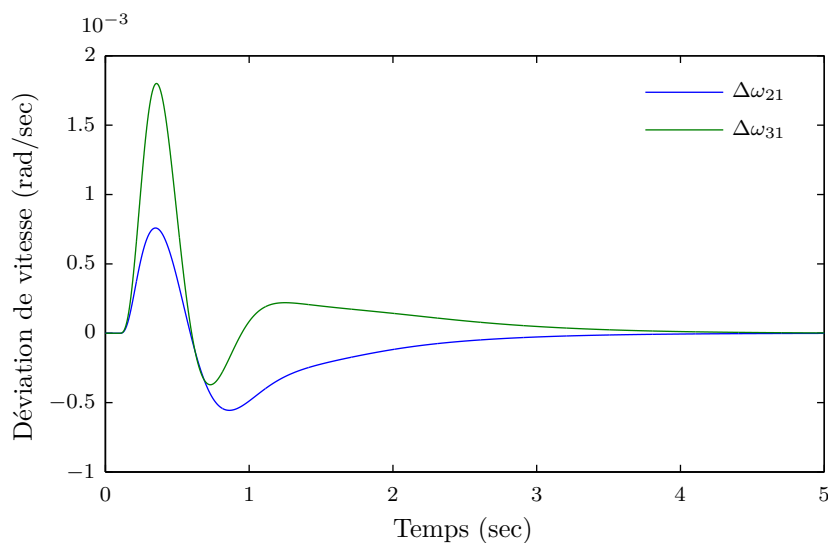


Figure 4.21 : Réponses du système avec la 3<sup>ème</sup> solution.

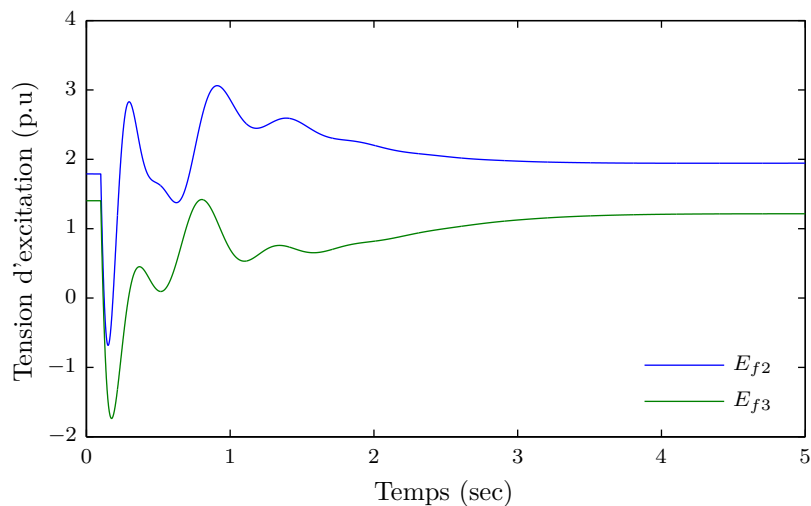


Figure 4.22 : Comparaison des performances des PSS (scénario n°1).

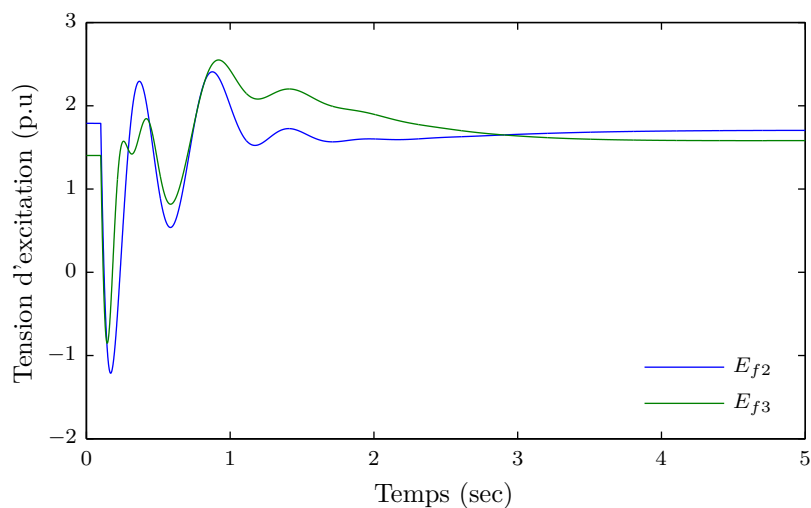


Figure 4.23 : Comparaison des performances des PSS (scénario n°2).

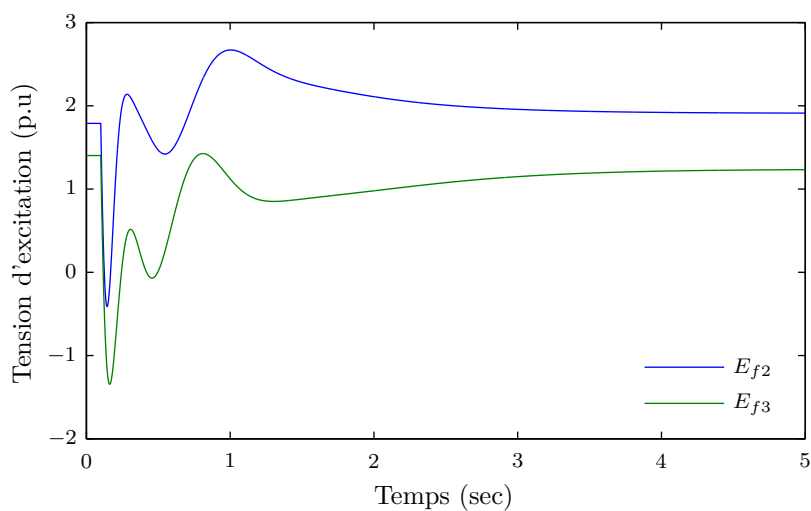


Figure 4.24 : Comparaison des performances des PSS (scénario n°3).

### 4.3.4 Sélection d'une solution de compromis

Pour sélectionner une solution de compromis, nous effectuons des simulations en utilisant toutes les solutions du front non dominé de la figure 4.18. Pour ce faire, nous considérons deux autres scénarios qui sont des courts-circuits d'une durée de 0.1 s survenus au bus 7 à la fin de la ligne 5-7 pour le deuxième scénario, et au bus 6 à la fin de la ligne 6-9 pour le troisième. Nous utilisons également les indicateurs ITAE et le dépassement afin d'évaluer les performances des solutions non dominées. Toutefois, nous considérons dans ce cas trois mesures de dépassement, deux pour les réponses des machines n°2 et n°3, désignées par  $D_2$  et  $D_3$ , respectivement, et une mesure pour le dépassement global  $D$  qui correspond à la moyenne de  $D_2$  et  $D_3$ .

Les résultats des tests sont montrés dans les figures 4.25 à 4.30, où les solutions sont représentées par des points dans le plan défini par les deux premières fonctions objectifs du problème. De plus, les valeurs des indicateurs ITAE et de dépassement sont illustrées par un dégradé de couleur du bleu vers le rouge. Ainsi, les points bleus correspondent aux solutions qui présentent les meilleures performances, tandis que la couleur rouge signifie que les performances sont dégradées.

Nous pouvons remarquer que les meilleures solutions d'après les indicateurs ITAE et le dépassement moyen  $D$  correspondent aux points qui se situent typiquement au centre du front non dominé. Par ailleurs, le dépassement  $D_2$  qui concerne la réponse de la machine n°2, est amélioré avec les solutions minimisant la première fonction objectif, et le dépassement  $D_3$  avec les solutions qui minimisent la deuxième fonction. Ces résultats montrent que les performances des machines n°2 et n°3 ainsi que du système global ne peuvent être améliorées simultanément.

A partir des résultats précédents, nous choisissons comme solution de compromis celle qui correspond au point dans l'espace des objectifs dont les coordonnées sont  $(f_1, f_2, f_3) = (0.023, 0.023, -0.51)$ . Afin d'illustrer les performances de cette solution, celle-ci est comparée avec une autre solution obtenue en utilisant une méthode conventionnelle qui permet de maximiser l'amortissement global du système. Les paramètres de ces deux solutions sont donnés dans le tableau 4.6.

Les résultats de comparaison des indicateurs ITSE et ISE sont donnés dans les figures 4.31 à 4.33. Nous constatons que les performances de la solution proposée sont meilleures que celles de la solution conventionnelle, et ce pour tous les scénarios et points de fonctionnement considérés.

Afin d'illustrer les résultats précédents, nous présentons dans les figures 4.34 à 4.39 quelques simulations effectuées en utilisant le modèle non linéaire du réseau, en considérant le point de fonctionnement correspondant à une forte charge. Les résultats montrent que la solution proposée permet d'améliorer la stabilité du réseau en diminuant les dépassements et les temps de convergence des réponses du système.

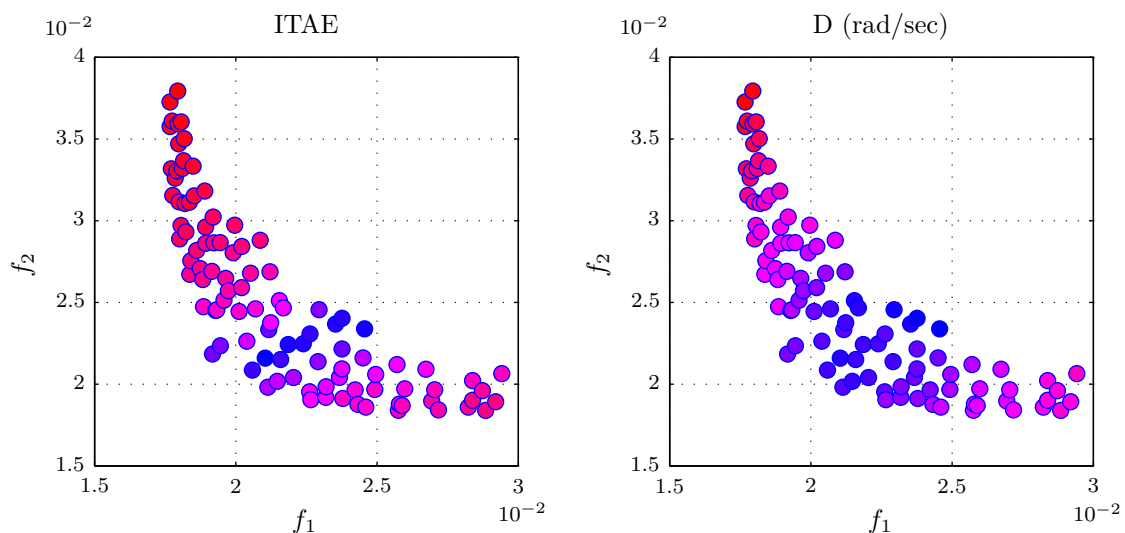


Figure 4.25 : Comparaison des performances des PSS (scénario n°1).

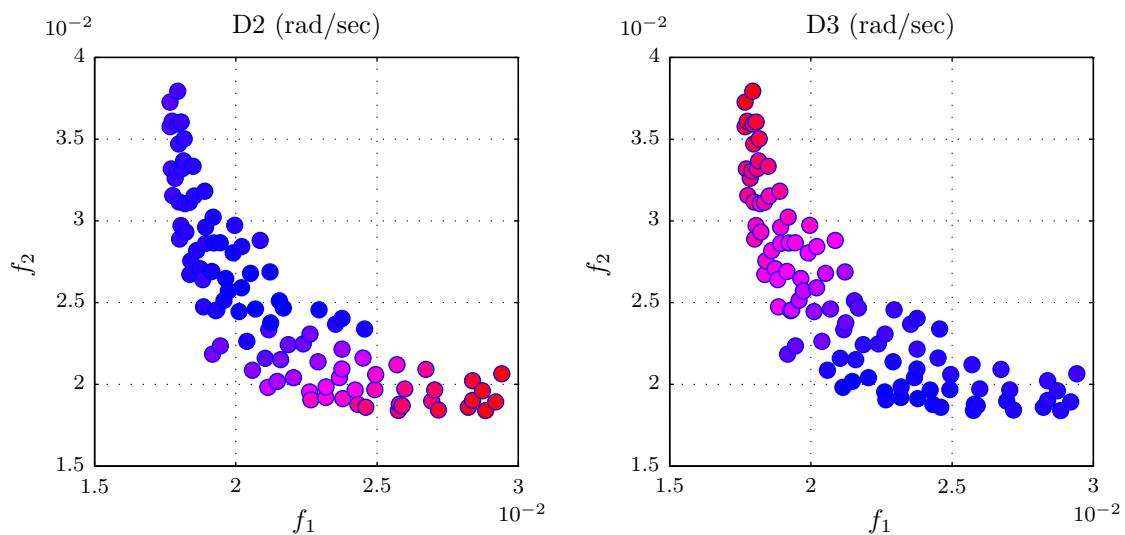


Figure 4.26 : Comparaison des performances des PSS (scénario n°1).

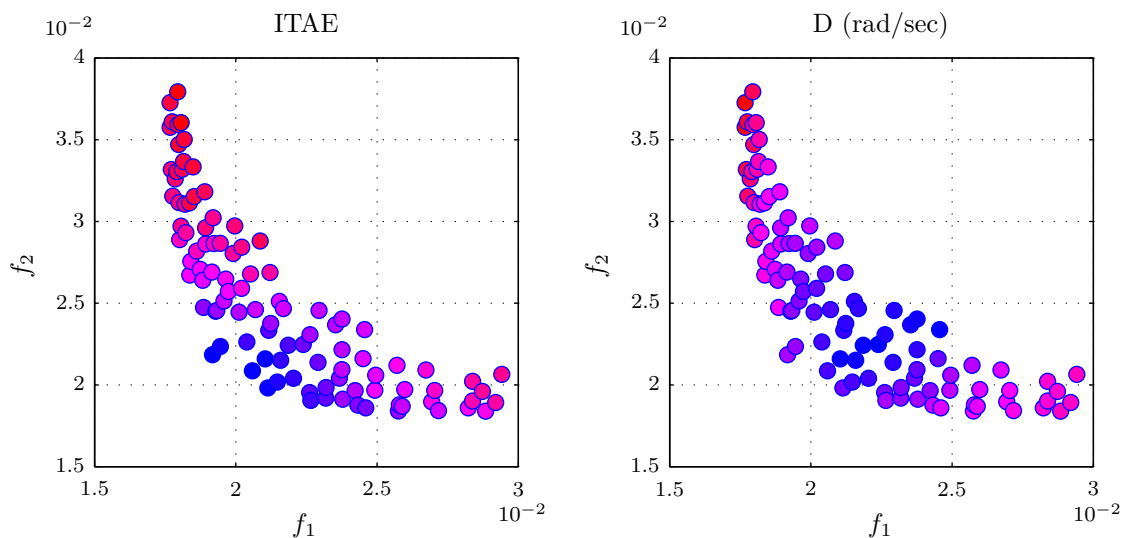


Figure 4.27 : Comparaison des performances des PSS (scénario n°2).

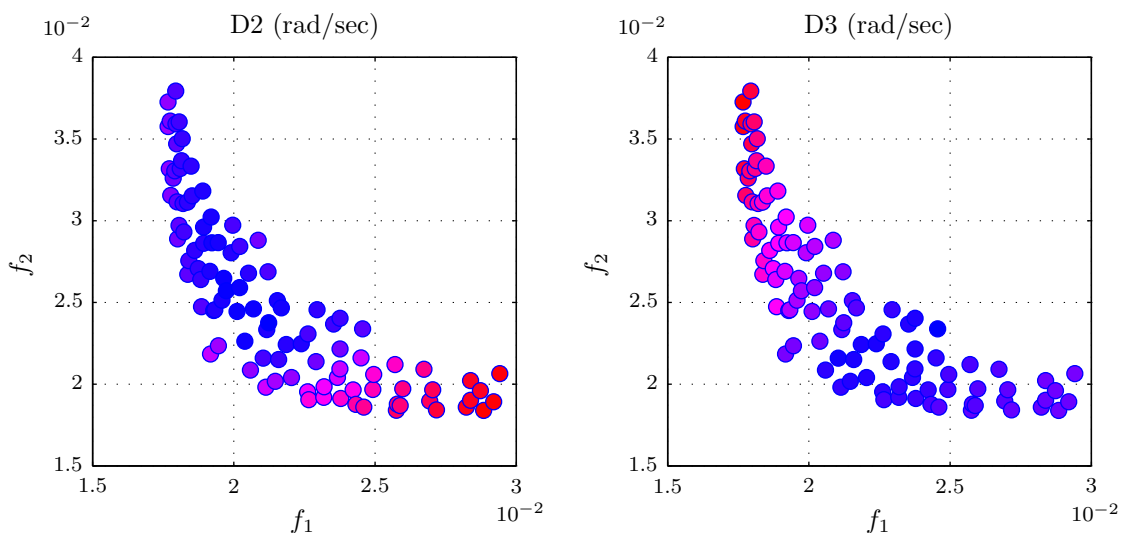


Figure 4.28 : Comparaison des performances des PSS (scénario n°2).

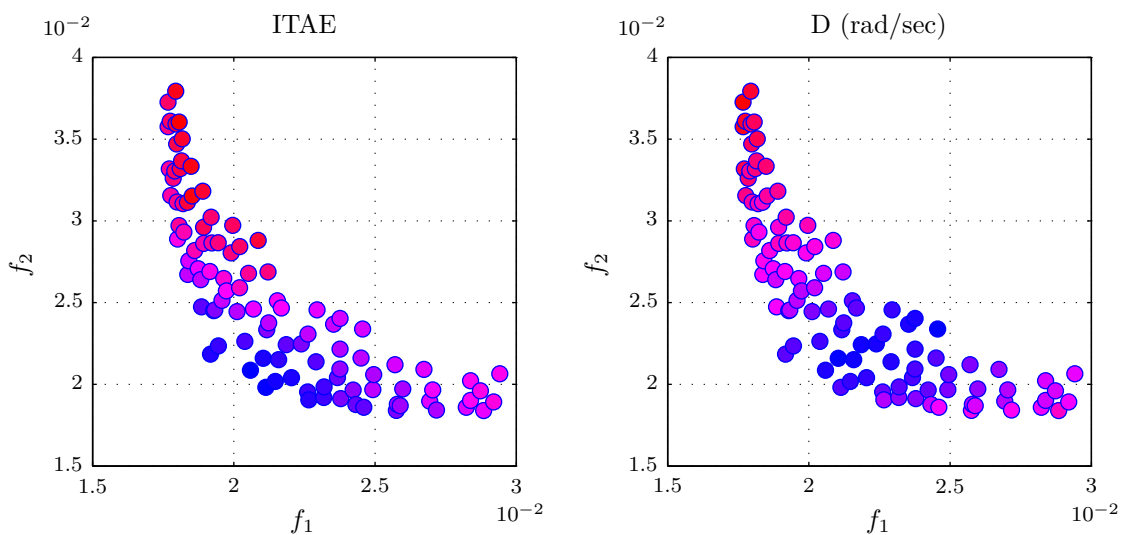


Figure 4.29 : Comparaison des performances des PSS (scénario n°3).

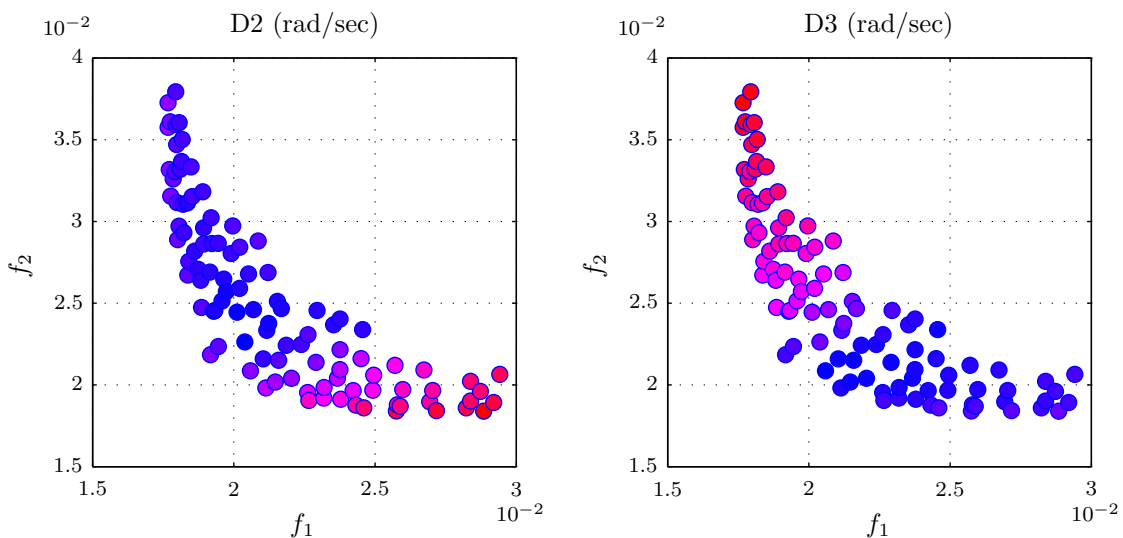


Figure 4.30 : Comparaison des performances des PSS (scénario n°3).

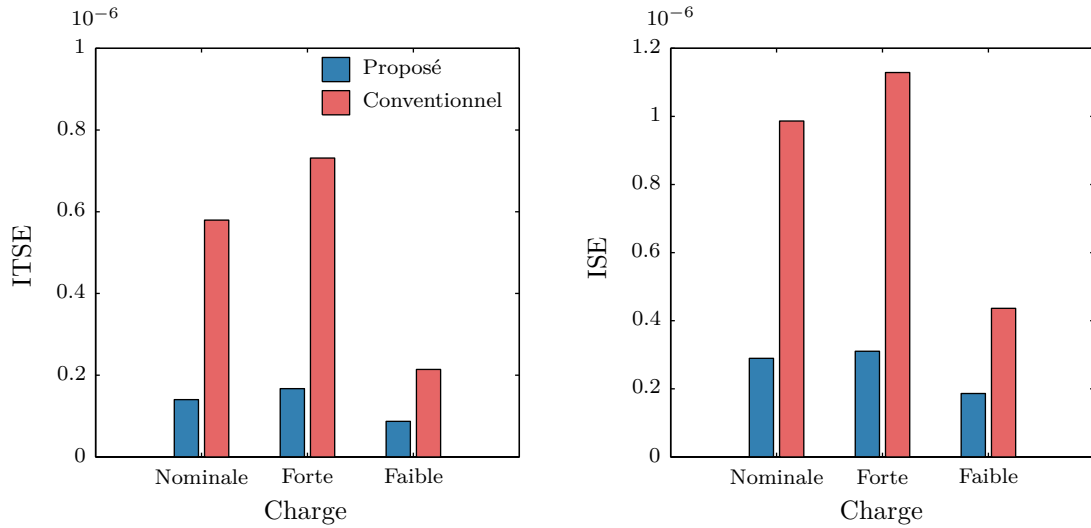


Figure 4.31 : Performances du PSS proposé (scénario n°1).

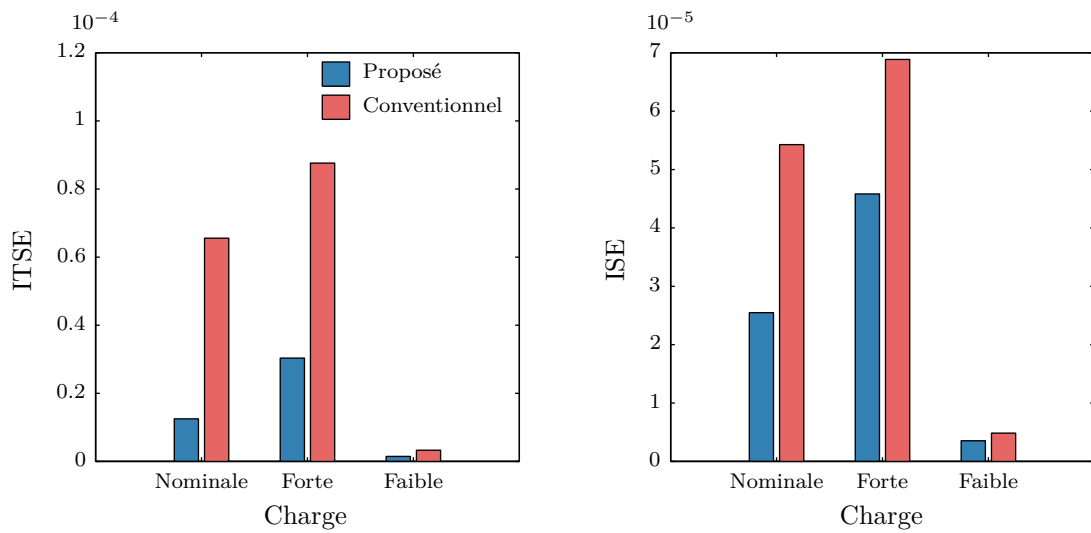


Figure 4.32 : Performances du PSS proposé (scénario n°2).

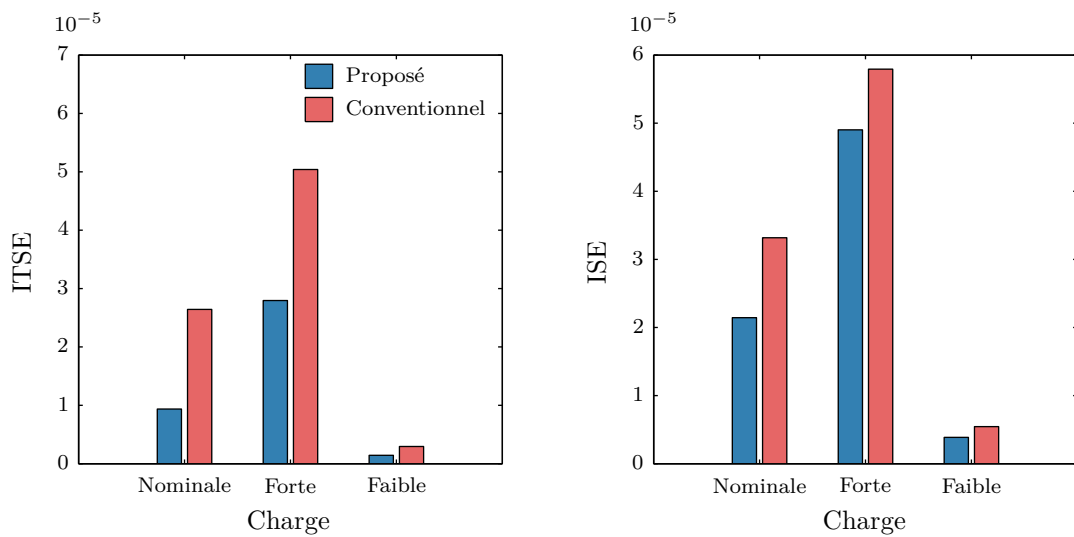


Figure 4.33 : Performances du PSS proposé (scénario n°3).

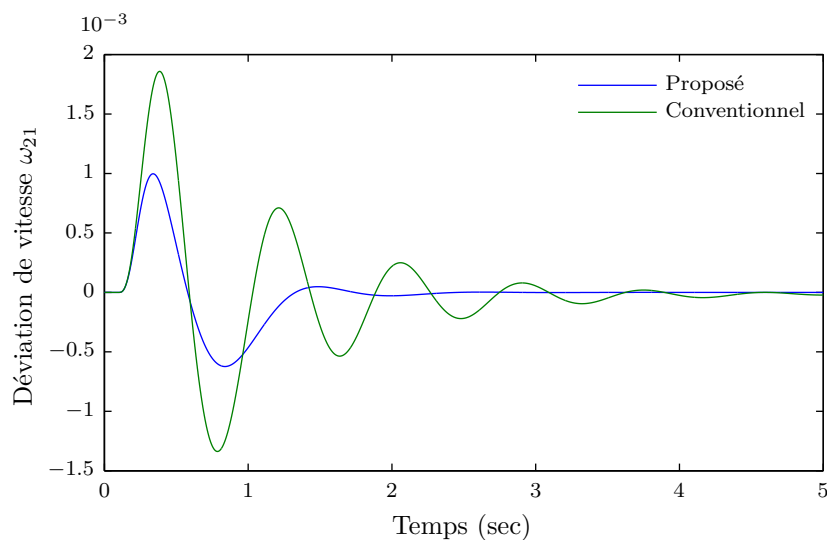


Figure 4.34 : Réponses de la 2<sup>ème</sup> machine (scénario n°1).

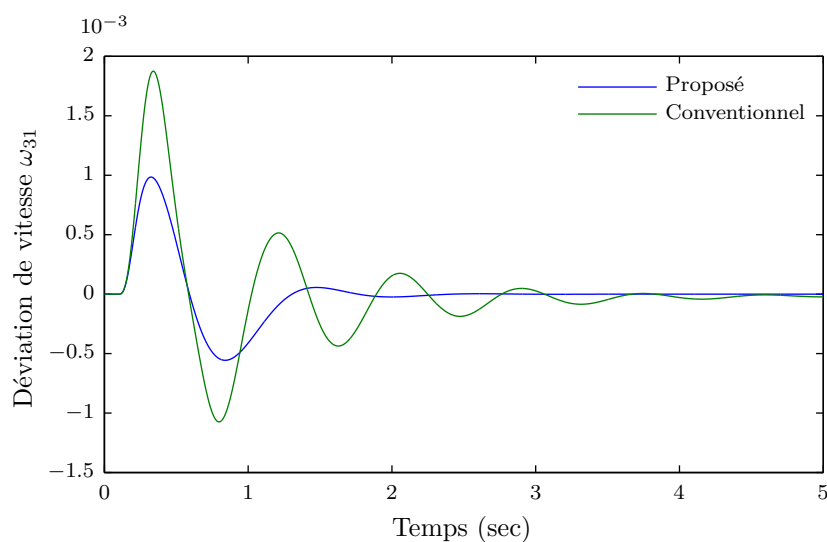


Figure 4.35 : Réponses de la 3<sup>ème</sup> machine (scénario n°1).

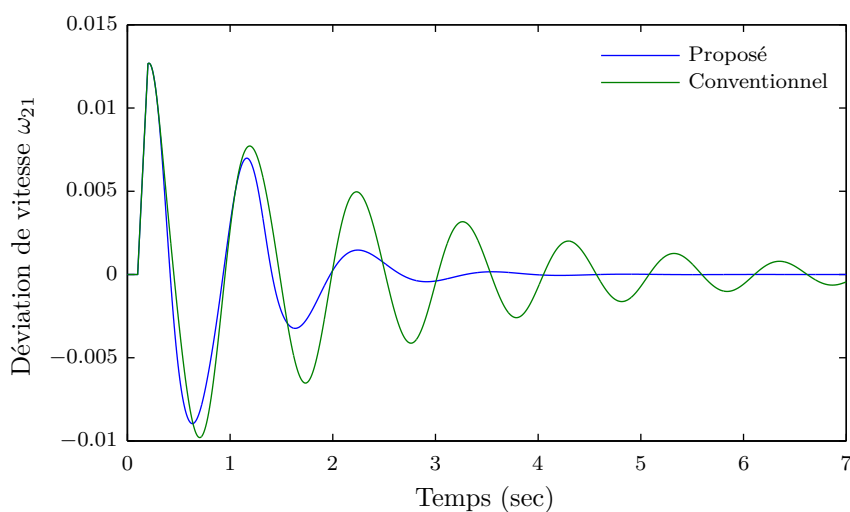


Figure 4.36 : Réponses de la 2<sup>ème</sup> machine (scénario n°2).

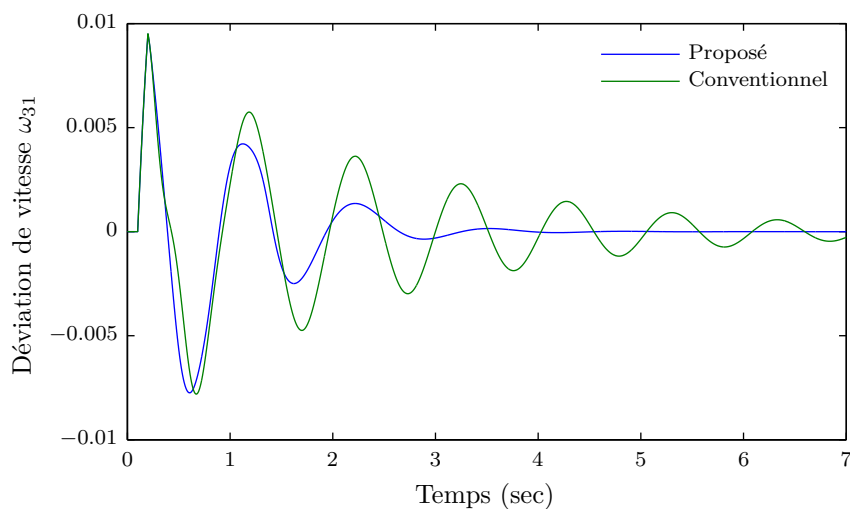


Figure 4.37 : Réponses de la 3<sup>ème</sup> machine (scénario n°2).

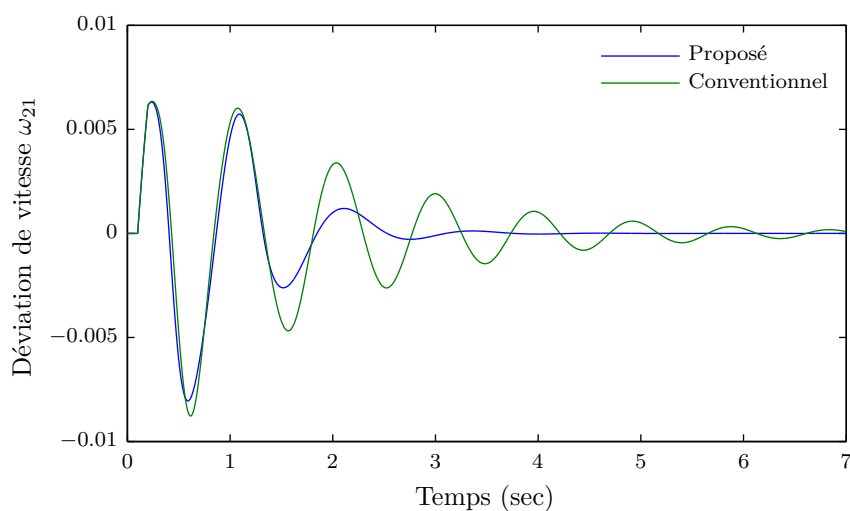


Figure 4.38 : Réponses de la 2<sup>ème</sup> machine (scénario n°3).

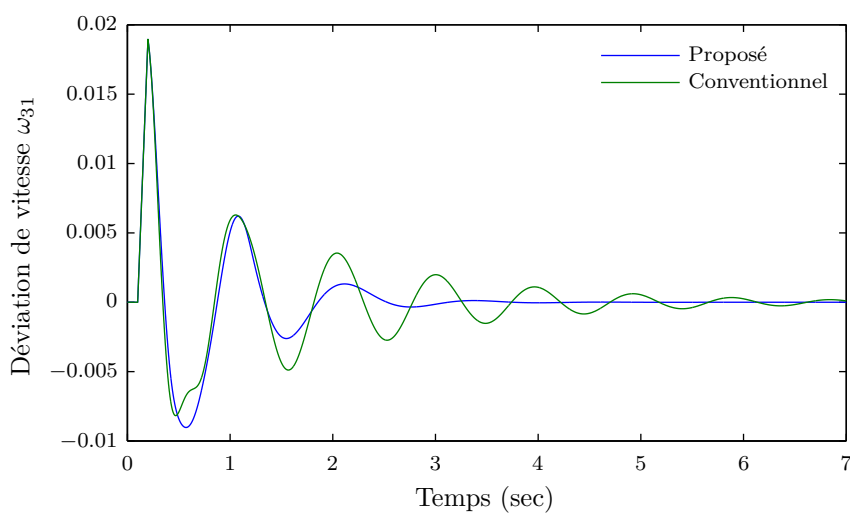


Figure 4.39 : Réponses de la 3<sup>ème</sup> machine (scénario n°3).



Tableau 4.6 : Paramètres des PSS sélectionnés pour le réseau à 3 machines.

PSS	Machine #	$K$	$T_1$	$T_2$	$T_3$	$T_4$
Conventionnel	2-3	3.9	0.18	0.05	0.18	0.05
Proposé	2	25.19	0.07	0.01	0.07	0.01
	3	31.37	0.07	0.01	0.09	0.01

## 4.4 Application à un réseau à 10-machines

Le dernier exemple de réseau électrique étudié dans ce travail est le réseau dit New England illustré sur la figure 4.40 [84]. Ce réseau est composé de 10 machines alimentant 19 charges, de 12 transformateurs, 34 lignes de transport et 39 noeuds. Les données de ce réseau sont présentées dans l'annexe D.

### 4.4.1 Formulation du problème

Le problème défini pour ce réseau concerne la minimisation de deux fonctions objectifs. Celles-ci mesurent la robustesse et la rapidité du système en boucle fermée et sont établies sur la base des valeurs propres du système [85]. Ces valeurs sont calculées en linéarisant le système autour du point de fonctionnement nominal. La formulation mathématique de ce problème est la suivante :

$$\min F(\mathbf{x}) = \begin{pmatrix} \frac{1}{l_1} \sum_{i=1}^{l_1} \sigma_i^1 \\ \frac{1}{l_2} \sum_{i=1}^{l_2} \sigma_i^2 \end{pmatrix} \quad (4.3)$$

où  $\mathbf{x}$  est le vecteur qui représente les variables de décision,  $\sigma_i^1$  sont les  $l_1$  premières parties réelles des valeurs propres complexes classées selon un ordre décroissant, et  $\sigma_i^2$  sont les  $l_2$  premières parties réelles de toutes les valeurs propres classées également selon un ordre décroissant. En minimisant la première fonction objectif, l'effet des modes oscillatoires est diminué, ce qui permet d'améliorer l'amortissement global du système. En minimisant la deuxième fonction objectif, la partie réelle des modes dominants est diminuée ce qui améliore donc la convergence du système. Enfin, les valeurs  $l_1 = 8$  et  $l_2 = 22$  sont considérées pour ce réseau, et les limites de l'espace de recherche des variables de décision sont données dans le tableau 4.7.

### 4.4.2 Résultats de l'optimisation

Le problème défini dans l'équation 4.3 est résolu en utilisant les algorithmes NSGA-II, GDE3, AMOSA, SMSP et MODE-2A, de la même façon que dans les cas précédents.

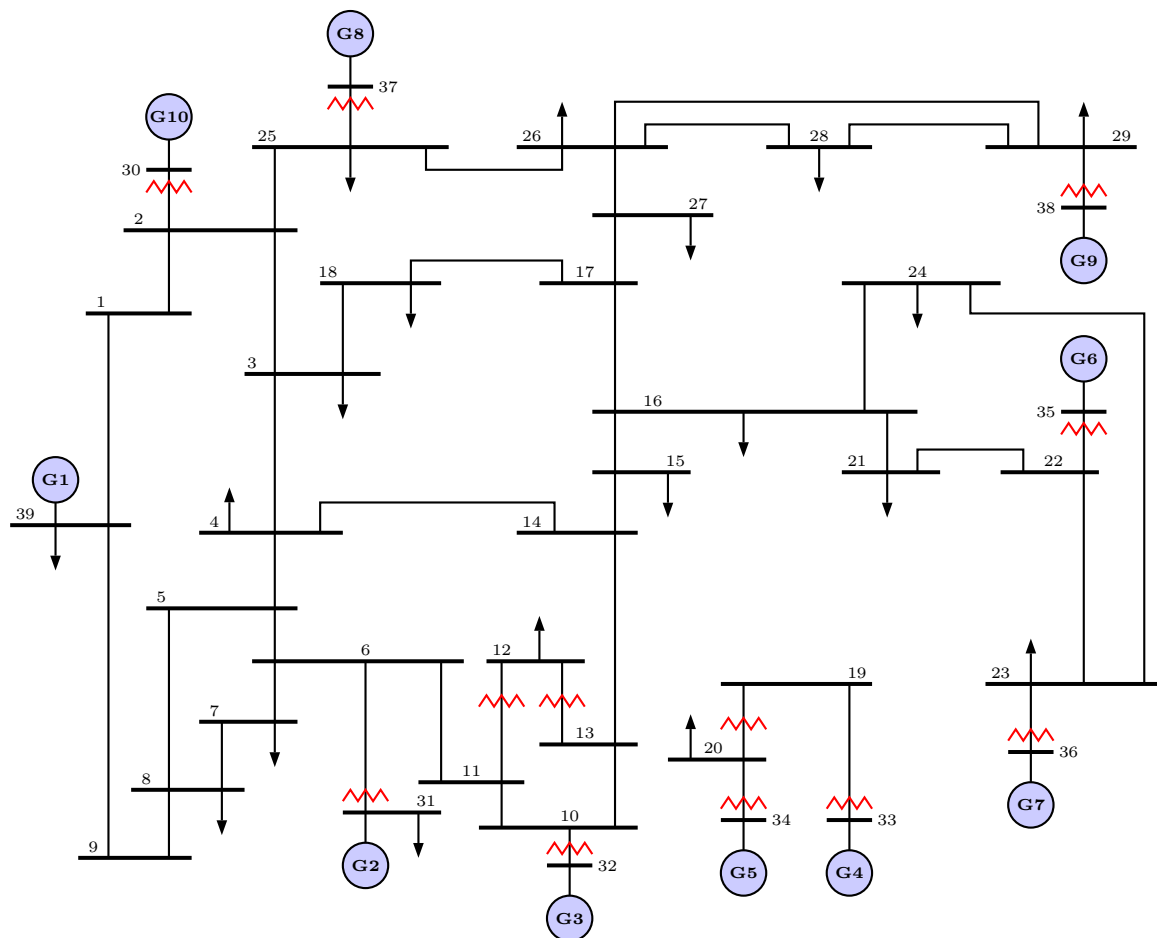


Figure 4.40 : Réseau à 10 machines.

Toutefois, le nombre de solutions non dominées est dans ce cas égal à 50 et le nombre total d'évaluations des fonctions objectifs est fixé à 2500.

Les résultats des métriques  $H$  et  $S$  sont donnés dans le tableau 4.8. Nous constatons que le meilleur résultat pour l'hypervolume est celui de l'algorithme SMPSO tandis que MODE-2A se classe deuxième. Par contre, pour la métrique d'espacement, les classements sont inversés pour ces deux algorithmes.

Pour mieux illustrer les performances des algorithmes utilisés concernant la métrique  $H$ , nous présentons dans la figure 4.41 les variations de cette métrique en fonction des étapes de l'optimisation. Nous remarquons que globalement les meilleurs résultats sont ceux de l'algorithme SMPSO, bien que, à la dernière étape de l'optimisation la meilleure

Tableau 4.7 : Espace de recherche pour le réseau à 10 machines.

Paramètres	Min	Max
Zéros	-100	0
Gains	0.1	$1.5 \cdot 10^4$

Tableau 4.8 : Résultats des métriques pour le réseau à 10 machines.

	NSGAI	GDE3	AMOSA	SMPSO	MODE-2A
H	1.70e+000	3	1.55e+000	5	1.56e+000
S	2.61e-001	3	3.72e+000	5	3.69e+000

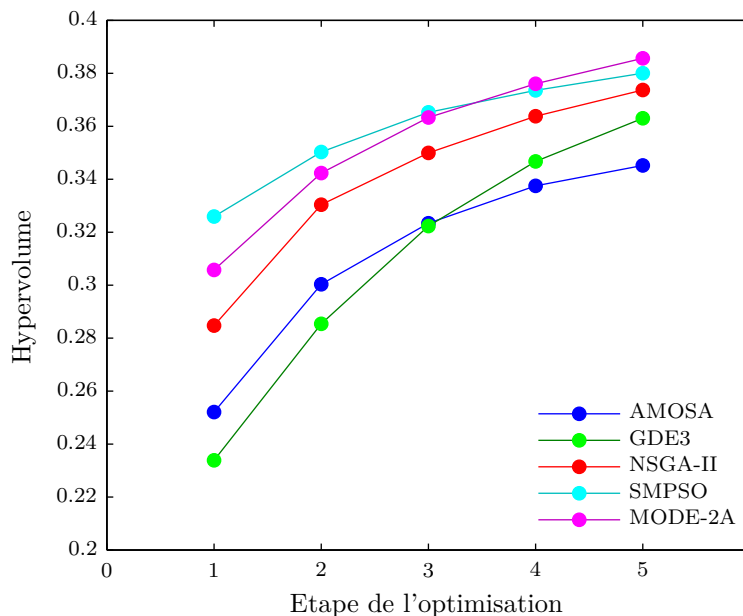


Figure 4.41 : Variation de l'hypervolume.

performance est obtenue avec MODE-2A. Nous constatons également que la convergence de l'algorithme GDE3 est très lente dans la première étape, mais devient beaucoup plus rapide durant les dernières étapes.

Le front non dominé final du problème traité est présenté dans la figure 4.42. Dans ce cas, le front est non convexe, et nous pouvons clairement remarquer le compromis qui existe entre les critères du problème.

### 4.4.3 Résultats de simulation

Dans le but de valider les résultats précédents, nous présentons dans ce qui suit quelques simulations effectuées en utilisant trois solutions non dominées. Les deux premières solutions correspondent aux points extrêmes du front de la figure 4.41, tandis que la troisième représente le point milieu de ce front. Le scénario considéré est une augmentation simultanée de 0.05 s de la tension de référence des machines n°2 à 10.

Les réponses du système en utilisant les deux premières solutions sont montrées dans les figures 4.43 à 4.46. Ainsi, nous constatons qu'avec la première solution, c'est-à-dire celle qui minimise la première fonction objectif, les réponses du système sont plus amorties

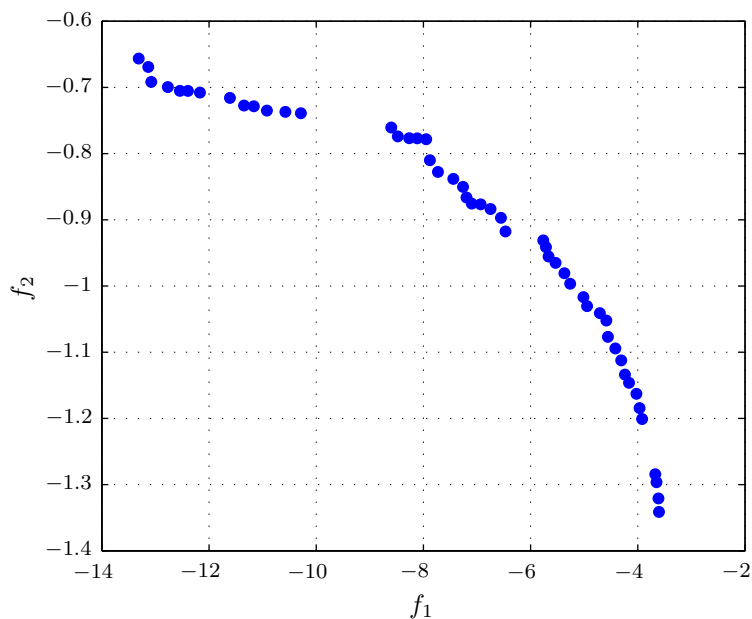


Figure 4.42 : Front non dominé.

en termes de dépassement et de fréquences d'oscillation. Par contre, avec la deuxième solution, le dépassement et les oscillations sont plus importants mais la convergence du système est améliorée. Nous constatons également que les signaux de commande sont plus stables en utilisant la première solution.

Par ailleurs, afin d'illustrer le compromis précédent, nous montrons dans les figures 4.47 et 4.48 les réponses de la machine n°2 en utilisant les trois solutions précédentes. Ainsi, nous constatons qu'un compromis existe entre les réponses de cette machine en termes de stabilité et de temps de convergence.

#### 4.4.4 Sélection d'une solution de compromis

Afin de sélectionner une solution de compromis, des simulations sont effectuées en utilisant toutes les solutions du front non dominé de la figure 4.42. Ces solutions sont comparées entre elles au moyen des indicateurs ITAE et le dépassement. Pour ce faire, deux autres scénarios sont considérés ; un court-circuit d'une durée de 0.15 s survenu au bus 15 à l'extrémité de la ligne 14-15, et un autre court-circuit qui semblable au précédent mais d'une durée de 0.1 s accompagné d'une augmentation de 0.05 p.u de la tension de référence des machines n°2 à 10.

Les résultats de ces tests sont donnés dans les figures 4.49 à 4.51, où les solutions sont classées selon l'ordre croissant de la première fonction objectif. Nous constatons que l'indicateur ITAE diminue en général lorsque le dépassement augmente, c'est-à-dire lorsque les valeurs de la première fonction objectif augmentent. Ceci montre que ces deux critères sont contradictoires et ne peuvent être améliorés simultanément.

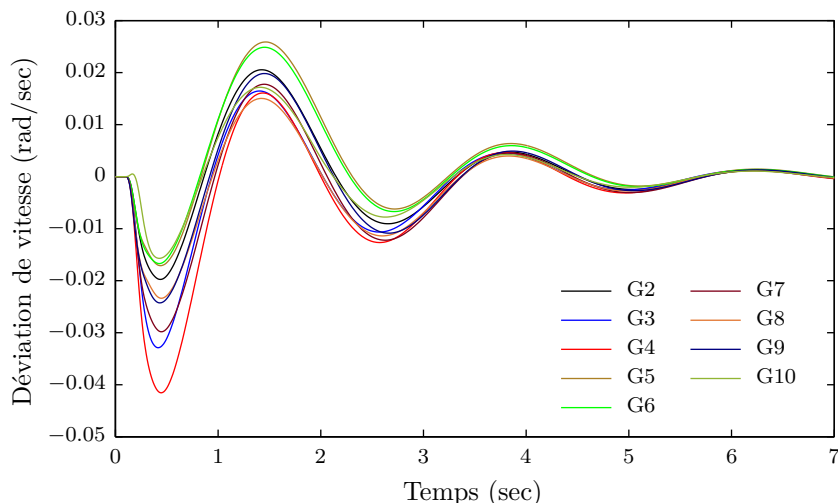


Figure 4.43 : Réponses du système avec la 1<sup>ère</sup> solution.

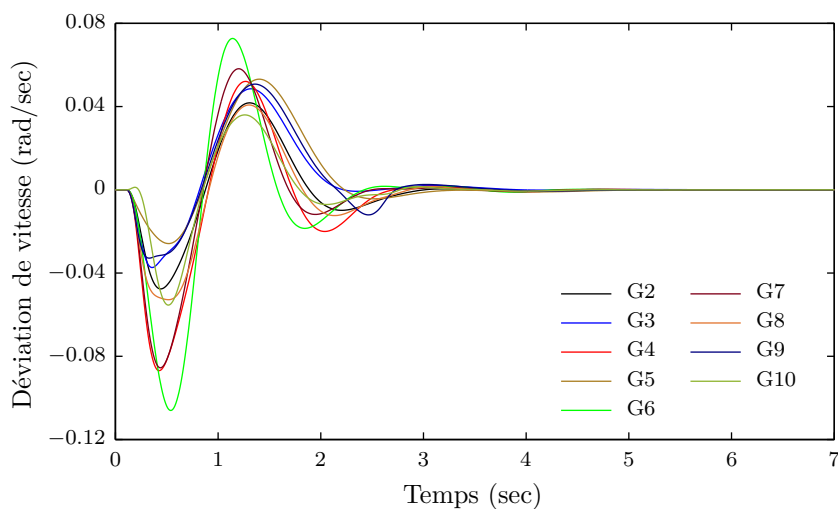


Figure 4.44 : Réponses du système avec la 2<sup>ème</sup> solution.

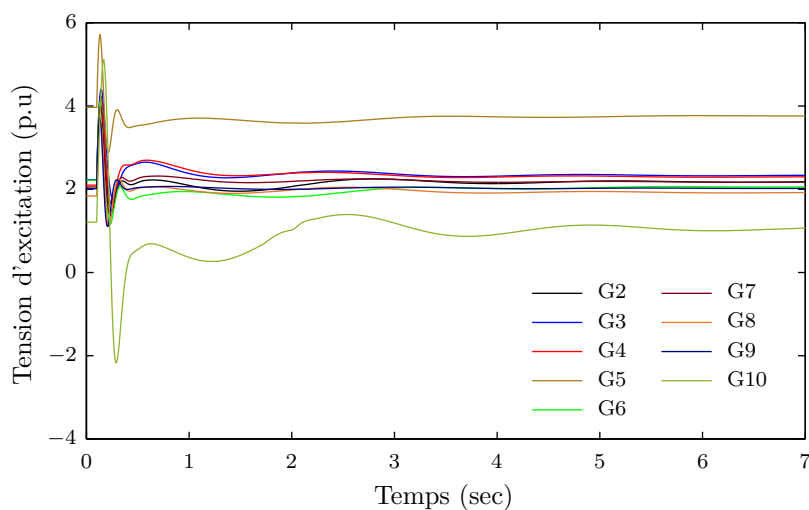


Figure 4.45 : Tensions d'excitation avec la 1<sup>ère</sup> solution.

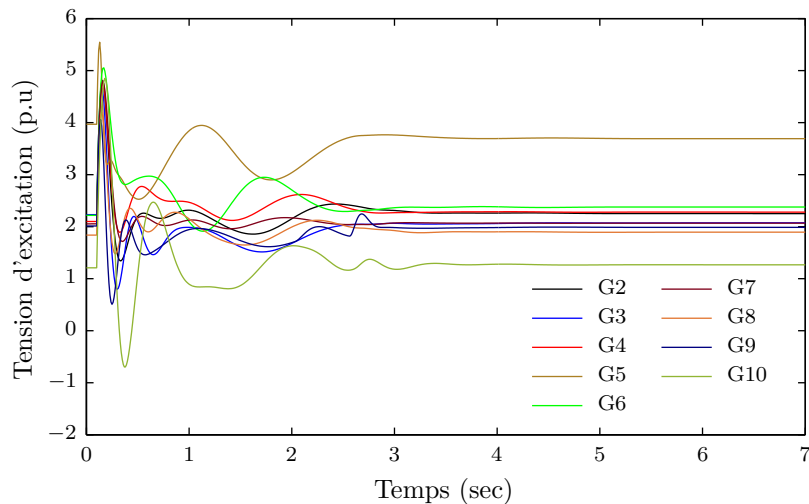


Figure 4.46 : Tensions d'excitation avec la 2<sup>ème</sup> solution.

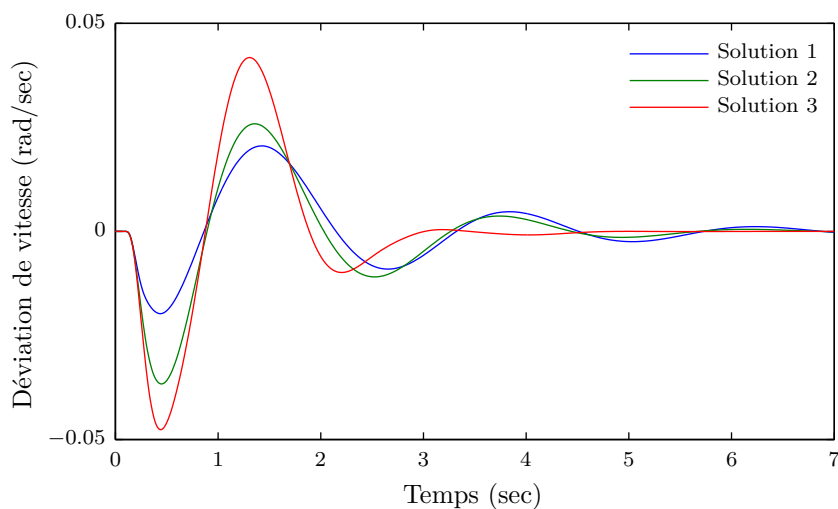


Figure 4.47 : Déviations de vitesse de la 2<sup>ème</sup> machine.

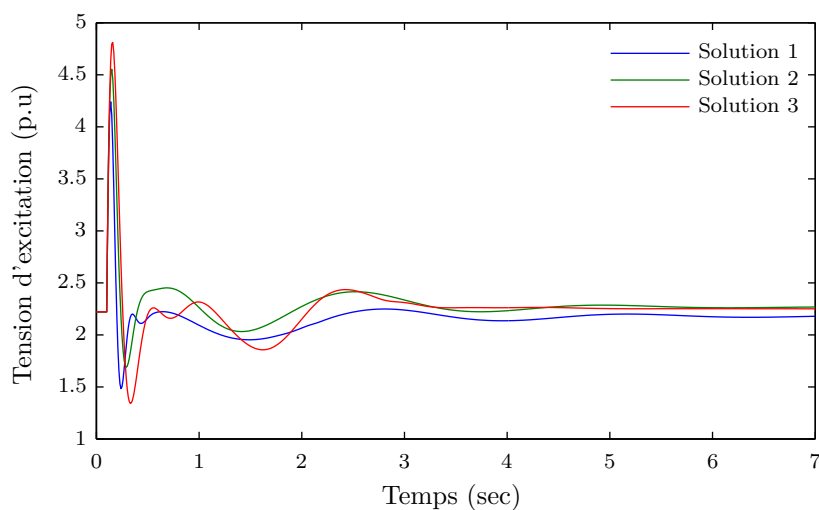


Figure 4.48 : Tensions d'excitation de la 2<sup>ème</sup> machine.

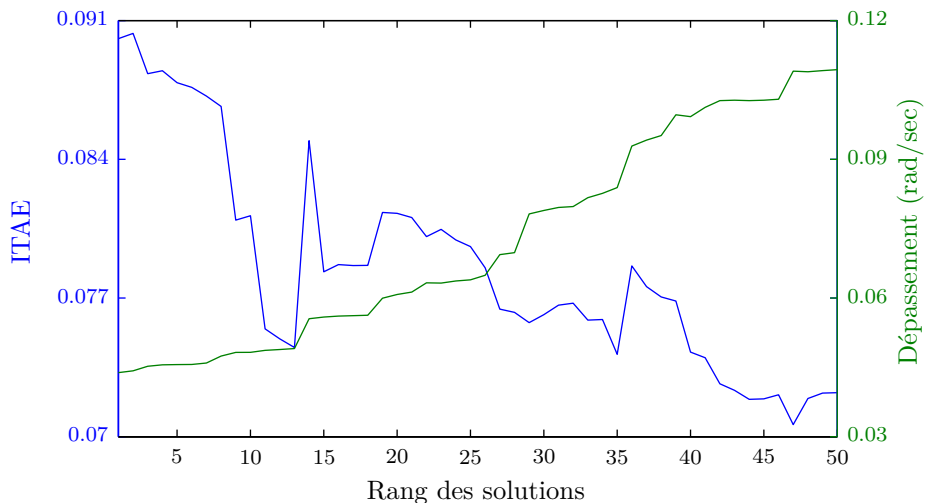


Figure 4.49 : Comparaison des performances des PSS (scénario n°1).

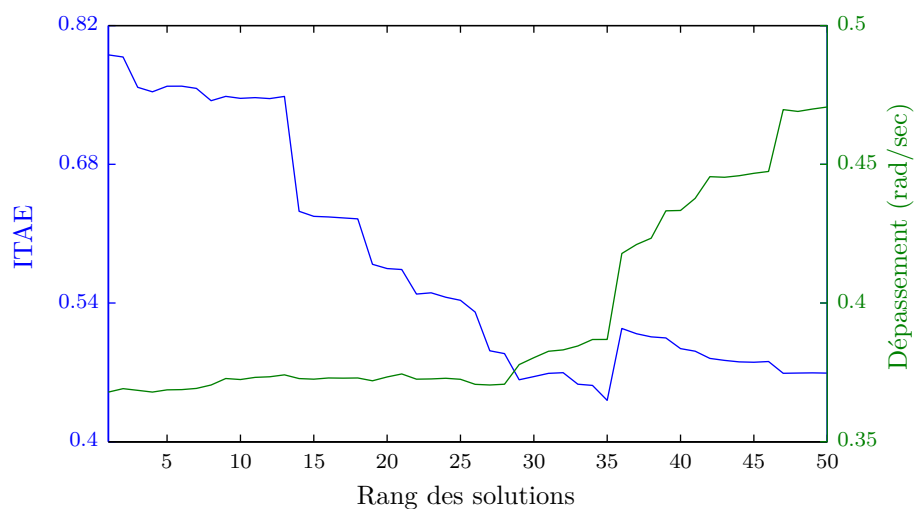


Figure 4.50 : Comparaison des performances des PSS (scénario n°2).

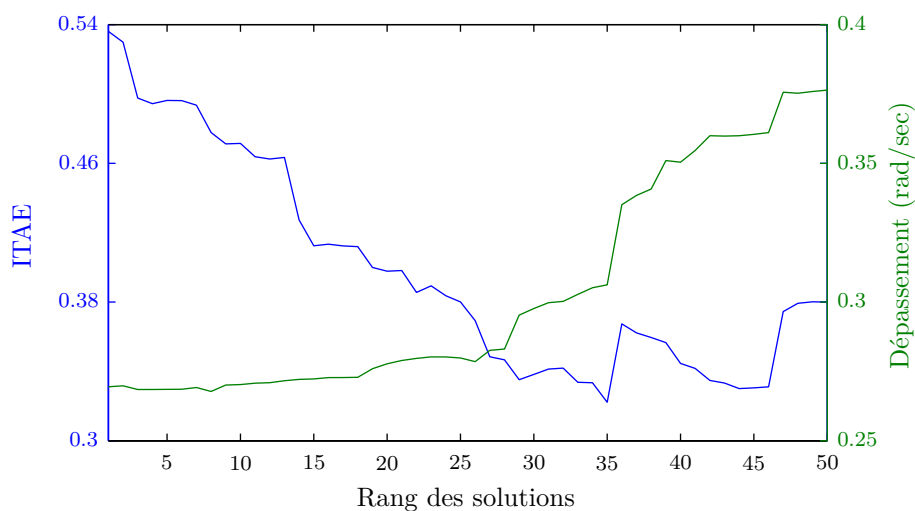


Figure 4.51 : Comparaison des performances des PSS (scénario n°3).

Pour illustrer l'efficacité des solutions obtenues, nous proposons comme solution de compromis celle qui correspond au point  $(f_1, f_2) = (-5.71, -0.94)$ . Cette solution est comparée avec une autre solution obtenue en utilisant une méthode de synthèse conventionnelle. Les paramètres de ces deux solutions sont donnés dans le tableau 4.9.

Tableau 4.9 : Paramètres des PSS sélectionnés pour le réseau à 10 machines.

PSS	Machine #	$K$	$T_1$	$T_2$	$T_3$	$T_4$
Conventionnel	2-10	14.70	0.15	0.05	0.15	0.05
	2	40.76	0.11	0.01	0.05	0.01
Proposé	3	49.24	0.07	0.01	0.06	0.01
	4	41.17	0.11	0.01	0.07	0.01
	5	41.00	0.07	0.01	0.09	0.01
	6	26.53	0.10	0.01	0.11	0.01
	7	47.36	0.06	0.01	0.06	0.01
	8	21.82	0.19	0.01	0.05	0.01
	9	50.76	0.06	0.01	0.06	0.01
	10	36.66	0.08	0.01	0.19	0.01

Les résultats de comparaison des deux solutions précédentes sont montrés dans les figures 4.52 à 4.54 concernant les indicateurs de performance ITSE et ISE. Les simulations sont effectuées en considérant les 3 scénarios précédents et 3 différents points de fonctionnement. Les résultats montrent que la solution proposée permet d'obtenir de meilleurs résultats par rapport à la solution conventionnelle en termes de stabilité et de rapidité de convergence.

Enfin, dans le but d'illustrer les résultats de comparaison, nous présentons dans les figures 4.55 à 4.60, quelques simulations effectuées en considérant les trois scénarios précédents et un point de fonctionnement qui correspond à une forte charge. Les réponses obtenues montrent que la solution proposée permet d'améliorer la stabilité du système par rapport à la solution conventionnelle.

## 4.5 Conclusion

Dans ce chapitre, nous avons proposé plusieurs approches de résolution pour des problèmes de synthèse optimale des PSS dans les réseaux électriques. Les problèmes abordés sont tous multiobjectifs et concernent plusieurs exemples de réseau électrique. La résolution de ces problèmes est effectuée en utilisant les algorithmes d'optimisation multiobjectif abordés dans les chapitres précédents. Les résultats obtenus montrent que l'algorithme MODE-2A permet de résoudre les problèmes considérés d'une façon plus efficace que les autres algorithmes de référence.



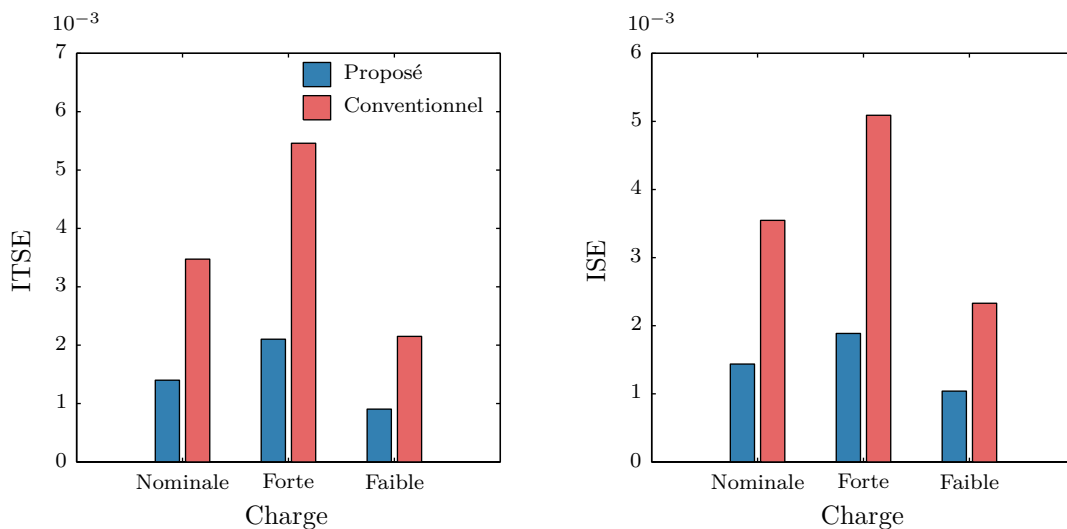


Figure 4.52 : Performances du PSS proposé (scénario n°1).

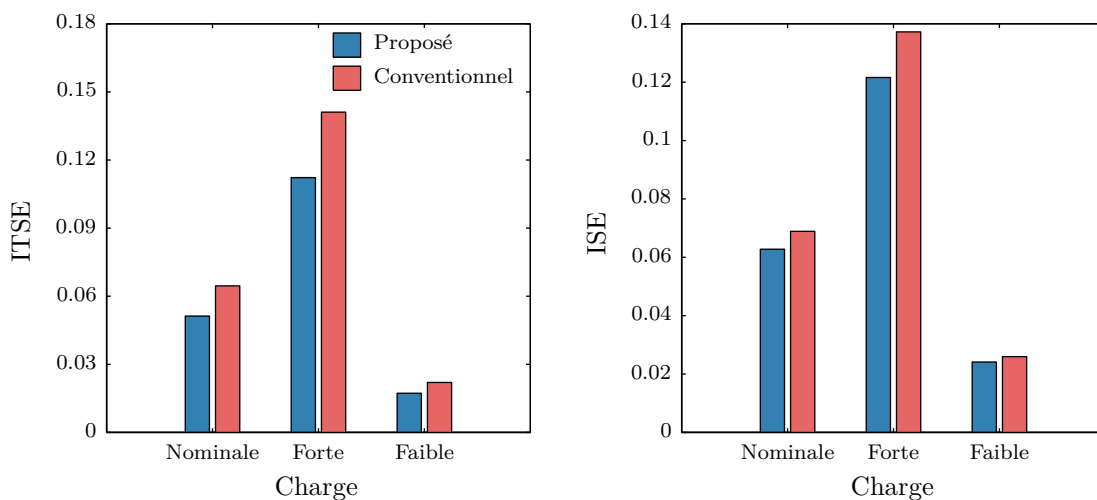


Figure 4.53 : Performances du PSS proposé (scénario n°2).

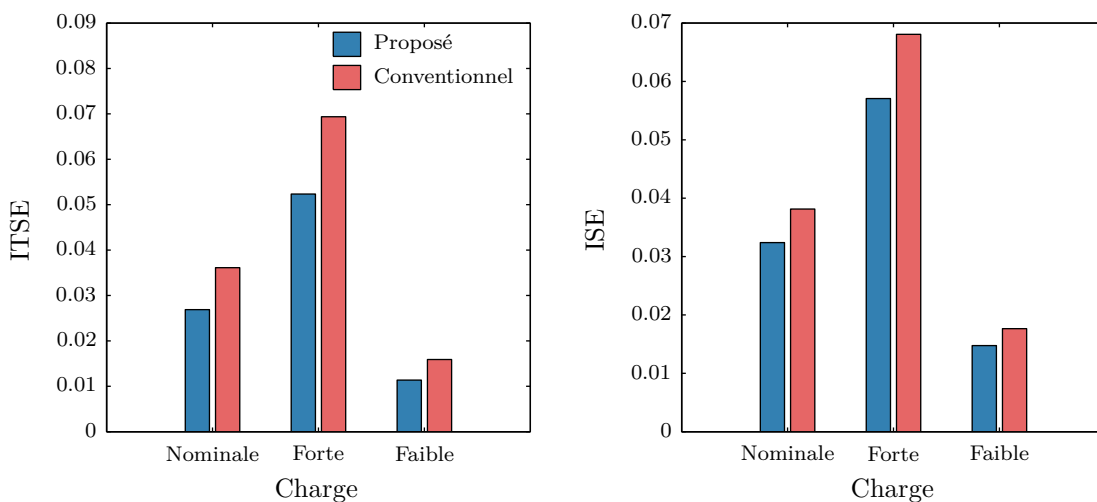


Figure 4.54 : Performances du PSS proposé (scénario n°3).

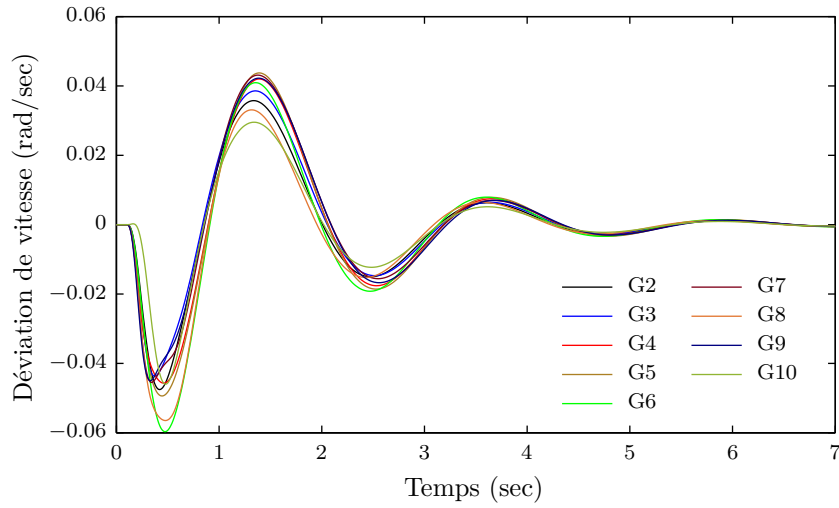


Figure 4.55 : Réponses du système avec le PSS proposé (scénario n°1).

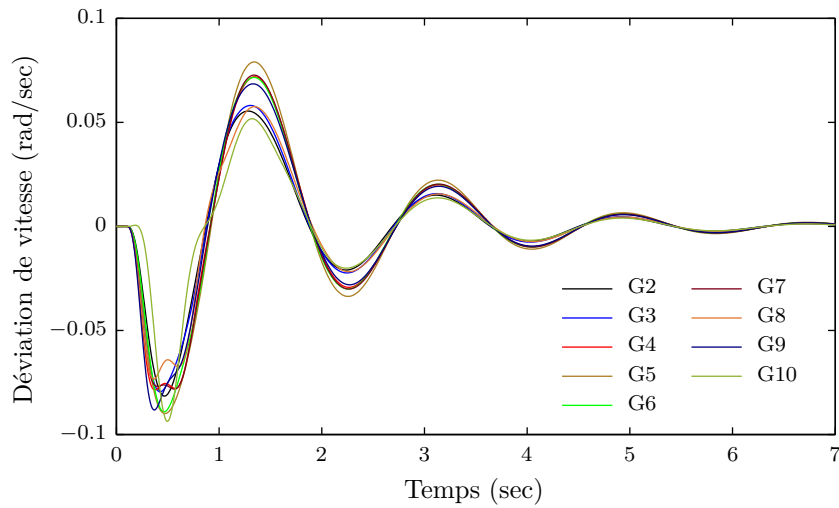


Figure 4.56 : Réponses du système avec le PSS conventionnel (scénario n°1).

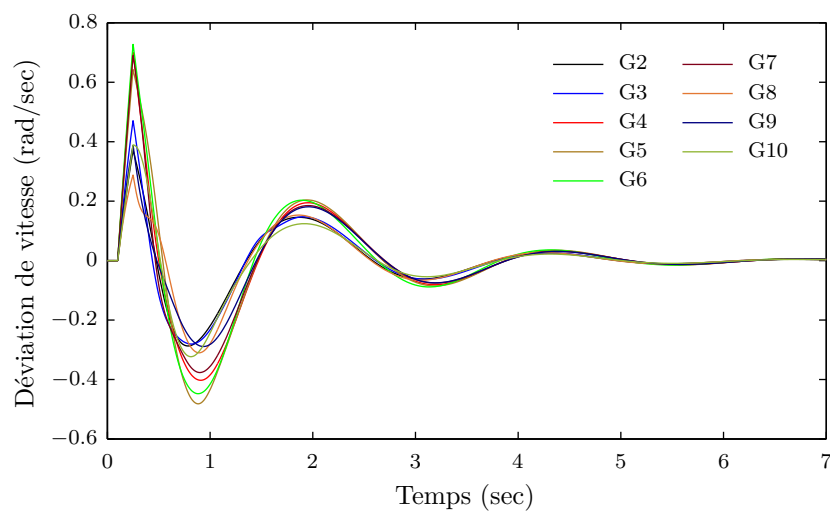


Figure 4.57 : Réponses du système avec le PSS proposé (scénario n°2).

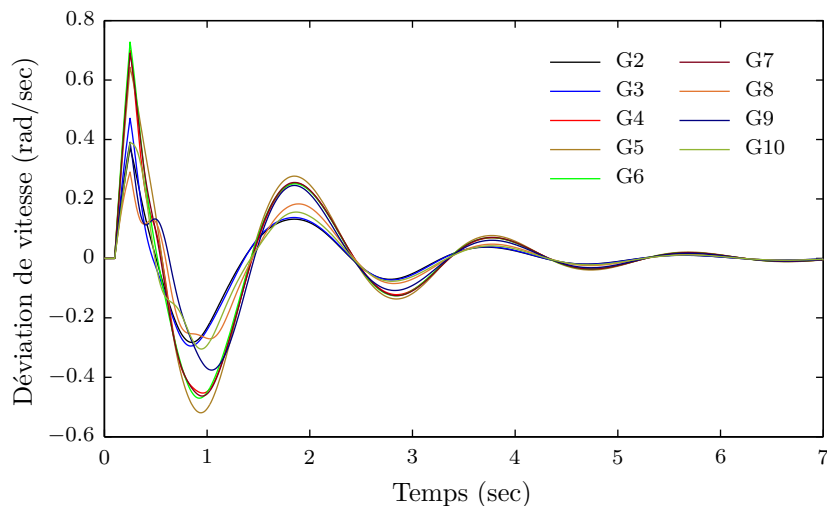


Figure 4.58 : Réponses du système avec le PSS conventionnel (scénario n°2).

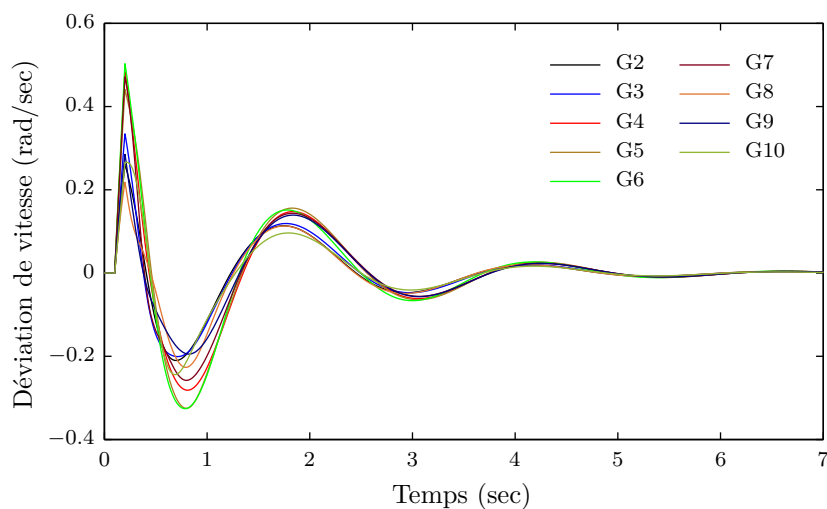


Figure 4.59 : Réponses du système avec le PSS proposé (scénario n°3).

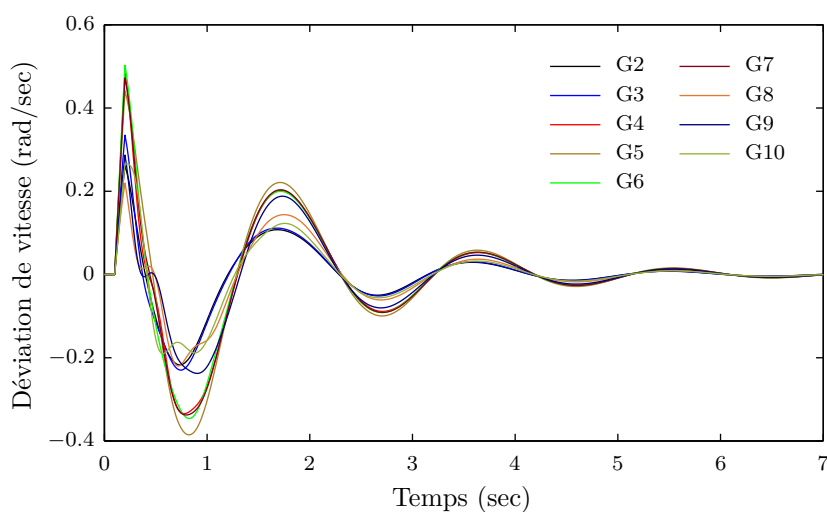


Figure 4.60 : Réponses du système avec le PSS conventionnel (scénario n°3).

## Chapitre 5

# Conclusions et perspectives

# Chapitre 5

## Conclusions et perspectives

### 5.1 Synthèse des travaux

Dans cette thèse, nous avons abordé plusieurs problèmes d'optimisation multiobjectif, concernant la synthèse des PSS dans les réseaux électriques au moyen des métaheuristiques. Ainsi, nous avons posé une problématique qui est en rapport avec deux axes de recherche. Le premier axe est purement théorique et concerne l'amélioration des performances des algorithmes d'optimisation multiobjectif basés sur les métaheuristiques. Le deuxième est d'ordre mathématique, appliqué sur un système réel, et concerne l'utilisation de ces méthodes pour la résolution des problèmes de dimensionnement optimal des PSS dans les réseaux électriques.

Actuellement, les métaheuristiques sont les méthodes de recherche les plus efficaces et les plus utilisées pour résoudre des problèmes d'optimisation difficile notamment les problèmes multiobjectifs. Ces méthodes sont simples à mettre en œuvre et permettent de rechercher des solutions optimales en un temps raisonnable. Ainsi, nous avons choisi d'utiliser quatre algorithmes multiobjectifs considérés comme des méthodes de référence dans la littérature. Ces méthodes sont l'algorithme génétique NSGA-II, l'algorithme à évolution différentielle GDE3, l'algorithme de recuit simulé AMOSA, et l'algorithme d'optimisation par essaims de particules SMPSO.

Dans le but de proposer une méthode de résolution qui est à la fois simple, efficace et robuste, nous avons développé un nouvel algorithme d'optimisation multiobjectif, nommé Multi-Objective Differential Evolution using two Archives ou MODE-2A, qui s'appuie principalement sur la technique de l'évolution différentielle. L'algorithme MODE-2A emploie trois ensembles de solutions ; l'ensemble principal ou la population courante, et deux archives pour mémoriser les solutions non dominées découvertes durant la recherche.

La première archive est utilisée dans le but de maintenir la diversité des solutions afin d'éviter les convergences prématurées vers les fronts locaux, et la deuxième pour mémoriser toutes les solutions non dominées découvertes durant la recherche. L'élimination des solutions supplémentaires est effectuée en se basant sur la distance de crowding pour l'archive de diversité et d'une façon aléatoire pour l'archive principale, ainsi le temps de calcul et la complexité de l'algorithme sont réduits. De plus, les deux

paramètres de l'évolution différentielle, c'est-à-dire le poids différentiel et la probabilité de croisement varient de façon aléatoire durant la recherche, ce qui rend la tâche du concepteur plus simple.

L'algorithme MODE-2A est comparé avec les autres algorithmes de référence en considérant la résolution de plusieurs problèmes de test à deux et à trois fonctions objectifs. L'évaluation des performances de ces algorithmes est réalisée en utilisant quatre métriques de performance ; la métrique de convergence  $GD$ , l'hypervolume  $H$ , la métrique proposée  $\bar{d}_{\max}$  et enfin la métrique d'espacement  $S$ .

En résumant les résultats de ces métriques, chacun des algorithmes utilisés est classé vis-à-vis des autres algorithmes pour chaque problème de test. Les résultats de cette comparaison sont récapitulés dans le tableau 5.1. Nous remarquons que dans le classement général l'algorithme MODE-2A est premier, suivi de AMOSA, GDE3, SMPSO et enfin NSGA-II. Ainsi, nous pouvons conclure que l'algorithme proposé dans cette thèse est capable de résoudre des problèmes académiques d'une manière plus efficace que les autres algorithmes de résolution.

Tableau 5.1 : Classements des algorithmes pour les problèmes de test.

Problème	NSGAI	GDE3	AMOSA	SMPSO	MODE-2A
ZDT1	3	4	2	5	1
ZDT2	4	3	2	5	1
ZDT3	3	4	2	5	1
ZDT4	4	5	2	1	3
ZDT6	3	5	2	4	1
DTLZ12	5	1	2	4	3
DTLZ22	4	5	3	2	1
DTLZ32	5	3	2	1	4
DTLZ42	4	2	5	1	3
DTLZ52	4	5	3	1	2
DTLZ62	4	3	2	5	1
DTLZ72	4	3	2	5	1
DTLZ82	4	3	5	2	1
DTLZ92	3	2	5	4	1
DTLZ13	5	1	2	3	4
DTLZ23	2	4	3	5	1
DTLZ33	5	1	2	4	3
DTLZ43	4	2	5	3	1
DTLZ53	4	5	2	1	3
DTLZ63	5	3	1	4	2
DTLZ73	4	3	2	5	1
DTLZ83	3	2	4	5	1
DTLZ93	3	2	5	4	1
Total	90	69	68	75	40

La deuxième partie de ce travail est consacrée à la résolution de plusieurs problèmes d'optimisation multiobjectif concernant la synthèse des PSS dans les réseaux électriques. Ainsi, nous avons proposé plusieurs approches pour le dimensionnement optimal des PSS, pour un réseau mono-machine et pour deux autres réseaux multimachines. Les fonctions objectifs sont définies au moyen des normes  $H_\infty$  et  $H_2$  ainsi que les valeurs propres du système en boucle fermée.

Concernant le réseau mono-machine, le problème est défini à l'aide de deux fonctions objectifs dans le but d'optimiser la robustesse et la rapidité de convergence du système en boucle fermée. Dans le cas du réseau à 3 machines, l'objectif est d'optimiser séparément les performances de deux machines du réseau ainsi que l'amortissement global du système. Enfin, pour le réseau à 10 machines, deux fonctions objectifs sont définies pour améliorer l'amortissement global et la rapidité de convergence du système.

Les résultats de comparaison pour ces trois problèmes sont récapitulés dans le tableau 5.2. Nous constatons que l'algorithme MODE-2A est toujours premier, ce qui prouve son efficacité à résoudre des problèmes réels. Toutefois, nous remarquons que l'algorithme NSGA-II présente, pour ces problèmes, les deuxièmes meilleures performances après MODE-2A, sachant qu'il est classé dernier pour les problèmes académiques. Ce résultat montre que les performances d'un algorithme quelconque dépendent sensiblement du problème traité et que les résultats des problèmes académiques ne suffisent pas pour conclure sur l'efficacité d'un algorithme.

Tableau 5.2 : Classements des algorithmes pour les problèmes de dimensionnement des PSS.

Réseau	NSGAI	GDE3	AMOS	SMPSO	MODE-2A
Mono-machine	2	5	3	4	1
À 3 machines	2	5	3	4	1
À 10 machines	3	5	4	1	2
Total	7	15	10	9	4

## 5.2 Perspectives

Les perspectives de recherche que nous proposons portent sur les méthodes d'optimisation multiobjectif, et d'autre part sur le dimensionnement optimal des PSS. Ces perspectives sont résumées dans ce qui suit.

**Amélioration de l'algorithme MODE-2A :** les deux paramètres  $CR$  et  $F$  jouent un rôle important dans les performances d'un algorithme à évolution différentielle. Ces deux paramètres sont considérés aléatoires dans MODE-2A, mais peuvent être fixés a priori ou adaptés selon l'évolution de la recherche.

**Utilisation d'autres métaheuristiques :** nous envisageons d'utiliser d'autres métaheuristiques performantes telles que l'optimisation par colonies de fourmis, les systèmes immunitaires artificiels et les algorithmes à estimation de distribution, ainsi que d'autres métaheuristiques plus récentes inspirées généralement de la nature. Par ailleurs, nous proposons de concevoir des algorithmes hybrides afin d'améliorer les performances des méthodes de résolution standard.

**Conception de nouveaux schémas de commande :** d'autres types de commande peuvent être testés tels le régulateur PID standard et d'ordre fractionnaire, la commande par retour d'état, la commande par mode glissant et la commande par logique floue. Par ailleurs, de nouvelles formulations des fonctions objectifs peuvent être envisagées afin de prendre en considération d'autres aspects de la commande tels que la variation du point de fonctionnement et des paramètres du réseau, ainsi que certains aspects concernant les signaux de commande (énergie, variance, etc.).

**Optimisation de la stabilité transitoire des réseaux électriques :** nous proposons également d'aborder le problème d'optimisation de la stabilité transitoire des réseaux électriques, en formulant des fonctions objectifs qui tiennent compte de la dynamique non linéaire des réseaux en réaction à des perturbations sévères telles que les courts-circuits ou les fortes variations paramétriques.

**Application de l'algorithme MODE-2A pour d'autres types de problèmes :** nous proposons enfin d'exploiter la partie théorique de ce travail pour d'autres types de problèmes, afin de valider les performances de l'algorithme proposé.



# Bibliographie

- [1] LIU, G.P. and YANG, J.B. and WHIDBORNE, J.F. *Multiobjective Optimisation and Control*. Hertfordshire : Research Studies Press Ltd., 2003. 336 p. ISBN 978-0863802645.
- [2] COLLETTE, Y. and SIARRY, P. *Multiobjective Optimization*. Berlin : Springer-Verlag, 2004. 293 p. ISBN 978-3540401827.
- [3] DEB, K. *Multi-Objective Optimization using Evolutionary Algorithms*. New York : John Wiley & Sons, Inc, 2001. 518 p. ISBN 978-0471873396.
- [4] FONSECA, C.M. *Multiobjective Genetic Algorithms with Application to Control Engineering Problems*. PhD thesis : Automatic Control and Systems Engineering. Sheffield : The University of Sheffield, 1995. 182 p.
- [5] SIARRY, P. (Ed). *Metaheuristics*. Basel : Springer International Publishing AG, 2016. 489 p. ISBN 978-3319454016.
- [6] KUNDUR, P. *Power System Stability and Control*. New York : McGraw-Hill, Inc., 1994. 1176 p. ISBN 978-0070359581.
- [7] ALKHATIB, H. *Etude de la Stabilité aux Petites Perturbations dans les Grands Réseaux Electriques : Optimisation de la Régulation par une Méthode Métaheuristique*. Thèse de Doctorat : Génie Electrique. Marseille : Université Paul Cezanne d'Aix-Marseille, 2008. 205 p.
- [8] KWANG, Y.L. and EL-SHARKAWI, A. *Modern Heuristic Optimization Techniques : Theory and Applications to Power Systems*. New Jersey : John Wiley & Sons, Inc., 2008. 624 p. ISBN 978-0471457114.
- [9] DUBEY, M. and GUPTA, P. Design of genetic-algorithm based robust power system stabilizer. *International Journal of Computational Intelligence*, 2005, vol. 2, n°1, p. 48-52.
- [10] PANDA, S. and ARDIL, C. Real-coded genetic algorithm for robust power system stabilizer design. *International Journal of Electrical, Computer, and Systems Engineering*, 2008, vol. 2, p. 6–13.
- [11] DAS, T.K. and VENAYAGAMOORTHY, G.K. and ALIYU, O.A. Bio-Inspired algorithms for the design of multiple optimal power system stabilizers : SPPSO and BFA. *IEEE Transactions on Industry Applications*, 2008, vol. 44, n°5, p. 1445—1457.
- [12] PANDA, S. and PADHY, N.P. Robust power system stabilizer design using particle swarm optimization technique. *World Academy of Science, Engineering and Technology*, 2008, vol. 46, p. 776–783.

- [13] PANDA, S. Robust coordinated design of multiple and multi-type damping controller using differential evolution algorithm. *International Journal of Electrical Power & Energy Systems*, 2011, vol. 33, n°4, p. 1018–1030.
- [14] SHAYEGHI, H. and SHAYANFAR, H.A. and GHASEMI, A. A robust ABC based PSS design for a SMIB power system. *International Journal on Technical and Physical Problems of Engineering*, 2011, vol. 3, p. 86–92.
- [15] ABDELAZIZ, A.Y. and ALI, E.S. Cuckoo Search algorithm based load frequency controller design for nonlinear interconnected power system. *Electrical Power and Energy Systems*, 2015, vol. 73, p. 632–643.
- [16] FARAH, A. and GUESMI, T. and HADJ ABDALLAH, H. and OUALI, A. A novel chaotic teaching–learning-based optimization algorithm for multi-machine power system stabilizers design problem. *International Journal of Electrical Power & Energy Systems*, 2016, vol. 77, p. 197–209.
- [17] KALIFULLAH, A. and SANKARAN, P. Robust design of power system stabilizer using harmony search algorithm. *Automatika*, 2014, vol. 55, n°2, p. 162–169.
- [18] ABDEL-MAGID, Y.L. and ABIDO, M.A. and AL-BAIYAT, S. and MANTAWY, A.H. Simultaneous stabilization of multimachine power systems via genetic algorithms. *IEEE Transactions on Power Systems*, 1999, vol. 14, n°4, p. 1428–1439.
- [19] NARESH, G. and RAMALINGA RAJU, M. and NARASIMHAM, S.V.L. and RAVINDRA, K. Optimal design of multimachine power system stabilizer using genetic algorithm. *Innovative Systems Design and Engineering*, 2011, vol. 2, n°4, p. 138–153.
- [20] ABIDO, M.A. Optimal design of power system stabilizers using particle swarm optimization. *IEEE Transactions on Energy Conversion*, 2002, vol. 17, n°3, p. 406–413.
- [21] SHAYEGHI, H. and SHAYANFAR, H.A. and SAFARI, A. and AGHMASHEH, R. A robust PSSs design using PSO in a multi-machine environment. *Energy Conversion and Management*, 2010, vol. 51, n°4, p. 696–702.
- [22] ISLAM, N.N. and HANNAN, M.A. and SHAREEF, H. and MOHAMED, A. and SALAM, M.A. Comparative study of popular objective functions for damping power system oscillations in multimachine system. *The Scientific World Journal*, 2014, vol. 2014, p. 1–8.
- [23] WANG, Z. and CHUNG, C.Y. and WONG, K.P. and TSE, C.T. Robust power system stabiliser design under multi-operating conditions using differential evolution. *IET Generation, Transmission & Distribution*, 2008, vol. 2, n°5, p. 690–700.
- [24] NARESH, G. and RAMALINGA RAJU, M. and NARASIMHAM, S.V.L. and RAVINDRA, K. Optimal multiobjective design of power system stabilizers using simulated annealing. *International Journal of Advances in Science and Technology*, 2011, vol. 3, n°1, p. 37–48.
- [25] ABIDO, M.A. and ABDEL-MAGID, Y.L. Eigenvalue assignments in multimachine power systems using tabu search algorithm. *Computers and Electrical Engineering*, 2002, vol. 28, n°6, p. 527–545.

- [26] ABIDO, M.A. Robust design of multimachine power system stabilizers using simulated annealing. °IEEE Transactions on Energy Conversion, 2000, vol. 15, n°3, p. 297–304.
- [27] LINDA, M.M. and NAIR, N.K. Optimal design of multi-machine power system stabilizer using robust ant colony optimization technique. *Transactions of the Institute of Measurement and Control*, 2011, vol. 34, n°7, p. 829–840.
- [28] FARSANGI, M.M. and KYANZADEH, S. and HAIDARI, S. and NEZAMABADI-POUR, H. Coordinated control of low-frequency oscillations using real immune algorithm with population management. *Energy Conversion and Management*, 2010, vol. 51, n°2, p. 271–276.
- [29] ALI, E.S. Optimization of power system stabilizers using BAT search algorithm. *International Journal of Electrical Power & Energy Systems*, 2014, vol. 61, p. 683–690.
- [30] ALSHAMMARI, B.M. Optimal design of multimachine power system stabilizers using Gbest-guided artificial bee colony algorithm. *International Journal of Engineering Research & Technology*, 2016, vol. 5, n°6, p. 330–338.
- [31] MISHRA, S. and TRIPATHY, M. and NANDA, J. Multi-machine power system stabilizer design by rule based bacteria foraging. *Electric Power Systems Research*, 2007, vol. 77, n°12, p. 1595–1607.
- [32] KHODABAKHSHIAN, A. and HEMMATI, R. Multi-machine power system stabilizer design by using cultural algorithms. *International Journal of Electrical Power & Energy Systems*, 2013, vol. 44, n°1, p. 571–580.
- [33] DEY, P. and BHATTACHARYA, A. and DAS, P. Tuning of power system stabilizer for small signal stability improvement of interconnected power system. *Applied Computing and Informatics*, 2017, in press.
- [34] SHAYEGHI, H. and SHAYANFAR, H.A. and JALILZADEH, S. and SAFARI, A. Multi-machine power system stabilizers design using chaotic optimization algorithm. *Energy Conversion and Management*, 2010, vol. 51, n°7, p. 1572–1580.
- [35] CHITARA, D. and NIAZI, K.R. and SWARNKAR, A. and GUPTA, N. Cuckoo search optimization algorithm for designing of multi-machine power system stabilizer. *IEEE Transactions on Industry Applications*, 2018, vol. 54, n°4, p. 3056–3065.
- [36] SINGH, M. and PATEL, R.N. and NEEMA, D.D. Robust tuning of excitation controller for stability enhancement using multi-objective metaheuristic Firefly algorithm. *Swarm and Evolutionary Computation*, 2018, in press.
- [37] EKINCI, S. and DEMIROREN, A. and HEKIMOGLU, B. Parameter optimization of power system stabilizers via kidney-inspired algorithm. *Transactions of the Institute of Measurement and Control*, 2018, in press.
- [38] AMELI, A. and FARROKHIFARD, M. and AHMADIFAR, A. and SAFARI, A. and SHAYANFAR, H.A. Optimal tuning of Power System Stabilizers in a multi-machine system using firefly algorithm. *12th International Conference on Environment and*

- Electrical Engineering*, Wroclaw, 5–8 May 2013.
- [39] SHAKARAMI, M.R. and DAVOUDKHANI, I.F. Wide-area power system stabilizer design based on Grey Wolf Optimization algorithm considering the time delay. *Electric Power Systems Research*, 2016, vol. 133, p. 149–159.
- [40] PANDA, S. and MOHANTYB, B. and HOTA, P.K. Hybrid BFOA–PSO algorithm for automatic generation control of linear and nonlinear interconnected power systems. *Electric Power Systems Research*, 2013, vol. 13, n°12, p. 4718–4730.
- [41] DEY, P. and BHATTACHARYA, A. and DAS, P. Tuning of power system stabilizer for small signal stability improvement of interconnected power system. *Applied Computing and Informatics*, 2017, in press.
- [42] PERES, W. and DE OLIVEIRA, E.J. and PASSOS FILHO, J.A. and DA SILVA JUNIOR, I.C. Coordinated tuning of power system stabilizers using bio-inspired algorithms. *International Journal of Electrical Power & Energy Systems*, 2015, vol. 64, p. 419–428.
- [43] ABIDO, M.A. Parameter optimization of multimachine power system stabilizers using genetic local search. *International Journal of Electrical Power & Energy Systems*, 2001, vol. 23, n°8, p. 785–794.
- [44] WANG, S.K. and CHIOU, J.P. and LIU, C.W. Parameters tuning of power system stabilizers using improved ant direction hybrid differential evolution. *International Journal of Electrical Power & Energy Systems*, 2009, vol. 31, n°1, p. 34–42.
- [45] ESLAMI, M. and SHAREE, H. and MOHAMED, A. Power system stabilizer design using hybrid multi-objective particle swarm optimization with chaos. *Journal of Central South University of Technology*, 2011, vol. 18, n°5, p. 1579–1588.
- [46] KHALEGHI, M. and FARSANGI, M.M. and NEZAMABADI-POUR, H. and LEE, K.Y. Pareto-optimal design of damping controllers using modified artificial immune algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 2011, vol. 41, n°2, 240–250.
- [47] ALSHAMMARI, B.M. and GUESMI, T. Optimal design of multimachine power system stabilizers using improved multi-objective particle swarm optimization. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 2016, vol. 10, n°12, p. 1384–1390.
- [48] GUESMI, T. and FARAH, A. and HADJ ABDALLAH, H. and OUALI, A. Robust design of multimachine power system stabilizers based on improved non-dominated sorting genetic algorithms. *Electrical Engineering*, 2018, vol. 100, n°3, p. 1351–1363.
- [49] SEBAA, K. and BOUDOUR, M. Robust power system stabilizers design using multi-objective genetic algorithm. *IEEE Power Engineering Society General Meeting*, 2007, Florida, 24–28 July 2007.
- [50] YASSAMI, H. and DARABIA, A. and RAFIEI, S.M.R. Power system stabilizer design using Strength Pareto multi-objective optimization approach. *Electric Power Systems Research*, 2010, vol. 80, n°7, p. 838–846.

- [51] KHALEGHI, M. and MOHAMMADIAN, M. and KHORASANINEJAD, M. and FARSANGI, M.M. Design of two power system stabilizers coordinately based on strength Pareto evolutionary algorithm. *4th IEEE Conference on Industrial Electronics and Applications (ICIEA 2009)*, Xi'an, 25–27 May 2009.
- [52] GHASEMI, A. and SHAYEGHI, H. and ALKHATIB, H. Robust design of multimachine power system stabilizers using fuzzy gravitational search algorithm. *Electrical Power and Energy Systems*, 2013, vol. 51, p. 190–200.
- [53] PAN, I. and DAS, S. Chaotic multi-objective optimization based design of fractional order  $PI^{\lambda}D^{\mu}$  controller in AVR system. *International Journal of Electrical Power & Energy Systems*, 2012, vol. 43, n°1, p. 393–407.
- [54] PANDA, S. Multi-objective non-dominated shorting genetic algorithm-II for excitation and TCSC-based controller design. *Journal of Electrical Engineering*, 2009, vol. 60, n°2, p. 86–93.
- [55] PASZEK, S. Use of Pareto optimisation for tuning power system stabilizers. *Bulletin of the Polish Academy of Sciences : Technical Sciences*, 2012, vol. 60, n°1, p. 125–131.
- [56] KHODABAKHSHIAN, A. and ESMAILI, M.R. and BORNAPOUR, M. Optimal coordinated design of UPFC and PSS for improving power system performance by using multi-objective water cycle algorithm. *International Journal of Electrical Power & Energy Systems*, 2016, vol. 83, p. 124–133.
- [57] SHAYEGHI, H. and GHASEMI, A. A multi objective vector evaluated improved honey bee mating optimization for optimal and robust design of power system stabilizers. *International Journal of Electrical Power & Energy Systems*, 2014, vol. 62, p. 630–645.
- [58] HASANOVIC, A. and FELIACHI, A. Robust PSS tuning through multiobjective optimization. *IEEE PES Power Systems Conference and Exposition*, New York, 10–13 October 2005.
- [59] PANDA, S. AND YEGIREDDY, N.K. Automatic generation control of multi-area power system using multi-objective non-dominated sorting genetic algorithm-II. *International Journal of Electrical Power & Energy Systems*, 2013, vol. 53, p. 54–63.
- [60] VIVEROS, E.R.C. and TARANTO, G.N. and FALCAO, D.M. Coordinated tuning of AVRs and PSSs by multiobjective genetic algorithms. *Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems*, Arlington, 6–10 November 2006.
- [61] COELLO, C.A.C. Metaheuristics for Multiobjective Optimization. *Tutorial in The IEEE World Congress on Computational Intelligence (IEEE WCCI 2016)*, 2016.
- [62] DEB, K. and PRATAP, A. and AGARWAL, S. and MEYARIVAN, T. A fast and elitist multiobjective genetic algorithm : NSGA-II. *IEEE Transaction on Evolutionary Computation*, 2002, vol. 6, n°2, p. 182–197.
- [63] GOLDBERG, D.E. (Ed). *Genetic Algorithms in Search, Optimization and Machine Learning*. Boston : Addison-Wesley Longman Publishing Co., Inc., 1989. 432 p. ISBN

- 978-0201157673.
- [64] KUKKONEN, S. and LAMPINEN, J. GDE3 : The third evolution step of generalized differential evolution. *IEEE Congress on Evolutionary Computation*, Edinburgh, 2–5 September 2005.
- [65] STORN, R. and PRICE, K. Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 1997, vol. 11, n°4, p. 341–359.
- [66] ENGELBRECHT, A.P. *Computational Intelligence : An Introduction*. Chichester : John Wiley & Sons Ltd., 2007. 628 p. ISBN 978-0470035610.
- [67] PRICE, K.V. and STORN, R.M. and LAMPINEN, J.A. *Differential Evolution : A Practical Approach to Global Optimization*. Berlin : Springer-Verlag, 2005. 539 p. ISBN 978-3642424168.
- [68] BANDYOPADHYAY, S. and SAHA, S. and DEB, K. A simulated annealing-based multiobjective optimization algorithm : AMOSA. *IEEE Transaction on Evolutionary Computation*, 2008, vol. 12, n°3, p. 269–283.
- [69] KIRKPATRICK, S. and GELATT, C.D. and VECCHI, M.P. Optimization by Simulated Annealing. *Science, New Series*, 1983, vol. 220, n°4598, p. 671–680.
- [70] ZITZLER, E. *Evolutionary Algorithms for Multiobjective Optimization : Methods and Applications*, PhD thesis : Computer Engineering. Zurich : Swiss Federal Institute of Technology (ETH), 1999. 132 p.
- [71] NEBRO, A.J. and DURILLO, J.J. and GARCÍA-NIETO, J. and COELLO COELLO, C.A. and LUNA, F. and ALBA, E. SMPSO : A new PSO-based metaheuristic for multi-objective optimization. *IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MCDM 2009)*, Nashville, 30 March–2 April, 2009.
- [72] KENNEDY, J. and EBERHART, R.C. Particle swarm optimization. *Proceedings of the 1995 IEEE International Conference on Neural Networks*, New Jersey, 27 November–1 December, 1995.
- [73] CLERC, M. *Particle Swarm Optimization*. London : ISTE Ltd., 2006. 244 p. ISBN 978-1905209040.
- [74] ZITZLER, E. and LAUMANN, M. and THIELE, L. *SPEA2 : Improving the strength Pareto evolutionary algorithm*. Technical report 103 : Computer Engineering and Networks Laboratory (TIK). Zurich : Swiss Federal Institute of Technology (ETH), 2001.
- [75] DEB, K. and THIELE, L. and LAUMANN, M. and ZITZLER, E. *Scalable test problems for evolutionary multiobjective optimization*. Technical report 2001001 : Kanpur Genetic Algorithms Laboratory (KanGAL). Kanpur : Indian Institute of Technology, 2001.
- [76] VAN VELDHUIZEN, D.A. *Multiobjective Evolutionary Algorithms : Classifications, Analyses, and New Innovations..* PhD thesis : Evolutionary Computation. Ohio : Air Force Institute of Technology, 1999. 270 p.

- [77] SCHOTT, J.R. *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization*. Master's thesis : Aeronautics and astronautics. Cambridge : Massachusetts Institute of Technology, 1995. 201 p.
- [78] LABDELAOUI, H. *Algorithmes Evolutionnaires Multiobjectifs : Application pour la Commande d'un Réseau Electrique Multimachine*. Mémoire de Magister : Automatique. Alger : École Nationale Polytechnique, 2007. 78 p.
- [79] ANDERSON, P.M. and FOUAD, A.A. *Power System Control and Stability*. Iowa : The Iowa State University Press, 1977. 464 p. ISBN 978-0813812458.
- [80] LABDELAOUI, H. and BOUDJEMA, F. and BOUKHETALA, D. Multiobjective optimal design of dual-input power system stabilizer using genetic algorithms. *Revue roumaine des sciences technique Série Électrotechnique et Énergétique* (Thomson Reuters JCR impact factor 2017/2018 : 1.11), 2017, vol. 62, p. 93–97.
- [81] LABDELAOUI, H. and BOUDJEMA, F. and BOUKHETALA, D. Multiobjective mixed  $H_2/H_\infty$  synthesis of power system stabilizers using particle swarm optimization. *4th International Conference on Electrical Engineering (ICEE'12)*, Algiers, 8–10 May 2012.
- [82] SÁNCHEZ-PEÑA, R.S. and SZNAIER, M. *Robust Systems Theory and Applications*. New York : John Wiley & Sons, Inc., 1998. 490 p. ISBN 978-0471176275.
- [83] PAL, B. and CHAUDHURI, B. *Robust Control in Power Systems*. New York : Springer Science+Business Media, Inc., 2005. 190 p. ISBN 978-1441938534.
- [84] ATHAY, T. and PODMORE, R. and VIRMANI, S. A Practical method for the direct analysis of transient stability. *In : IEEE Transactions on Power Apparatus and Systems*, 1979, p. 573–584.
- [85] LABDELAOUI, H. and BOUDJEMA, F. and BOUKHETALA, D. A multiobjective tuning approach of power system stabilizers using particle swarm optimization. *Turkish Journal of Electrical Engineering & Computer Sciences* (Impact Factor 2017/2018 : 0.58), 2016, vol. 24, n°5, p. 3898–3909.
- [86] ABIDO, M.A. *Intelligent Techniques Approach to Power Systems Identification and Control*. PhD thesis : Electrical engineering. Dhahran : King Fahd University of Petroleum and Minerals, 1997. 262 p.

## Annexe A

# Modèle dynamique de la machine

Equations différentielles du modèle [86]

$$\dot{\delta} = \omega - \omega_0 \quad (\text{A.1})$$

$$\dot{\omega} = -\frac{D}{M}(\omega - \omega_0) + \frac{\omega_0}{M}(P_m - P_e) \quad (\text{A.2})$$

$$\dot{E}'_q = \frac{1}{T'_{d0}}(Ef - E'_q - (X_d - X'_d)I_d) \quad (\text{A.3})$$

$$\dot{E}'_d = \frac{1}{T_A}(K_A(V_{réf} - V_t + V_{PSS})) \quad (\text{A.4})$$

$$P_e = E'_q I_q + E'_d I_d + (X'_q - X'_d) I_q I_d \quad (\text{A.5})$$

$$V_t = \sqrt{(E'_d + X'_q I_q)^2 + (E'_q - X'_d I_d)^2} \quad (\text{A.6})$$



**Nomenclature**

$\delta$	Angle de charge
$\omega$	Vitesse rotorique
$\omega_0$	Vitesse de synchronisme
$P_m$	Puissance mécanique
$P_e$	Puissance électrique active
$Q_e$	Puissance électrique réactive
$D$	Constante d'amortissement
$M$	Moment d'inertie
$E'_d$	F.é.m transitoire d'axe direct
$E'_q$	F.é.m transitoire d'axe en quadrature
$E_f$	Tension d'excitation
$T'_{d0}$	Constance de temps en court-circuit d'axe direct
$T'_{q0}$	Constance de temps en court-circuit d'axe en quadrature
$I_d$	Courant d'axe direct
$I_q$	Courant d'axe en quadrature
$X_d$	Réactance d'axe direct
$X_q$	Réactance d'axe en quadrature
$X'_d$	Réactance transitoire d'axe direct
$X'_q$	Réactance transitoire d'axe en quadrature
$V_t$	Tension terminale du générateur
$V_{réf}$	Tension de référence du générateur
$K_A$	Gain de l'AVR
$T_A$	Constante de temps de l'AVR
$V_{PSS}$	Tension de sortie du PSS

## Annexe B

### Données du réseau mono-machine

Tableau B.1 : Paramètres du réseau [79].

Paramètre	Valeur
$X_d$ (p.u)	1.72
$X_q$ (p.u)	1.68
$X'_d$ (p.u)	0.45
$X'_q$ (p.u)	0.59
$T'_{d0}$ (s)	6.3
$T'_{q0}$ (s)	0.43
$H$ (p.u)	4
$D$ (p.u)	0
$X_t$ (p.u)	0.15
$X_e$ (p.u)	0.2
$\omega_0$ (rad/s)	377
$K_A$	50
$T_A$ (s)	0.05
$E_{f \min}$ (p.u)	-10.0
$E_{f \max}$ (p.u)	10.0
$V_{PSS \min}$	-0.2
$V_{PSS \max}$	0.2

Tableau B.2 : Points de fonctionnement.

Charge	$V_t$ (p.u)	$P_e$ (p.u)	$Q_e$ (p.u)
Nominale	1	0.8	0.2
Forte	1	1.0	0.5
Faible	1	0.4	0.1

#### Nomenclature

- $X_t$  Réactance du transformateur  
 $X_e$  Réactance de la ligne

## Annexe C

### Données du réseau à 3 machines

Tableau C.1 : Paramètres des machines [86].

Paramètre	G1	G2	G3
$X_d$ (p.u)	0.1460	0.8958	1.3125
$X_q$ (p.u)	0.0608	0.1198	0.1813
$X'_d$ (p.u)	0.0969	0.8645	1.2578
$X'_q$ (p.u)	0.0969	0.1969	0.2500
$T'_{d0}$ (s)	8.96	6	5.89
$T'_{q0}$ (s)	0.31	0.535	0.6
$H$ (p.u)	23.64	6.4	3.01
$D$ (p.u)	0	0	0
$K_A$	100	100	100
$T_A$ (s)	0.05	0.05	0.05
$E_{f\min}$ (p.u)	-10.0	-10.0	-10.0
$E_{f\max}$ (p.u)	10.0	10.0	10.0
$V_{PSS\min}$	-0.1	-0.1	-0.1
$V_{PSS\max}$	0.1	0.1	0.1

Tableau C.2 : Paramètres des lignes [86].

Noeud de départ	Noeud d'arrivée	$R$	$X$	$B$
1	4	0.0000	0.0576	0.0000
2	7	0.0000	0.0625	0.0000
3	9	0.0000	0.586	0.0000
4	5	0.0100	0.0850	0.1760
4	6	0.0170	0.0920	0.1580
5	7	0.0320	0.1610	0.3060
6	9	0.0390	0.1700	0.3580
7	8	0.0085	0.0720	0.1490
8	9	0.0119	0.1008	0.2090

Tableau C.3 : Points de fonctionnement [86].

Charge	Paramètre	G1	G2	G3
Nominale	$P_e$ (p.u)	0.7164	1.6300	0.8500
	$Q_e$ (p.u)	0.2705	0.0665	-0.1086
Forte	$P_e$ (p.u)	2.2073	1.9200	1.2800
	$Q_e$ (p.u)	1.0879	0.5635	0.3588
Faible	$P_e$ (p.u)	0.3623	0.8000	0.4500
	$Q_e$ (p.u)	0.1620	-0.1085	-0.2042
	$V_t$ (p.u)	1.0400	1.0250	1.0250

Tableau C.4 : Charges du réseau [86].

Charge	Noeud 5		Noeud 6		Noeud 8	
Nominale	1.25	0.50	0.90	0.30	1.00	0.35
Forte	2.00	0.80	1.80	0.60	1.50	0.60
Faible	0.65	0.55	0.45	0.35	0.50	0.25

### Nomenclature

- $R$  Résistance de la ligne
- $X$  Réactance de la ligne
- $B$  Susceptance de la ligne



Tableau D.2 : Paramètres des lignes [86].

Noeud de départ	Noeud d'arrivée	$R$	$X$	$B$	Rapport de transformation
1	2	0.0035	0.0411	0.6987	-
1	39	0.0010	0.0250	0.7500	-
2	3	0.0013	0.0151	0.2572	-
2	25	0.0070	0.0086	0.1460	-
3	4	0.0013	0.0213	0.2214	-
3	18	0.0011	0.0133	0.2138	-
4	5	0.0008	0.0128	0.1342	-
4	14	0.0008	0.0129	0.1382	-
5	6	0.0002	0.0026	0.0434	-
5	8	0.0008	0.0112	0.1476	-
6	7	0.0006	0.0092	0.1130	-
6	11	0.0007	0.0082	0.1389	-
7	8	0.0004	0.0046	0.0780	-
8	9	0.0023	0.0363	0.3804	-
9	39	0.0010	0.0250	1.2000	-
10	11	0.0004	0.0043	0.0729	-
10	13	0.0004	0.0043	0.0729	-
13	14	0.0009	0.0101	0.1723	-
14	15	0.0018	0.0217	0.3660	-
15	16	0.0009	0.0094	0.1710	-
16	17	0.0007	0.0089	0.1342	-
16	19	0.0016	0.0195	0.3040	-
16	21	0.0008	0.0135	0.2548	-
16	24	0.0003	0.0059	0.0680	-
17	18	0.0007	0.0082	0.1319	-
17	27	0.0013	0.0173	0.3216	-
21	22	0.0008	0.0140	0.2565	-
22	23	0.0006	0.0096	0.1846	-
23	24	0.0022	0.0350	0.3610	-
25	26	0.0032	0.0323	0.5130	-
26	27	0.0014	0.0147	0.2396	-
26	28	0.0043	0.0474	0.7802	-
26	29	0.0057	0.0625	1.0290	-
28	29	0.0014	0.0151	0.2490	-
12	11	0.0016	0.0435	0.0000	1.006
12	13	0.0016	0.0435	0.0000	1.006
6	31	0.0000	0.0250	0.0000	1.070
10	32	0.0000	0.0200	0.0000	1.070
19	33	0.0007	0.0142	0.0000	1.070
20	34	0.0009	0.0180	0.0000	1.009
22	35	0.0000	0.0143	0.0000	1.025
23	36	0.0005	0.0272	0.0000	1.000
25	37	0.0006	0.0232	0.0000	1.025
2	30	0.0000	0.0181	0.0000	1.025
29	38	0.0008	0.0156	0.0000	1.025
19	20	0.0007	0.0138	0.0000	1.060

Tableau D.3 : Points de fonctionnement [86].

Machine	Nominale		Forte		Faible	
	$P_e$ (p.u)	$Q_e$ (p.u)	$P_e$ (p.u)	$Q_e$ (p.u)	$P_e$ (p.u)	$Q_e$ (p.u)
G1	10.0000	0.8828	13.2007	2.5186	6.9051	-0.6062
G2	5.2081	1.9825	6.7705	3.4072	3.6457	0.8518
G3	6.5000	2.0514	8.4500	3.6511	4.5500	0.8028
G4	6.3200	1.0991	8.2160	2.2116	4.4240	0.2462
G5	5.0800	1.6576	6.6040	2.4670	3.5560	1.0140
G6	6.5000	2.1241	8.4500	3.6078	4.5500	0.9831
G7	5.6000	1.0117	7.2800	2.0241	3.9200	0.2462
G8	5.4000	0.0044	7.0200	0.7559	3.7800	-0.5563
G9	8.3000	0.2284	10.7900	1.5459	5.8100	-0.7545
G10	2.5000	1.4616	3.2500	2.3888	1.7500	0.7067

Tableau D.4 : Charges du réseau [86].

Charge	Nominale		Forte		Faible	
	$P_e$ (p.u)	$Q_e$ (p.u)	$P_e$ (p.u)	$Q_e$ (p.u)	$P_e$ (p.u)	$Q_e$ (p.u)
Noeud 3	3.2200	0.0240	4.1860	0.0312	2.2540	0.0168
Noeud 4	5.0000	1.8400	6.5000	2.3920	3.5000	1.2880
Noeud 7	2.3380	0.8400	3.0394	1.0920	1.6366	0.5880
Noeud 8	5.2200	1.7600	6.7860	2.2880	3.6540	1.2320
Noeud 12	0.0750	0.8800	0.0975	1.1440	0.0525	0.6160
Noeud 15	3.2000	1.5300	4.1600	1.9890	2.2400	1.0710
Noeud 16	3.2900	0.3230	4.2770	0.4199	2.3030	0.2261
Noeud 18	1.5800	0.3000	2.0540	0.3900	1.1060	0.2100
Noeud 20	6.2800	1.0300	8.1640	1.3390	4.3960	0.7210
Noeud 21	2.7400	1.1500	3.5620	1.4950	1.9180	0.8050
Noeud 23	2.4750	0.8460	3.2175	1.0998	1.7325	0.5922
Noeud 24	3.0860	-0.9220	4.0118	-1.1986	2.1602	-0.6454
Noeud 25	2.2400	0.4720	2.9120	0.6136	1.5680	0.3304
Noeud 26	1.3900	0.1700	1.8070	0.2210	0.9730	0.1190
Noeud 27	2.8100	0.7550	3.6530	0.9815	1.9670	0.5285
Noeud 28	2.0600	0.2760	2.6780	0.3588	1.4420	0.1932
Noeud 29	2.8350	0.2690	3.6855	0.3497	1.9845	0.1883
Noeud 31	0.0920	0.0460	0.1196	0.0598	0.0644	0.0322
Noeud 39	11.040	2.5000	14.352	3.2500	7.7280	1.7500