

Ministère de l'Enseignement Supérieur et de
la Recherche Scientifique
Université Amar Telidji
Laghouat
Faculté de Technologie



وزارة التعليم العالي والبحث العلمي
جامعة عمار تليجي
الأغواط
كلية التكنولوجيا

Département d'électrotechnique

Thèse rédigée par
Ridha Djamel MOHAMMEDI
Grade de Magister (Université de Laghouat)

Pour l'obtention du Diplôme de Doctorat en Génie Electrique
Option : Réseaux Électriques
--THEME--

Visualisation de l'Information par le Développement d'Interfaces Graphiques pour la Conduite des Réseaux Electriques

Devant le jury

BENALIA Atallah
HELLAL Abdelhafid
ARIF Salem
BOUDOUR Mohamed
BOUKTIR Tarek
CHETTIH Saliha

Prof. à l'Université de Laghouat
Prof. à l'E.N.P. Alger
Prof. à l'Université de Laghouat
Prof. à l'U.S.T.H.B. Alger
Prof. à l'Université de Sétif
MCA. à l'Université de Laghouat

Président
Rapporteur
Rapporteur
Examineur
Examineur
Examineur

ملخص

الهدف من هذه الأطروحة هو تطوير برنامج متين من أجل تصوير، محاكاة وإيجاد الحلول المثلى للشبكات الكهربائية بحيث يكون هذا البرنامج مرناً، قابل للتطوير وسهل الصيانة باستخدام تقنيات البرمجة الشيئية الموجهة بحيث تكون الشبكات منمذجة على شكل مجموعة من الأشياء المنظمة والمهيكلية على شكل تسلسل هرمي من الأصناف. هذه الهيكلية من الأصناف ليست من أجل تمثيل عناصر الشبكة الكهربائية فحسب بل أيضاً لتمثيل التطبيقات المتعلقة بالشبكات الكهربائية. ثلاث تطبيقات تم إنشاؤها وبرمجتها باستخدام لغة البرمجة دلفي. هذه التطبيقات هي الجدولة المثالية، الإدراج الأمثل للطاقة اللامركزية في شبكة التوزيع من حيث الموقع والحجم وإعادة تشكيل شبكة التوزيع. تم إضافة واجهة المستخدم الرسومية وقاعدة البيانات إلى البرنامج الذي تم إنشاؤه من أجل محاكاة الشبكات الكهربائية.

الكلمات المفتاحية

البرمجة الشيئية الموجهة، واجهة المستخدم الرسومية، الجدولة المثالية، الطاقة اللامركزية، إعادة تشكيل شبكة التوزيع

Résumé

L'objectif de cette thèse est le développement d'un logiciel robuste pour visualisation, la simulation et l'optimisation des réseaux électriques qui est en particulier flexible, extensible et facile à maintenir en utilisant les avantages de la méthodologie orienté-objet où le réseau électrique est modélisé comme un ensemble d'objets liés et structurés dans une hiérarchie de classes. Cette structure de classes est conçue non seulement pour représenter les éléments physiques du système, mais également les concepts abstraits telles que les fonctions d'analyse des réseaux électriques (applications). Trois applications sont conçues et implémentées en langage de programmation Delphi. Il s'agit de l'Engagement de Turbines (Unit Commitment), l'optimisation de l'emplacement et de la taille des générateurs dispersés (DGs) dans les réseaux de distribution, et la reconfiguration optimale des réseaux de distributions, la GUI avec l'éditeur graphique et la base de données visuelle développés forment un prototype de simulation des réseaux électriques.

Mots-clés

Programmation Orientée Objet, GUI, Engagement de Turbines, Générateurs Dispersés, Reconfiguration Optimale des Réseaux de Distribution

Abstract

The objective of this thesis is the development of a robust software for visualization, simulation, and optimization of electrical networks which is particularly flexible, extensible and easy to maintain using the benefits of the object-oriented methodology where the electrical network is modeled as a set of linked objects and structures in a hierarchy of classes. This class structure is suited not only to represent physical elements that form the system, but also abstract concepts, such as analysis and synthesis methodologies applied to power system (applications). Three applications are designed and implemented in the programming language Delphi. These applications are: Unit Commitment, Optimal allocation and sizing of dispersed generators (DGs) in the distribution networks, and the optimal reconfiguration of networks of distributions, the GUI with the graphical editor and the visual data base developed form a prototype of a simulation tool for power system analysis.

Key-words

Object-Oriented Programming, GUI, Unit Commitment, Dispersed Generation, Reconfiguration of Distribution network

Remerciements

Je tiens à exprimer mes plus vifs remerciements à mon encadreur monsieur Abdelhafid HELLAL, Professeur à l'Ecole Nationale Polytechnique d'Alger, de m'avoir proposé le sujet de cette thèse et en me faisant profiter de ses conseils et ses grandes compétences dans le domaine des réseaux électriques.

Mes remerciements les plus vifs, vont à mon Co-encadreur Monsieur Salem ARIF, Professeur à l'université Laghouat, pour le suivi et les conseils pertinents apportés durant la réalisation de ce travail.

Je remercie également le Prof. A. BENALIA (président de jury) et les membres du jury le Prof. M. BOUDOUR, le Prof. T. BOUKTIR, et DR. S. CHETTIH de l'intérêt dont ils font preuve à mon égard pour lire ce mémoire et en assistant à ma soutenance.

Enfin, je voudrais exprimer mes profonds remerciements à mes parents et à toute ma famille par leur soutien permanent durant les années que consacrées à la réalisation de cette thèse.

PUBLICATIONS SCIENTIFIQUES

PUBLICATIONS

- [1] R.D. Mohammedi, A. Hellal, S. Arif, M. Mosbah, "Optimal DG Placement and Sizing in Radial Distribution Systems Using NSGA-II for Power Loss Minimization and Voltage Stability Enhancement", (2013) International Review of Electrical Engineering (IREE), 8(6), pp. 1806-1814.
- [2] R.D. Mohammedi, A. Hellal, S. Arif, "An Object-Oriented Programming Approach to Security-Constrained Unit Commitment Problem", (2014) International Review of Electrical Engineering (IREE), 9(6), pp. 1180-1189.
- [3] S. Arif, R. D. Mohammedi, A. Hellal, A Memory Simulated Annealing Method to the Unit Commitment Problem with Ramp Constraints , Arabian Journal for Science and Engineering , June 2012, Volume 37, Issue 4, pp 1021-1031.

CONFÉRENCES INTERNATIONALES

- [4] A. Kouzou, R.D. Mohammedi, and A. Hellal, "An efficient biogeography-based optimization algorithm for smart radial distribution power system reconfiguration", First Workshop on Smart Grid and Renewable Energy (SGRE), 2015, March 23-24, Doha Qatar.
- [5] R.D. Mohammedi, A. Kouzou, "Distribution Network Reconfiguration Using Cuckoo Search Based Optimization", Fourth international conference on Electrical Engineering 2015, December 13-15, 2015, Boumedes, Algeria.
- [6] R.D. Mohammedi, A. Kouzou, "Optimal Reconfiguration of a Radial Power Distribution Network based on Metaheuristic Optimization Algorithms", The Fourth International Conference on Electric Power and Energy Conversion Systems (EPECS 2015), American University of Sharjah, 2015, 24-26 November, Sharjah, United Arab Emirates.

- [7] A. Kouzou, R.D. Mohammedi, "Distribution Network Reconfiguration Using Biogeography-Based Optimization", International Conference Multiphysics Modeling Simulation for Systems Design MMSSD 2014, December 17-19, 2014 Sousse, Tunisia.
- [8] R.D. Mohammedi , A. Hellal, M. Mosbah, S. Arif, " An Efficient BBO Algorithm for Optimal Allocation and Sizing of Shunt Capacitors in Radial Distribution Networks", Fourth international conference on Electrical Engineering 2015, December 13-15, 2015, Boumedes, Algeria.
- [9] M. Mosbah, S. Arif, R.D. Mohammedi, A. khattara, "Optimal Sizing and Placement of Distributed Generation in Transmission System", 4th International Conference on Renewable Energy: Generation and Applications, February 08-10, 2016, Belfort, France.
- [10] R. D. Mohammedi, S. Arif, A. Hellal, " La Résolution du problème d’engagement des turbines : Unit Commitment par la méthode hybride RT/RS", 6th International Conference on Electrical Engineering, 11-13 Octobre, 2010, Batna, Algérie.
- [11] R. D. Mohammedi, S. Arif, A. Hellal, "Application de la méthode hybride RT/RS au problème d’engagement des turbines "Unit Commitment", The International Conference on Electrical Networks, ICEN’ 10, Sidi Bel-Abbes, Algeria, september 28 –29, 2010.

CONFERENCES NATIONALES

- [12] R. D. Mohammedi, S. Arif, A. Hellal, " Application de la méthode hybride RT/RS au problème d’engagement des turbines "Unit Commitment" , Conférence sur les Systèmes Embarqués, ESC’09, Ecole Militaire Polytechnique, 05-06 mai, 2009, Alger, Algérie.

Sommaire

Introduction Générale	1
1.1 Introduction	1
1.2 Objectif et Contributions.....	2
1.3 Structure de la thèse	3
Chapitre 1: Problème d'Engagement des Turbines : Unit Commitment.....	5
1.1 Introduction	5
1.2 Notions de base	5
1.2.1 Définition d'un réseau électrique	5
1.2.2 Structure des réseaux électriques	6
1.2.3 Caractéristiques des centrales thermiques et fluctuation de la charge	6
1.2.4 Fluctuations de la consommation	8
1.3 Formulation du problème de l'écoulement de puissance optimal	9
1.3.1 Fonction objectif	10
1.3.2 Contraintes d'égalités	10
1.3.3 Contraintes d'inégalités	10
1.4 Problème de l'Unit Commitment	12
1.4.1 Formulation du Problème de l'UCP.....	12
1.5 Techniques avancées de résolution de l'UCP	15
1.5.1 Métaheuristiques	15
1.5.2 Recuit Simulé (RS)	16
1.5.3 Méthode de recherche taboue (RT).....	18
1.6 Approche proposée : la méthode hybride Recherche Taboue – Recuit Simulé RT/RS	19
1.6.1 Codage de la solution	20
1.6.2 Création de la solution initiale	20
1.6.3 Diversification (perturbation aléatoire).....	21
1.6.4 Intensification (recherche locale).....	24
1.6.5 Algorithme	25
1.7 Conclusion	27
Chapitre 2: Reconfiguration des réseaux de distribution en présence des DGs	29
2.1 Introduction	29
2.2 Notions de base	29
2.2.1 Ecoulement de puissance	29
2.2.2 La Stabilité de Tension.....	32
2.2.3 Courbe d'effondrement de tension.....	33

2.2.4	La production décentralisée	34
2.3	Emplacement et dimensionnement optimal de DGs	35
2.3.1	Fonctions objectifs	35
2.3.2	Contraintes égalités	36
2.3.3	Contraintes inégalités	36
2.3.4	Manipulation des contraintes	36
2.4	Reconfiguration du réseau de distribution en présence des DGs	37
2.4.1	Ouverture et fermeture des interrupteurs (switches)	38
2.4.2	Réseau de distribution et théorie des graphes	39
2.5	Algorithme NSGA II.....	41
2.5.1	Principe	41
2.5.2	Calcul de la distance de crowding.....	43
2.5.3	Choix de la meilleure solution de compromis.....	44
2.6	Application de l'algorithme NSGA-II sur le problème l'optimisation de l'emplacement et de la taille des DGs	44
2.7	Application de l'algorithme NSGA-II sur le problème de la reconfiguration optimale.....	45
2.8	Conclusion	47
Chapitre 3:	Modélisation Orientée Objet des Réseaux Electriques	49
3.1	Introduction	49
3.2	Notions sur la modélisation orientée objet.....	50
3.2.1	Notions de classe et d'objet.....	50
3.2.2	Principes de la programmation orientée objet.....	51
3.2.3	Avantages de la POO	52
3.3	Langage unifié de modélisation (UML).....	53
3.3.1	Modèle statique (Diagramme de classes).....	53
3.3.2	Modèle Dynamique	57
3.4	Modélisation orienté objet des réseaux électriques	58
3.4.1	Structure objet du réseau électrique	59
3.4.2	Modélisation orienté objet des applications (UCP comme exemple)	65
3.5	Module de l'interface utilisateur graphique (GUI)	69
3.6	Conclusion	71
Chapitre 4:	Description du Logiciel Développé	73
4.1	Introduction	73
4.2	Présentation du logiciel <i>VisualPower</i>	73
4.3	Représentation graphique des réseaux électriques	74
4.3.1	Données du jeu de barres	77
4.3.2	Données de la ligne de transport	78

4.3.3	Données du générateur	80
4.3.4	Données de générateur d'énergie dispersé (DG).....	82
4.4	Utilisation du logiciel.....	83
4.4.1	Exécution de l'application UCP	84
4.4.2	Exécution de l'application OTEDG	86
4.4.3	Exécution de l'application RORD.....	88
4.5	Conclusion	89
Chapitre 5: Tests et Applications aux Réseaux Electriques.....		91
5.1	Introduction	91
5.2	Application de la méthode RT/RS au problème de l'UC.....	91
5.2.1	Réseau IEEE 57-nœuds.....	92
5.2.2	Réseau Algérien 59-nœuds	94
5.3	Optimisation de la taille et de l'emplacement des DGs dans les réseaux de distribution	98
5.4	Reconfiguration du réseau de distribution en présence des DGs	106
5.5	Conclusion	115
Conclusion Générale		116
Perspectives et Suggestions.....		116
Annexe A.....		125
Annexe B.....		128
Annexe C.....		134

Liste des figures

Figure 1.1 Structure des réseaux électriques	6
Figure 1.2 Fonction coût de production	8
Figure 1.3 Exemple de la fluctuation journalière de la charge*	9
Figure 1.4 Comparaison des techniques du recuit et de la trempe	17
Figure 1.5 Minima locaux et minimum global.....	17
Figure 1.6 Exemples de perturbation	22
Figure 1.7 Exemple de réparation de la contrainte.....	23
Figure 1.8 Exemple de réparation de la contrainte.....	23
Figure 1.9 Exemple de réparation de la contrainte.....	23
Figure 1.10 Processus de recherche (RT/RS).....	25
Figure 1.11 Algorithme de la méthode hybride RT/RS	26
Figure 2.1 Schéma équivalent d'une branche	31
Figure 2.2 Algorithme de l'écoulement de puissance	32
Figure 2.3 Courbe d'effondrement de tension	34
Figure 2.4 Réseau constitué de trois boucles	38
Figure 2.5 Classement des solutions dans les frontières de Pareto	42
Figure 2.6 Principe de l'algorithme NSGA-II.....	43
Figure 3.1 Exemple de représentation graphique d'une classe	50
Figure 3.2 Association entre les classes du réseau électrique	54
Figure 3.3 Agrégation par référence et agrégation par composition	55
Figure 3.4 Représentation de l'héritage	56
Figure 3.5 Package A importe les services du package B	57
Figure 3.6 Une partie de la class TPowerComponents	60
Figure 3.7 Une partie de la class TBus	61
Figure 3.8 Une partie de la class TBranch.....	62
Figure 3.9 Une partie de la class TShuntElement.....	63
Figure 3.10 Une partie de la class TCompensateur.....	63
Figure 3.11 Une partie de la class TGenerator.....	64
Figure 3.12 Digramme des classes	66
Figure 3.13 Diagramme d'activités pour la résolution de l'UCP.....	67
Figure 3.14 Modèle objet proposé pour l'application UCP	68
Figure 3.15 Structure générale	70
Figure 4.1 Vue générale de l'éditeur graphique	75
Figure 4.2 Enregistrement du projet actuel	76
Figure 4.3 Barre d'outils Fichiers	77
Figure 4.4 Palette des composants électriques	77
Figure 4.5 Les applications proposées dans la version actuelle	77
Figure 4.6 La boîte de dialogue de Bus.....	78
Figure 4.7 La boîte de dialogue de ligne de transport.....	79
Figure 4.8 La boîte de dialogue du générateur: Coûts	80
Figure 4.9 La boîte de dialogue du générateur: puissances et contrôle de tension.....	80
Figure 4.10 La boîte de dialogue du générateur: Unit Commitment.....	81
Figure 4.11 La boîte de dialogue d'un DG de type photovoltaïque	82
Figure 4.12 La boîte de dialogue d'un DG de type éolienne	83
Figure 4.13 Fiche de l'interface de l'application UCP.....	84

Figure 4.14	Représentation graphique et tabulaire de l'UCP.....	85
Figure 4.15	Fiche de l'interface de l'application OTEDG	86
Figure 4.16	Représentation graphique et tabulaire de l'OTEDG	88
Figure 4.17	Fiche de l'interface de l'application RORD	89
Figure 5.1	Profil de la charge du réseau IEEE 57 nœuds sur un intervalle de 24 h	92
Figure 5.2	Les valeurs optimales des puissances générées (IEEE 57-nœuds)	94
Figure 5.3	Schéma unifilaire du réseau de production et transport algérien (1997)	96
Figure 5.4	Les valeurs optimales des puissances générées (Sonelgaz 59-nœuds)	98
Figure 5.5	Indice de stabilité à chaque jeu de barres avant et après l'installation du DG	101
Figure 5.6	Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 12-nœuds	102
Figure 5.7	Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 33-nœuds	102
Figure 5.8	Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 69-nœuds	103
Figure 5.9	Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 118-nœuds.....	103
Figure 5.10	Pertes de puissance active au niveau de chaque branche du réseau IEEE 12-nœuds avant et après l'installation du DG	104
Figure 5.11	Pertes de puissance active au niveau de chaque branche du réseau IEEE 33-nœuds avant et après l'installation du DG	104
Figure 5.12	Pertes de puissance active au niveau de chaque branche du réseau IEEE 69-nœuds avant et après l'installation du DG	105
Figure 5.13	Pertes de puissance active au niveau de chaque branche du réseau IEEE 12-nœuds avant et après l'installation du DG	105
Figure 5.14	Capacité maximale de charge avant et après l'installation du DG	106
Figure 5.15	Amélioration du profil de tension par rapport au cas sans reconfiguration et	110
Figure 5.16	Amélioration du profil de tension par rapport au cas sans reconfiguration et	110
Figure 5.17	Amélioration du profil de tension par rapport au cas sans reconfiguration et	111
Figure 5.18	Réseau IEEE 33-nœuds avec la configuration initiale en absence du	112
Figure 5.19	Réseau IEEE 33-nœuds avec la nouvelle configuration optimale en présence du	112
Figure 5.20	Réseau IEEE69-nœuds avec la configuration initiale en absence du	113
Figure 5.21	Réseau IEEE 69-nœuds avec la nouvelle configuration optimale en présence du	113
Figure 5.22	Réseau IEEE 118-nœuds avec la configuration initiale en absence du	114
Figure 5.23	Réseau IEEE 118-nœuds avec la nouvelle configuration optimale en présence du	114

Liste des tableaux

Tableau 3.1 Valeurs de la multiplicité	55
Tableau 3.2 Principales fonctions utilisées dans l'application UCP	68
Tableau 4.1 Données du jeu de barres	78
Tableau 4.2 Données de ligne de transport	79
Tableau 4.3 Données du générateur	81
Tableau 5.1 Meilleure planification du système IEEE 57-nœuds pour le cas 1.....	93
Tableau 5.2 Meilleure planification du système IEEE 57-nœuds pour le cas 2.....	93
Tableau 5.3 Comparaison entre LR/GA et RT/RS (IEEE 57-bus)	94
Tableau 5.4 Données techniques et économiques des générateurs du réseau test Algérien	95
Tableau 5.5 Comparaison entre LR/GA et RT/RS (IEEE 57-bus)	96
Tableau 5.6 Meilleure planification du système Sonelgaz 59-nœuds pour le cas 1.....	97
Tableau 5.7 Meilleure planification du système Sonelgaz 59-nœuds pour le cas 2.....	97
Tableau 5.8 Paramètres de simulations adoptées pour le NSGA-II.....	98
Tableau 5.9 Application de la méthode proposée sur l'OTEDG	100
Tableau 5.10 Application de la méthode analytique, PSO et CS sur l'OTEDG.....	100
Tableau 5.11 Résultats avant et après reconfiguration optimale en absence <i>des DGs</i>	108
Tableau 5.12 Résultats avant et après reconfiguration optimale en présence des DGs	109

Liste des abréviations

GUI	Graphical User Interface (Interface graphique pour l'utilisateur)
TS	Tabu Search (Algorithme Recherche Tabou)
DG	Génération d'énergie dispersée (Distributed generation)
UCP	Unit Commitment Problem (Problème d'Engagement de Turbines)
OPF	Optimal Power Flow (Ecoulement de Puissance Optimal)
EDP	Economic Dispatch Problem (Dispatching Economique)
UML	Unified Modeling Language (Langage Unifié de Modélisation)
MOO	Modélisation Orientée Objets
POO	Programmation Orientée Objets
SA	Simulated Annealing (Recuit Simulé)
RT/RS	Recherche Tabou- Recuit Simulé
NSGA	Non Sorting Genetic Algorithm
PSO	Algorithme à essaim de particules (Particle Swarm Optimiser)
CS	Cuckoo search (Recherche Coucou)
LR/GA	Relaxation Lagrangienne / Algorithme Génétique

Introduction Générale

1.1 Introduction

Les systèmes d'énergie électrique constituent une des composantes essentielles du développement économique. Leur rôle est de fournir aux utilisateurs le produit électricité au moindre coût dans des conditions de qualité et de sécurité satisfaisantes. Pour atteindre cet objectif, il est nécessaire que tout au long de la chaîne conception-réalisation- exploitation, de disposer des moyens permettant de comprendre, d'analyser et de prédire le comportement des systèmes d'énergie électrique. Cette analyse passe naturellement aujourd'hui par la simulation des systèmes électriques. La simulation du régime stationnaire et dynamique de n'importe quel domaine de technologie exige les outils du logiciel informatique afin de comprendre le comportement complet du système. D'autre part dans la technologie du domaine des systèmes électriques, la nécessité des outils de software informatique est inévitable en raison de la complexité impliquée dans les calculs et aussi en raison du coût élevé des équipements électriques.

Pour planifier et analyser le comportement du système électrique actuel et futur sous différentes conditions de fonctionnement et sous diverses contraintes, un outil informatique puissant est nécessaire. Certains simulateurs disponibles sur le marché (tels que, EuroStag, SPARD, ETAP, PowerWorld, EasyPower, CYME, DigSILENT, NEPLAN, UWPFLOW.. etc.) sont limités à des applications bien spécifiques [1]. Bien qu'ils soient efficaces et capables de simuler des systèmes électriques de grandes tailles, leurs modèles sont souvent encapsulés et difficiles, ou voire impossible, à examiner et/ou modifier. Aussi, ces simulateurs exigent souvent une formation consistante et par conséquent, inadaptés pour l'utilisation académique et pédagogique.

Pour une utilisation académique et pédagogique, il est plus important que la modélisation des composants soit transparente et flexible [1, 2] . Ces qualités peuvent être offertes par l'utilisation de la programmation orientée objet (POO) [2].

La POO s'appuie sur la métaphore des objets communiquant entre eux. Selon la POO, un système peut être vu comme un ensemble d'objets qui collaborent pour assurer une mission globale. Actuellement la modélisation orientée objet (MOO) des réseaux électriques fait sortir deux grandes structures de classes : la structure des éléments physiques du système électrique et les méthodes d'analyse (applications) appliquées au système ou sur ces éléments [1-7]. Dans ce travail, le processus de création des structures de classes représentatives des

entités diverses du réseau électrique dans son ensemble sont divisées en abstractions distinctes. On considère deux abstractions principales : le réseau électrique (classes des éléments physiques) et les fonctions de calcul (classes des applications).

La MOO offre plusieurs avantages dont on cite, la flexibilité, la fiabilité, la maintenabilité, la réutilisabilité, la modularitéetc. La modélisation de systèmes complexes, tels que le système électrique, est devenue encore simple et facile par l'utilisation du langage standard unifié de modélisation (en anglais, Unified Modeling Language UML) [8]. Cette approche, initialement introduite dans les langages de programmation célèbres comme C++, JAVA, DELPHI, est maintenant largement employée par la communauté des développeurs de logiciels de grande envergure [9].

D'autre part, les phénomènes physiques ainsi que les données d'un réseau électrique sont mieux assimilées si l'information est représentée sous forme graphique contrairement à la forme numérique. Pour des systèmes complexes, où les interactions homme-machine sont nombreuses, la spécification de la GUI (Graphical User Interface) par une approche orientée objets offre des avantages indéniables en terme de génie logiciel. La partie la plus importante de la GUI concerne la représentation des diagrammes unifilaires ou l'éditeur graphique.

1.2 Objectif et Contributions

L'objectif de cette thèse est le développement d'un logiciel robuste pour visualisation, l'analyse et l'optimisation des réseaux électriques qui est en particulier flexible, extensible et facile à maintenir en utilisant les avantages de la méthodologie orientée-objet où le réseau électrique est modélisé comme un ensemble d'objets liés et structurés dans une hiérarchie de classes. Chaque classe décrit un objet encapsulant sa structure de données et son comportement. Les objets réagissent réciproquement et coopèrent entre eux-mêmes en utilisant l'envoi de message "message passing". Toutes les parties du logiciel (méthodes numériques, base de données, interface utilisateur graphique...) ont été construites sous forme d'objets en employant des concepts de l'héritage pris du langage informatique Delphi afin d'admettre une structuration convenable des modèles. Ce logiciel peut être utilisé pour des fins pédagogiques ou pour la modélisation et l'analyse des systèmes de puissance électrique.

Trois applications intégrées dans la version actuelle de notre logiciel sont présentées dans cette thèse, toutes les trois étant liées à la conduite des réseaux électriques que ce soit au niveau de la production ou de la distribution : engagement de turbines (Unit Commitment),

optimisation de l'emplacement et de la taille de générateurs dispersés (DGs) dans les réseaux de distribution, et reconfiguration optimale des réseaux en présence de DGs.

1.3 Structure de la thèse

- Le Chapitre 1 présente la formulation détaillée de la première application (Unit Commitment) sans et avec les contraintes d'écoulement de puissance optimal et propose une nouvelle technique se basant sur l'hybridation de deux métaheuristiques (la recherche taboue et le recuit simulé) pour la résolution du problème.
- Le Chapitre 2 présente deux applications reliées aux réseaux électriques de distribution, à savoir l'optimisation de l'emplacement et de la taille des générateurs dispersés (DGs) dans les réseaux de distribution, et la reconfiguration optimale des réseaux en présence des DGs.
- Le Chapitre 3 introduit et présente les différents concepts fondamentaux de la technique orientée objets. Il s'agit de donner les définitions strictes et précises de ces concepts. Il présente également la notation UML et une brève description des patrons de conception
- Le Chapitre 4 présente une description générale du logiciel développé concernant, la visualisation, la simulation et l'analyse des systèmes électriques.
- Le Chapitre 5 expose des simulations sur les réseaux test IEEE de différentes tailles ainsi que le réseau Algérien. Les résultats obtenus sont aussi analysés et comparés avec ceux trouvés dans la littérature.
- Enfin, la Conclusion qui résume brièvement ce travail et les contributions majeures, et une discussion des travaux futurs conclut la présentation.

Chapitre 1

Problème d'Engagement des Turbines : Unit Commitment

Chapitre 1: Problème d'Engagement des Turbines : Unit Commitment

1.1 Introduction

L'exploitation des réseaux électriques pose de nombreux problèmes d'ordre technique et économique. Les programmes de gestions élaborées par les experts doivent garantir en tout temps et en tout lieu la couverture de l'énergie demandée, d'assurer une qualité acceptable de la puissance livrée et de procurer une sécurité d'alimentation élevée avec un coût aussi faible que possible. En l'absence de possibilité de stockage de l'énergie à grande échelle, il est indispensable de maintenir à tout instant l'équilibre entre la production et la consommation avec minimisation du coût des unités de production.

Le problème d'établir un programme "allumage/extinction" optimal pour des centrales électriques dans un système de production d'énergie s'appelle le problème d'engagement des turbines (*Unit Commitment Problem UCP*). Un sous problème de l'UCP doit trouver la production optimale pour une combinaison donnée des unités en marche. Ce problème s'appelle le problème de répartition économique (*Economic Dispatch Problem EDP*).

1.2 Notions de base

1.2.1 Définition d'un réseau électrique

On appelle réseau électrique l'ensemble des infrastructures permettant d'acheminer l'énergie électrique des centres de production (centrales électriques), vers les consommateurs d'électricité, la structure des réseaux électriques est présentée dans la Figure 1.1.

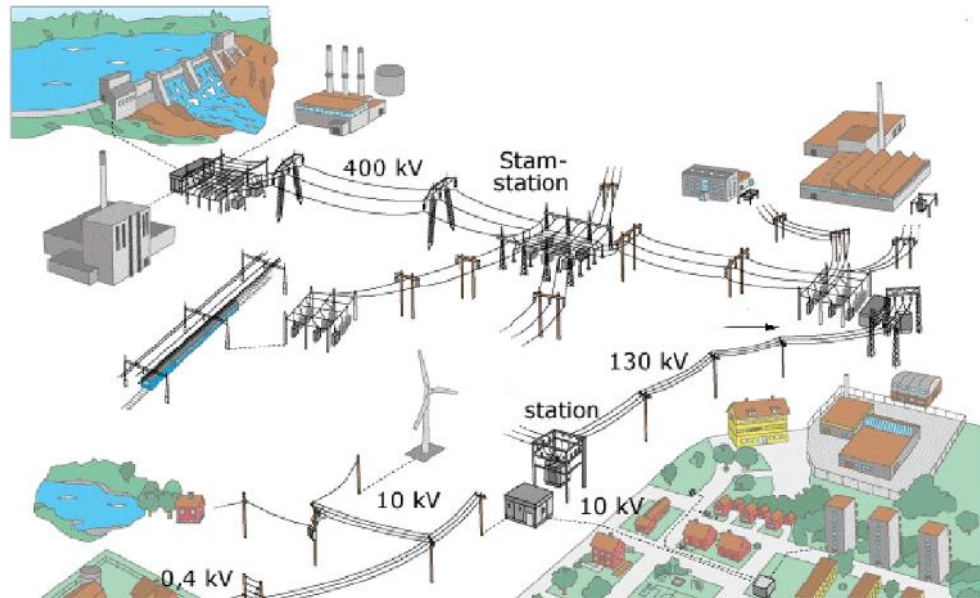


Figure 1.1 Structure des réseaux électriques

1.2.2 Structure des réseaux électriques

- **Réseau de transport** : ce réseau a pour mission principale d'assurer l'acheminement de l'énergie des centres de production vers les centres de consommations.
- **Réseau d'interconnexion** : est un réseau de transport qui a la particularité d'assurer l'échange énergétique entre les différentes zones géographiques.
- **Réseau de répartition** : ce réseau est un réseau intermédiaire qui a pour mission d'assurer la livraison de l'énergie à des grands consommateurs.
- **Réseau de distribution** : Ils ont pour rôle de fournir aux réseaux d'utilisation la puissance dont ils ont besoin. L'énergie électrique fournie aux consommateurs (clients) par l'intermédiaire du réseau de distribution de la Sonelgaz selon quatre niveaux de tension normalisés: 30 kV -10 kV et 0.4 KV.

1.2.3 Caractéristiques des centrales thermiques et fluctuation de la charge

a) Moyens de production

Nous définissons un moyen de production d'énergie électrique comme toute installation capable de convertir une source d'énergie primaire en énergie électrique pouvant être injectée sur un réseau. La centrale de production est la composante élémentaire de l'organisation des moyens de production d'électricité.

Une centrale peut regrouper sur un même site plusieurs unités de production, ou tranches, souvent de même technologie et de même puissance. Chaque unité de production est composée d'une turbine (ou en général un moteur) et d'un alternateur pour produire de

l'énergie électrique à courant alternatif. Les moteurs ou turbines peuvent être entraînés par différentes sources d'énergie primaires.

b) Centrales thermiques

Les centrales thermiques à vapeur brûlent du charbon, du pétrole ou du gaz pour vaporiser de l'eau. La vapeur ainsi produite se détend dans une turbine à vapeur, qui entraîne un alternateur produisant l'électricité. Ces centrales sont caractérisées par un rendement énergétique moyen. Leurs coûts variables dépendent notamment du prix du combustible utilisé. Elles sont réputées pour être souples à l'exploitation, une fois en fonctionnement, mais lentes à mettre en fonctionnement (une centrale à fuel ou au charbon prend environ 16h pour démarrer à froid, et 8h pour démarrer à chaud).

Les turbines à combustion, ou les moteurs Diesel à combustion interne, utilisent la détente des gaz produits par la combustion de pétrole ou de gaz. Ces centrales sont caractérisées par un rendement énergétique bas. En revanche, elles sont réputées pour être très flexibles et rapides au démarrage (entre 15 minutes et 1 h).

Les cycles combinés sont le fruit d'une technologie hybride. Une turbine à combustion (ou plusieurs) fonctionne en parallèle d'une turbine à vapeur. Le gaz d'échappement de la turbine à combustion est utilisé pour produire de la vapeur dans une chaudière classique. Cette chaudière alimente la turbine à vapeur. Ces centrales sont caractérisées par un très bon rendement, et donc un coût variable relativement faible (mais dépendant du prix du combustible). En terme de souplesse, ces centrales sont un peu plus souples qu'une centrale thermique classique en cycle simple. Le temps de démarrage est, également, plus court qu'une centrale classique [10].

c) Caractéristiques des centrales thermiques

La Figure 1.2 montre un exemple de coût d'exploitation d'une centrale thermique. Ce coût est normalement représenté par une équation quadratique [11] du type :

$$\text{Coût} = F(P) = a + bP + cP^2 \quad (1.1)$$

Avec P est la quantité de puissance produite. La constante a est appelée coût de marche à vide (*No-load cost* en anglais) et représente le coût pour tenir démarrée une unité de production à vide. Le coût incrémental (ou marginal) de production est le coût pour produire une unité supplémentaire d'énergie. Ce coût est important pour prendre les décisions d'exploitation à court terme :

$$\text{Coût marginal} = dF(P)/dP = b + 2cP \quad (1.2)$$

Outre le coût variable à court terme, d'autres caractéristiques spécifiques sont importantes à mentionner pour la production d'électricité. C'est le cas notamment du coût spécifique pour démarrer ou arrêter l'unité de production (coût de démarrage et d'arrêt, *start-up cost* et *shutdown cost* en anglais). Le coût de démarrage correspond au coût de l'énergie nécessaire pour mettre en fonctionnement toutes les installations permettant la production d'électricité (chaudières, pompes, etc.). Ce coût dépend normalement de l'état de l'unité de production au moment de démarrage (démarrage à froid ou à chaud). Certaines contraintes techniques sont aussi importantes pour l'exploitation. Généralement, l'unité de production ne peut fonctionner de manière stable qu'à partir d'un niveau de production minimal (capacité minimale de production) et jusqu'à un niveau maximal de production (capacité maximale de production). L'inertie propre des moyens de production limite la vitesse à laquelle les unités de production peuvent changer leur niveau de production. La vitesse maximale de changement du niveau de production pour une période de temps donné est appelée contrainte de rampe (*ramp rate*). Il existe aussi un temps minimal pour le démarrage (*start-up time*) et un temps minimal d'extinction (*shut-down time*).

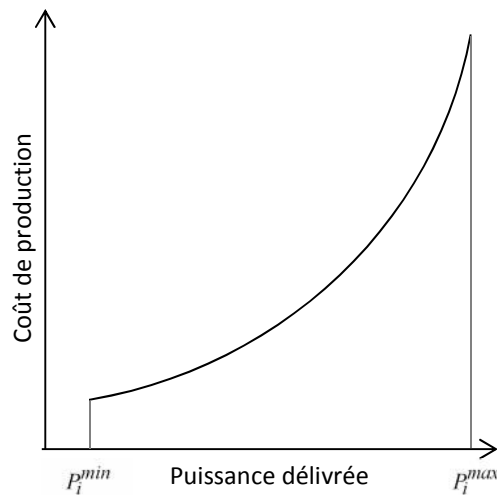


Figure 1.2 Fonction coût de production

1.2.4 Fluctuations de la consommation

La consommation d'électricité varie en permanence en fonction des multiples variations de consommations qui interviennent « sans préavis » sur l'ensemble du système.

Les différentes utilisations individuelles de l'énergie électrique, à chaque moment, se traduisent par de fortes fluctuations de la consommation dans le temps. Cependant, ces fluctuations ont un certain caractère cyclique au cours de la journée, de la semaine, et de l'année.

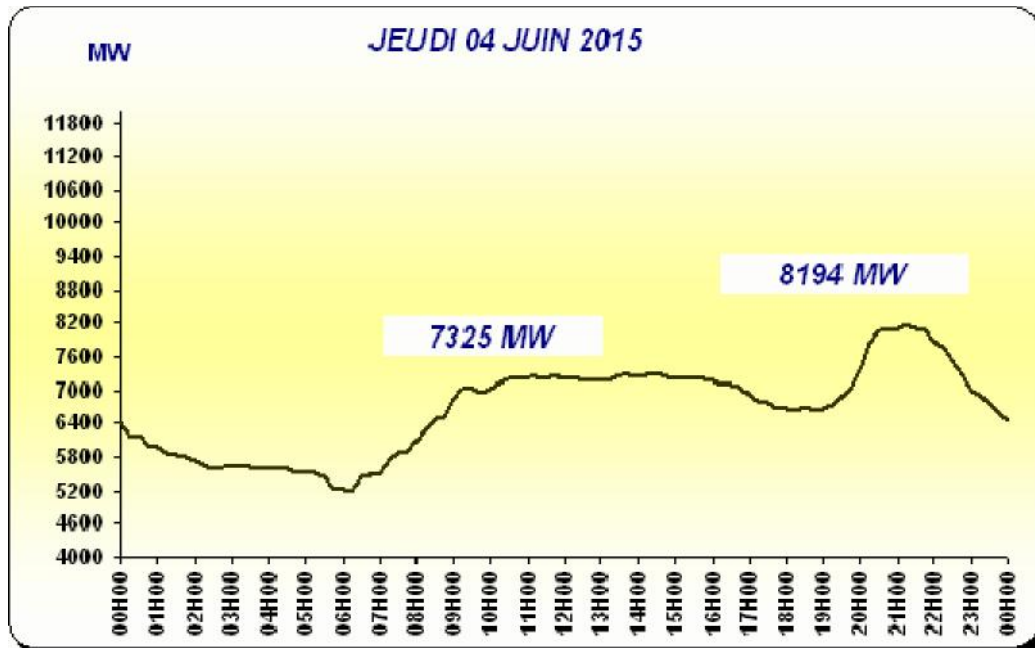


Figure 1.3 Exemple de la fluctuation journalière de la charge*

La Figure 1.3 met en exergue les fluctuations de la consommation en représentant la consommation d'électricité en Algérie correspondant au 04/06/2015*. La consommation d'électricité varie constamment au cours d'une même journée, d'une même semaine, et au fil de l'année. Elle reflète les horaires de travail, les jours de congés, les saisons et est très sensible à la température.

1.3 Formulation du problème de l'écoulement de puissance optimal

Le problème de la répartition optimale des puissances est un problème d'optimisation dont l'objectif est de minimiser le coût total de la production de la puissance d'un réseau électrique. Si on prend en considération seulement la fonction objectif, on parle alors d'une optimisation sans contraintes. Mais si on prend en considération les équations de l'écoulement de puissance, on est donc devant un problème d'optimisation avec contraintes d'égalités. Si on prend de plus les limites min. et max. des puissances générées par les alternateurs, la surcharge des lignes de transports et les niveaux de tensions admissibles pour les jeux de barres de charges, on est alors devant un problème d'optimisation avec contraintes d'égalités et d'inégalités.

* Données correspondant au 04/06/2015. Source : <http://www.ose.dz/courbes.php>

Le problème de l'écoulement de puissance optimal est donné sous une forme standard d'optimisation avec contraintes d'égalités et d'inégalités comme suit [12, 13]:

$$\begin{aligned}
 & \min. F(x) \text{ (fonction objectif)} \\
 & \text{selon} \\
 & g_i(x) = 0, i = 1, 2, \dots, n \text{ (contraintes d'égalités)} \\
 & \text{et} \\
 & h_j(x) \leq 0, j = 1, 2, \dots, n \text{ (contraintes d'inégalités)}
 \end{aligned} \tag{1.3}$$

1.3.1 Fonction objectif

Cette fonction reflète le besoin de minimiser le coût total de la production des puissances actives. On suppose que le coût individuel de chaque centre de production dépende uniquement de la génération de la puissance active :

$$F = \sum_{i=1}^{n_g} f_i = \sum_{i=1}^{n_g} a_i + b_i P_{Gi} + c_i P_{Gi}^2 \tag{1.4}$$

1.3.2 Contraintes d'égalités

Ces contraintes sont l'image des lois physiques gouvernant le système électrique. Elles sont représentées par les équations non linéaires de l'écoulement de puissance. Il faut que la somme des puissances active et réactive injectées dans chaque jeu de barres soit égale à zéro.

$$g_i(x_1, \dots, x_n) = 0 \quad i = 1, \dots, m \tag{1.5}$$

$$\Delta P_i = 0 = V_i \sum_{j=1}^n V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}) - P_{Gi} + P_{Di} \tag{1.6}$$

$$\Delta Q_i = 0 = V_i \sum_{j=1}^n V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}) - Q_{Gi} + Q_{Di} \tag{1.7}$$

1.3.3 Contraintes d'inégalités

En pratique, on ne doit pas dépasser les limites des éléments physiques du réseau électrique tels que les générateurs, les transformateurs à prises de charge, et les transformateurs de phase.

En plus des contraintes sur les puissances actives à chaque générateur qui a une influence directe sur la fonction coût, on peut citer d'autres contraintes d'inégalités [12, 13]:

- ✓ La puissance réactive générée Q_{Gi} qui est limitée par une borne inférieure $Q_{Gi \min}$ et une borne supérieure $Q_{Gi \max}$

$$Q_{Gi \min} \leq Q_{Gi} \leq Q_{Gi \max} \quad i = 1, \dots, n_g \quad (1.8)$$

- ✓ Les transformateurs à prises de charge ont des déviations max. et min. du niveau de tension par rapport à la tension nominale. De même les transformateurs à angles de phase ont des décalages max. et min. des phases des tensions. Les deux types de transformateurs forment les contraintes d'inégalités suivantes:

$$\begin{cases} t_{ij \min} \leq t_{ij} \leq t_{ij \max} \\ r_{ij \min} \leq r_{ij} \leq r_{ij \max} \end{cases} \quad (1.9)$$

- ✓ Pour maintenir la sécurité du système électrique, les lignes de transport et les transformateurs de puissances ont des limites sur le transit de puissance apparente. Ces limites sont dues aux pertes thermiques dans les conducteurs, et/ou la stabilité du système. Elles sont représentées par une contrainte d'inégalité, qui limitera le carré de puissance en MVA d'un transformateur ou d'une ligne de transport.

$$|S_{ij}|^2 - |S_{ij \max}|^2 \leq 0 \quad (1.10)$$

- ✓ Pour garder la qualité de service électrique et la sécurité du système, les niveaux de tension des jeux de barres doivent toujours être entre leurs limites max. et min. Ces limites exigent encore l'addition des contraintes d'inégalités.

$$V_{i \min} \leq V_i \leq V_{i \max} \quad (1.11)$$

Donc il y'a n contraintes d'égalités et m contraintes d'inégalités et le nombre des variables du problème est égal à la taille du vecteur des variables de contrôle (y compris puissances active et réactive générées, niveaux de tension des jeux de barres, prises des transformateurs,... etc.).

La solution du problème d'OPF exige la formulation de la fonction Lagrangien appelée aussi la fonction de coût augmentée suivante:

$$L = F + \sum_{i=1}^n \lambda_i g_i + \sum_{j=1}^m \tilde{\lambda}_j h_j \quad (1.12)$$

Les conditions nécessaires pour trouver un minimum de L appelées conditions de Kuhn-Tucker sont les suivantes:

$$\begin{cases} \frac{\partial L}{\partial x_i} = 0 & \& \frac{\partial L}{\partial \lambda_i} = g_i(x) = 0 & i = 1, \dots, n \\ \frac{\partial L}{\partial \tilde{\lambda}_j} = h_j(x) \leq 0 & & & j = 1, \dots, m \\ \tilde{\lambda}_j h_j(x) = 0 & \& \tilde{\lambda}_j > 0 & \end{cases} \quad (1.13)$$

1.4 Problème de l'Unit Commitment

Puisque l'activité humaine suit des cycles, la plus part des services fournis à la population sont rythmés. En ce qui concerne l'électricité, la demande totale d'énergie sera généralement plus élevée, durant la journée et en début de soirée quand les charges industrielles sont fortes et l'éclairage allumé, et plus faible pendant la nuit et ce jusqu'à la matinée. En outre, la demande d'électricité suit un cycle hebdomadaire puisque la charge est plus faible le week-end que la semaine. Mais pourquoi est-ce un problème dans la conduite des réseaux électriques? Pourquoi ne pas simplement démarrer et maintenir en fonctionnement autant de centrales qu'il le faut pour couvrir la demande maximale? La réponse est purement économique : maintenir trop de centrales en fonctionnement est bien trop coûteux pour être viable et une somme d'argent considérable serait épargnée si on démarrait les unités comme il le faut. Ajoutons qu'il est en réalité impossible de maintenir une unité en permanence en marche puisque les centrales sont mises à l'arrêt lorsqu'elles doivent être révisées. Le problème d'Unit Commitment consiste à choisir quelles unités de production seront opérationnelles, à l'arrêt, ou en réserve chaude, de manière à maximiser le profit du parc de production. Les unités doivent satisfaire la charge ainsi que la réserve tournante. De plus, chaque unité possède ses propres limites de production. Il s'agit donc d'un problème d'optimisation à grande échelle sous contraintes. Les approches les plus utilisées pour sa résolution sont des arrangements de la liste de priorités, programmation dynamique et relaxation de Lagrange [11, 14]. Néanmoins les références [15-24] présentent plusieurs approches pour la résolution du problème d'UC basé sur des formulations des métaheuristiques.

1.4.1 Formulation du Problème de l'UCP

a) Fonction objective de l'UCP

L'objectif du problème de l'Unit Commitment est la minimisation de la fonction coût de production total [25]:

$$\min_{P_i(t), U_i(t)} \left[F_T = \sum_{t=1}^{N_t} \sum_{i=1}^N [F_i(P_i(t))U_i(t) + ST_i(t)U_i(t) + DC_i(t)(1-U_i(t))U_i(t-1)] \right] \quad (1.14)$$

Avec

$U_i(t)$ est l'état de l'unité i à l'instant t : un zéro signifie que la centrale est à l'arrêt, un 1 qu'elle est en fonctionnement.

$F_i(P_i(t))$ est le coût de production de l'unité i à l'instant t , dans le cas le plus fréquent :

$$F_i(P_i(t)) = a_i + b_i P_i(t) + c_i P_i^2(t) \quad (1.15)$$

$$i=1, \dots, N$$

$$t=1, \dots, N_t$$

$ST_i(t)$ représente le coût de redémarrage de l'unité i à l'instant t , il dépend du temps pendant lequel l'unité i a été éteinte X_i^{OFF} , il peut être représenté par l'une des deux formes:

Fonction à deux pas [26]:

$$ST_i(t) = \begin{cases} HSC_i, & \text{Si } MDT_i \leq X_i^{OFF} \leq MDT_i + SC_i \\ CSC_i, & \text{Si } X_i^{OFF} > MDT_i + SC_i \end{cases} \quad (1.16)$$

CSC_i (Cold Start Cost) coût de redémarrage à froid de l'unité i (\$).

HSC_i (Hot Start Cost) coût de redémarrage à chaud de l'unité i (\$).

SC_i (Cold Start) durée de redémarrage à froid de l'unité i (heures).

Fonction exponentielle [27]:

$$ST_i(t) = b_0 \left(1 - \exp\left(-X_i^{OFF} / \dagger\right) \right) + b_1 \quad (1.17)$$

$DC_i(t)$ représente le coût d'extinction de l'unité i à l'instant t , il est souvent constant.

Remarque

Pour tenir compte des contraintes des zones interdites et des contraintes environnementales, la fonction de coût de production $F_i(P_i(t))$ sera modifiée suivant le cas, ses contraintes sont considérées parmi les contraintes du sous- problème de répartition économique de puissance.

b) Contraintes de l'UCP

Le problème du l'Unit Commitment est lié aux contraintes suivantes :

b.1) Contraintes du Système

(1) Demande à satisfaire

La puissance délivrée par les unités allumées doit être égale à la somme de la charge totale et les pertes :

$$\sum_{i=1}^N P_i(t) U_i(t) = P_D(t) + P_L(t) \quad t=1, \dots, N_t \quad (1.18)$$

(2) Réserve à garantir

Une notion très importante est celle de la sécurité. En effet, on n'est jamais à l'abri d'une défaillance technique, et on ne peut tolérer le fait de couper l'électricité dans tout un secteur dès qu'une unité est forcée de déclencher. Il faut donc une certaine quantité de réserves tournantes à chaque instant t notée $P_R(t)$. On considère généralement que la réserve doit assurer 10% de la demande :

$$\sum_{i=1}^N P_i^{max}(t) U_i(t) = P_D(t) + P_R(t) \quad t=1, \dots, N_t \quad (1.19)$$

Les pertes par le système de transmission peuvent être évaluées approximativement par :

$$P_L(t) = \sum_{i=1}^N \sum_{j=1}^N P_i(t) B_{ij} P_j(t) + \sum_{i=1}^N B_{i0} P_i(t) + B_{00} \quad (1.20)$$

Avec :

P_j : Puissance générée par l'unité j (MW).

B_{ij} : Élément (i, j) d'une matrice carrée de dimension $(N \times N)$.

B_{i0} : Élément i d'un vecteur de dimension N .

B_{00} : Pertes constantes (MW).

b.2) Contraintes des Unités

(3) Puissance Bornée

Pour chaque unité allumée, la puissance délivrée doit être incluse entre la limite minimale et la limite maximale de génération :

$$P_i^{min} \leq P_i(t) \leq P_i^{max} \quad \text{Si} \quad U_i(t) = 1 \quad (1.21)$$

$$i=1, \dots, N$$

$$t=1, \dots, N_t$$

(4) Temps minimal d'allumage

Quand une unité est démarrée, on ne peut pas l'atteindre que si le temps de marche dépasse le temps minimal de fonctionnement de cette unité :

$$MUT_i \leq X_i^{ON} \quad i=1, \dots, N \quad (1.22)$$

(5) Temps minimal d'extinction

Quand une unité est éteinte, on ne peut pas la démarrer que si le temps d'arrêt dépasse le temps minimal d'extinction de cette unité :

$$MDT_i \leq X_i^{OFF} \quad i=1, \dots, N \quad (1.23)$$

L'inertie propre des moyens de production limite la vitesse à laquelle les unités de production peuvent changer leur niveau de production. La vitesse maximale de changement du niveau de production pour une période de temps donné est appelée contrainte de rampe (*ramp rate*). Pour chaque unité i , on exige une élévation maximale de production UR_i et une chute maximale de production DR_i ce qui ajoute les contraintes (6) et (7) :

(6) Élévation maximale de la puissance

$$P_i(t) \leq \min(P_i^{max}, P_i(t-1) + UR_i) \quad si \quad (1.24)$$

$$U_i(t-1) = 0 \quad et \quad U_i(t) = 1$$

$$i=1, \dots, N$$

$$t=1, \dots, N_t$$

(7) Chute maximale de la puissance

$$P_i(t) \geq \max(P_i^{min}, P_i(t-1) - DR_i) \quad si \quad (1.25)$$

$$U_i(t-1) = 1 \quad et \quad U_i(t) = 0$$

$$i=1, \dots, N$$

$$t=1, \dots, N_t$$

(8) Autres contraintes [26]

Il existe d'autres contraintes tels que : l'état initial des unités, les puissances initiales délivrées par les unités, la contrainte d'équipement, et le fonctionnement obligatoire de certaines unités (Must-Run).

1.5 Techniques avancées de résolution de l'UCP

1.5.1 Métaheuristiques [28]

Les métaheuristiques* forment une famille d'algorithmes d'optimisation (également appelés algorithmes d'approximation) visant à résoudre des problèmes d'optimisation difficile issus de la recherche opérationnelle pour lesquels on ne connaît pas de méthode classique plus efficace. Elles permettent de résoudre des problèmes d'optimisation auxquels les ingénieurs et les décideurs sont régulièrement confrontés. Les métaheuristiques sont généralement des algorithmes stochastiques itératifs, qui progressent vers un optimum par échantillonnage d'une fonction objectif.

* Le mot métaheuristique est dérivé de la composition de deux mots grecs:
 - heuristique qui vient du verbe heuriskein (euriskein) et qui signifie 'trouver'
 - meta qui est un suffixe signifiant 'au-delà', 'dans un niveau supérieur'.

L'exécution des métaheuristiques se déroule en trois phases :

- Diversification
- Intensification
- Mémoire

La diversification permet de bien couvrir l'espace des solutions, et de déterminer les zones « prometteuses » ; l'intensification permet d'approfondir la recherche, à l'intérieur de chacune des zones prometteuses localisées.

La mémoire est le support de l'apprentissage permettant à l'algorithme de ne tenir compte que des zones où l'optimum est susceptible de se trouver et de garder en mémoire les résultats passés pour guider l'optimisation aux itérations suivantes.

Les métaheuristiques progressent itérativement et alternativement entre les phases de diversification, d'intensification et d'apprentissage. La phase originale est souvent choisie aléatoirement puis l'algorithme continue jusqu'à ce qu'un critère d'arrêt (fonctions objectifs = optimums) soit atteint.

Les métaheuristiques sont souvent inspirées par des systèmes naturels, qu'ils soient pris en physique (cas du recuit simulé), en biologie de l'évolution (cas des algorithmes génétiques) ou encore en éthologie (cas des algorithmes de colonies de fourmis ou de l'optimisation par essaims particulaires).

1.5.2 Recuit Simulé (RS) [28, 29]

Le Recuit Simulé (*Simulated Annealing*) a été proposé par les physiciens Kirkpatrick, Gelatt et Vacchi en 1983 [29]. Le recuit s'inspire de méthodes de simulation de Metropolis (années 50) en mécanique statistique. L'analogie historique s'inspire du *recuit des métaux* (*annealing*) en métallurgie : un métal refroidi trop vite présente de nombreux défauts microscopiques, c'est l'équivalent d'un minimum local pour un problème d'optimisation combinatoire. Si on le refroidit lentement, les atomes se réarrangent, les défauts disparaissent, et le métal a alors une structure très ordonnée, équivalente du minimum global pour un problème d'optimisation combinatoire (Figure 1.4).

Une autre analogie est de voir l'ensemble des solutions comme un paysage de montagne comme la Figure 1.5. Une pierre lâchée dans un tel paysage va glisser dans le premier creux venu : c'est l'analogie d'une recherche locale simple. Une balle élastique peut, au contraire, rebondir et contourner les obstacles. L'énergie initiale de la balle va diminuer lentement à chaque rebond, et la balle finira plus bas que la pierre, du moins en probabilité.

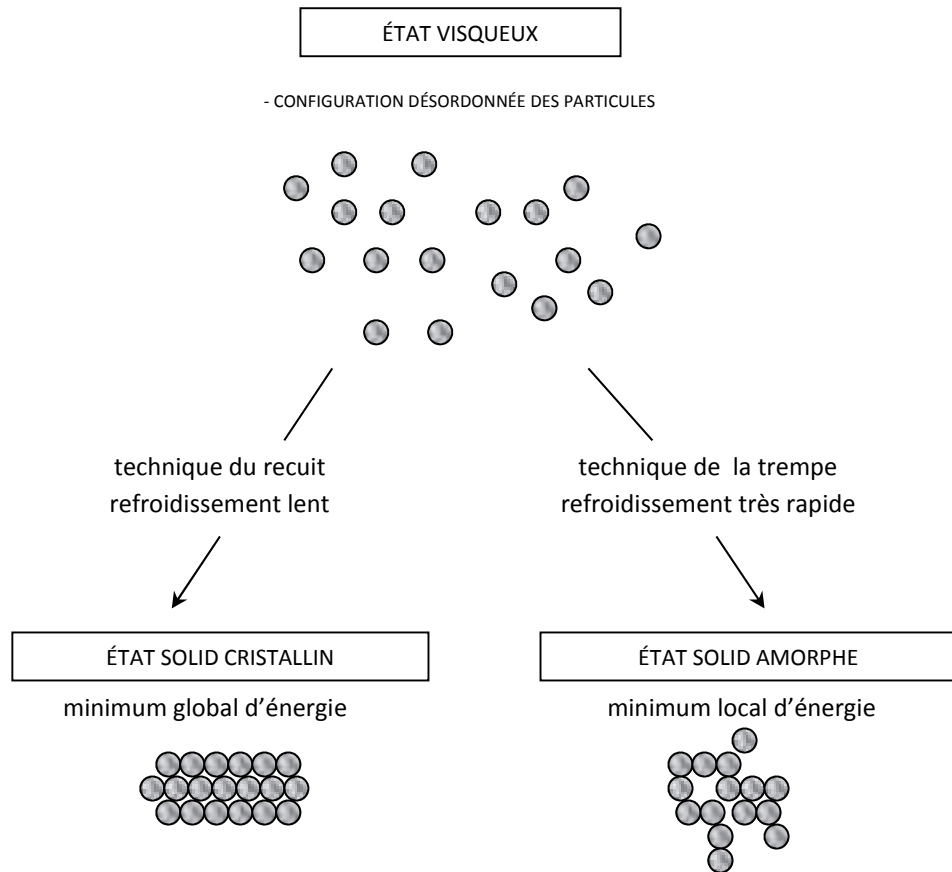


Figure 1.4 Comparaison des techniques du recuit et de la trempe

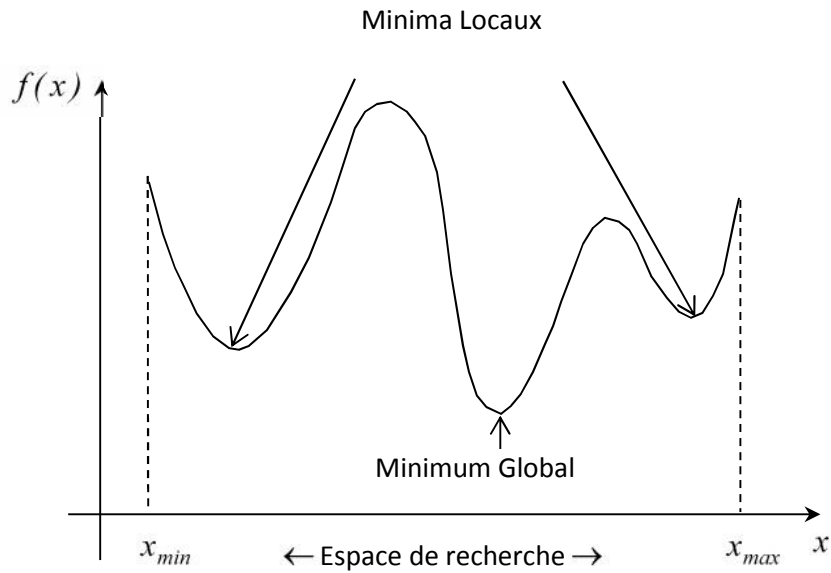


Figure 1.5 Minima locaux et minimum global

Le recuit simulé en optimisation combinatoire n'a plus qu'un lointain rapport avec la thermodynamique. L'énergie du système est représentée par un réel arbitraire T , la température. A partir d'une recherche locale quelconque pour un problème, on obtient une méthode de recuit comme suit (on part toujours d'une solution réalisable initiale s) [29]:

- On tire *au sort* une transformation $s \rightarrow s', s' \in V(s)$, au lieu de chercher la meilleure ou la première solution voisine améliorante comme une recherche locale classique.
- On calcule la *variation de coût* $\Delta f = f(s') - f(s)$.
- Si $\Delta f \leq 0$, le coût diminue et on effectue la transformation améliorante comme dans une recherche locale ($s := s'$).
- Si $\Delta f > 0$, le coût remonte, c'est un rebond, on calcule une *probabilité d'acceptation* $pa = \exp(-\Delta f / T)$, puis on tire au sort p dans $[0, 1]$. Si $p \leq pa$, la transformation est déclarée *acceptée*, bien qu'elle dégrade le coût, et on fait ($s := s'$). Sinon, la transformation est *rejetée* : on garde s pour l'itération suivante.
- Pour assurer la *convergence* (analogie avec la balle qui rebondit de moins en moins), T est diminuée lentement à chaque itération, par exemple $T = k.T$, $k = 0.999$ par exemple. On peut aussi décroître T par paliers. Pour être efficace, un recuit doit diminuer T assez lentement, en plusieurs milliers ou dizaines de milliers d'itérations. Il dure en tout cas beaucoup plus longtemps qu'une recherche locale, puisque les transformations améliorantes de cette dernière sont diluées parmi de nombreux rebonds.
- On s'arrête quand T atteint un seuil fixe v , proche de 0.

Le réglage des paramètres est assez délicat. Il est prudent de prévoir deux tests d'arrêt supplémentaires : un limitant le nombre d'itérations à une valeur $MaxItr$, et un limitant le nombre d'itérations sans changement de coût à une valeur $MaxGel$, $MaxGel$ doit être assez grand. Contrairement à une recherche locale, un recuit simulé peut en effet s'en sortir d'un plateau horizontal par des transformations à coût nul.

En cas de refroidissement trop rapide, la solution finale n'est pas la meilleure trouvée et il vaut donc mieux stocker en cours de route toute solution améliorante. Voici des exemples de valeurs typiques des paramètres : $MaxItr = 10000, v = 10^{-2}, MaxGel = 200, k = 0.9995$.

1.5.3 Méthode de recherche taboue (RT)

a) Principe [28, 30]

Les méthodes Taboues (*tabu search*) ont été proposées par Glover en 1986 [30]. Elles sont de conception plus récente que le recuit. Elles ont trois caractéristiques fondamentales :

- A chaque itération, on examine complètement le voisinage $V(s)$ de la solution actuelle s , et on va sur la meilleure solution s' , *même si le coût remonte*.

- On s'interdit de revenir sur une solution visitée dans un passé proche grâce à une liste taboue TL (*tabu list*) de longueur limitée NT , qui stocke de manière compacte les NT dernières solutions visitées. On cherche donc s' dans $V(s) - T$.
- On conserve en cours de route la meilleure solution trouvée s^* car, contrairement au recuit, c'est rarement la dernière. On stoppe après un nombre maximal $NMax$ d'itérations, ou après un nombre maximal d'itérations sans améliorer la meilleure solution, ou quand $V(s) - T = \emptyset$. Ce dernier cas, très rare, ne peut se produire que sur de très petits problèmes, pour lesquels le voisinage tout entier peut être enfermé dans TL .

Au cours de sa progression, une méthode taboue échappe ainsi aux minima locaux : même si s est un minimum local, l'heuristique va s'échapper de la région $V(s)$ en empruntant un *col*. En début de calcul, la méthode trouve une suite de solutions améliorées, comme une recherche locale. On voit ensuite le coût osciller, puis redescendre vers un meilleur minimum local. Les améliorations deviennent de plus en plus rares au cours des itérations.

b) Liste taboue TL

Glover a montré qu'une liste taboue de taille $NT = 7$ à 20 suffit en pratique pour empêcher l'algorithme de boucler en revenant sur une solution déjà visitée. TL fonctionne donc comme une sorte de *mémoire à court terme*. A chaque itération, la NT -ième solution de TL (la plus ancienne) est écrasée par la dernière solution examinée. En pratique, TL se gère simplement avec une structure de *fichier (file)*.

Une technique simple pour gérer la liste taboue est d'interdire de repasser par les NT dernières valeurs de la fonction objectif : il suffit de stocker uniquement le coût entier des NT dernières solutions, les résultats peuvent être honorables, sauf si la fonction objectif prend relativement peu de valeurs différentes.

1.6 Approche proposée : la méthode hybride Recherche Taboue – Recuit Simulé RT/RS

L'approche proposée est une hybridation entre le recuit simulé (RS) et la recherche taboue (RT). Cette hybridation nous permettra de profiter de l'avantage de la méthode Recuit Simulé: acceptation des « mauvaises » solutions ce qui permet alors d'explorer une plus grande partie de l'espace de recherche et tend à éviter de s'enfermer trop vite dans la recherche d'un optimum local, et l'avantage de la Recherche Taboue qui consiste à interdire

(d'où le nom de *tabou*) de revenir sur les dernières positions explorées puisque les positions déjà explorées sont conservées dans une *Liste Taboue* d'une taille donnée qui est un paramètre ajustable de l'heuristique.

1.6.1 Codage de la solution

La solution est représentée par une matrice de dimension $[N \times N_t]$,

$$S = \begin{pmatrix} U_1(1) & U_1(2) & \dots & U_1(t) & \dots & U_1(N_t) \\ U_2(1) & U_2(2) & \dots & U_2(t) & \dots & U_2(N_t) \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ U_i(1) & U_i(2) & \dots & U_i(t) & \dots & U_i(N_t) \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ U_N(1) & U_N(2) & \dots & U_N(t) & \dots & U_N(N_t) \end{pmatrix}$$

Avec :

$U_i(t)$ est l'état de l'unité i à l'instant t : un 0 signifie que la centrale est à l'arrêt, un 1 quelle est en fonctionnement.

1.6.2 Création de la solution initiale

Les méthodes de parcours commencent toujours par le calcul d'une solution initiale, cette section présente la technique utilisée pour calculer une solution initiale. Cette technique assure de commencer toujours par la même solution initiale ce qui réduit considérablement l'écart entre les résultats, de plus cette solution est réalisable et proche de la solution optimale.

D'abord on doit créer une liste qui met en ordre décroissant les charges $P_D(t)$, $t = 1, \dots, N_t$, on l'appelle *Pcharge*.

Ensuite, on doit créer une liste de priorité d'unités *PL*, cette liste met en ordre décroissant le coût de production des unités en pleine charge [11]:

$$\left(a_i + b_i P_i^{max} + c_i P_i^{max^2} \right) / P_i^{max} \quad (1.26)$$

L'algorithme qui génère cette solution initiale est comme suit :

Début

Initialiser la S_0 de taille $[N \times N_t]$ dont tous les éléments sont égaux à 1

Pour Temps = 1 : N_t

Pour Unité = 1 : N

Unité = $PL(\text{Unité})$, et Temps = $Pcharge(\text{Temps})$

Si $S_0(\text{Unité}, \text{Temps}) = 1$ alors

```

Éteindre l'unité  $S_0(\text{Unité}, \text{Temps})=0$ ,
Si les contraintes (1.19), (1.22), (1.23), (1.24) et (1.25) violées
alors
Redémarrer l'unité :  $S_0(\text{Unité}, \text{Temps})=1$ 
FinSi
FinSi
FinPour
FinPour
Fin

```

1.6.3 Diversification (perturbation aléatoire) [31]

a) Phase I : Perturbation

La section suivante présente comment effectuer une perturbation aléatoire sur la solution actuelle S , cette technique nous assure *la diversification* sur l'espace de recherche.

```

Début
Choisir unité  $\sim U(1 \dots N)$ , et temps  $\sim t(1 \dots N_t)$ 
Si  $S(\text{unité}, \text{temps})=0$  alors
Trouver quand l'unité a été éteinte  $t_{OFF}$ 
Trouver quand l'unité a été redémarrée  $t_{ON}$ 
Durée  $= t_{OFF} - t_{ON}$ 
Si Durée  $= MDT_{\text{unité}}$  alors
Pour  $t = t_{OFF}$  jusqu'à  $t_{ON} - 1$ 
 $S(\text{unité}, t) = 1$ 
FinPour
Sinon
Si rand  $< 0.5$  alors
 $S(\text{unité}, t_{OFF}) = 1$ 
Sinon  $S(\text{unité}, t_{OFF}) = 0$ 
FinSi
Pour  $t = t_{OFF} + 1$  jusqu'à  $t_{ON} - 1$ 
Si  $S(\text{unité}, t-1) = 1$  et  $X_i^{ON} \geq MUT_{\text{unité}}$ 
Si rand  $< 0.5$  alors
 $S(\text{unité}, t) = 1$ 
Sinon

```

```

    Si  $t \leq t_{ON} - MDT_{unité}$ 
         $S(unité,t)=0$ 
    Sinon
         $S(unité,t)=1$ 
    FinSi
    FinSi
    Sinon  $S(unité,t-1)=1$ 
    FinSi
    Si  $S(unité,t-1)=0$  et  $X_i^{OFF} \geq MDT_{unité}$ 
        Si  $rand < 0.5$  alors
             $S(unité,t)=1$ 
        Sinon  $S(unité,t)=0$ 
        FinSi
    Sinon  $S(unité,t-1)=0$ 
    FinSi
    FinPour
    FinSi
    ...
    
```

(rand génère un nombre aléatoire inclus entre 0 et 1)

Et pour $S(unité,temps)=1$, on applique le même algorithme avec un changement de 0 par 1, MDT par MUT , X^{OFF} par X^{ON} etc.

La Figure 1.6 présente un exemple d'une perturbation aléatoire effectuée sur la solution initiale avec $MDT_i=2$ heures et $MUT_i=2$ heures.

Unité\Temps						t				
i	1	1	0	0	0	1	1	0	0	0
						↓				
i	1	1	0	0	0	0	0	0	0	0

Cas I Durée= MUT_i

Unité\Temps						t				
i	0	0	1	1	1	1	1	1	1	0
						↓				
i	0	0	1	1	0	0	0	1	1	0

Cas II Durée $\geq MUT_i$

Figure 1.6 Exemples de perturbation

b) Phase II : Réparation

Après avoir effectué une perturbation sur la solution initiale, la nouvelle solution peut violer quelques contraintes, et il faut rendre cette solution réalisable (Réparation). Par exemple si la contrainte (1.19) est violée (Figure 1.7), c'est-à-dire qu'à l'instant t , la puissance maximale des unités allumées à cette heure est inférieure à la somme de la charge et la réserve. Alors, on a besoin d'allumer plus d'unités à cette heure et le choix des unités à allumer est déterminé suivant la liste de priorité.

Unité\Temps			$t-2$	$t-1$	t	$t+1$		
i	0	0	0	1	1	1	0	0
j	0	0	0	0	0	1	1	1
					↓			
i	0	0	0	1	1	1	0	0
j	0	0	0	0	1	1	1	1

Figure 1.7 Exemple de réparation de la contrainte (1.19)

De plus après avoir effectué la phase I et la réparation précédente, les contraintes (1.22) et (1.23) peuvent être violées, il faut aussi les réparer, la Figure 1.8 et la Figure 1.9 illustrent des exemples de réparation de ces contraintes avec $MUT_i=2heures$ et $MDTi=3heures$.

Unité\Temps			$t-2$	$t-1$	t	$t+1$		
i	0	0	0	0	1	0	0	0
					↓			
i	0	0	0	0	1	1	0	0

Figure 1.8 Exemple de réparation de la contrainte (1.22)

Unité\Temps			$t-2$	$t-1$	t	$t+1$		
i	1	1	1	0	0	1	1	1
					↓			
i	1	1	1	0	0	0	1	1

Figure 1.9 Exemple de réparation de la contrainte (1.23)

1.6.4 Intensification (recherche locale) [31]

Le voisinage d'une solution est un sous-ensemble de solutions qu'il est possible d'atteindre par une série de transformations données. Et la recherche locale consiste à déterminer à partir de cette sous ensemble la meilleure solution (optimum local). Dans cette section, nous présentons la procédure suivie pour la recherche locale pour assurer une convergence plus rapide.

La procédure de la recherche locale (Voisinage) proposée suit presque le même algorithme proposé dans la référence [31]. La seule différence est que le choix de l'unité à éteindre est déterminé suivant la liste de priorité des unités $PL(t)$ tandis que le choix du temps est fait suivant la liste de priorité des charges $Pcharge$. L'algorithme suivant présente la procédure suivie pour l'intensification dans l'approche proposée :

1. *Début*
2. *Initialiser $t=1, i=1$*
3. *$t = Pcharge(t), i=PL(i)$*
4. *Si $S(i,t)=1$ alors*
5. *$S(i,t)=0$*
6. *Si les contraintes (1.19), (1.22), (1.23), (1.24) et (1.25) violées alors*
7. *Redémarrer l'unité $S(i,t)=1$ Aller à 15*
8. *Sinon*
 Comparer la réduction du coût de
 production à l'heure t due à l'extinction
 de l'unité $i : C_1$, et l'augmentation
 du coût C_2 due à l'allumage de l'unité i
 à cette heure.
9. *Si $C_2 > C_1$ alors*
10. *Restorer l'unité $i S(i,t)=1$*
11. *Sinon*
12. *Aller à 15*
13. *FinSi*
14. *FinSi*
15. *Si $i < Taille(PL(t))$ alors $i=i+1$ retourner à 3*
16. *Sinon Aller à 17 FinSi*
17. *Si $t < N_t$ alors $t=t+1, i=1$ retourner à 3*
18. *Sinon Aller à 19 FinSi*
19. *Fin*

Cette méthode progresse itérativement et alternativement entre les phases de diversification, d'intensification et d'apprentissage, ce processus peut être schématisé par la Figure 1.10, et l'algorithme général de recherche sera présenté dans le paragraphe suivant.

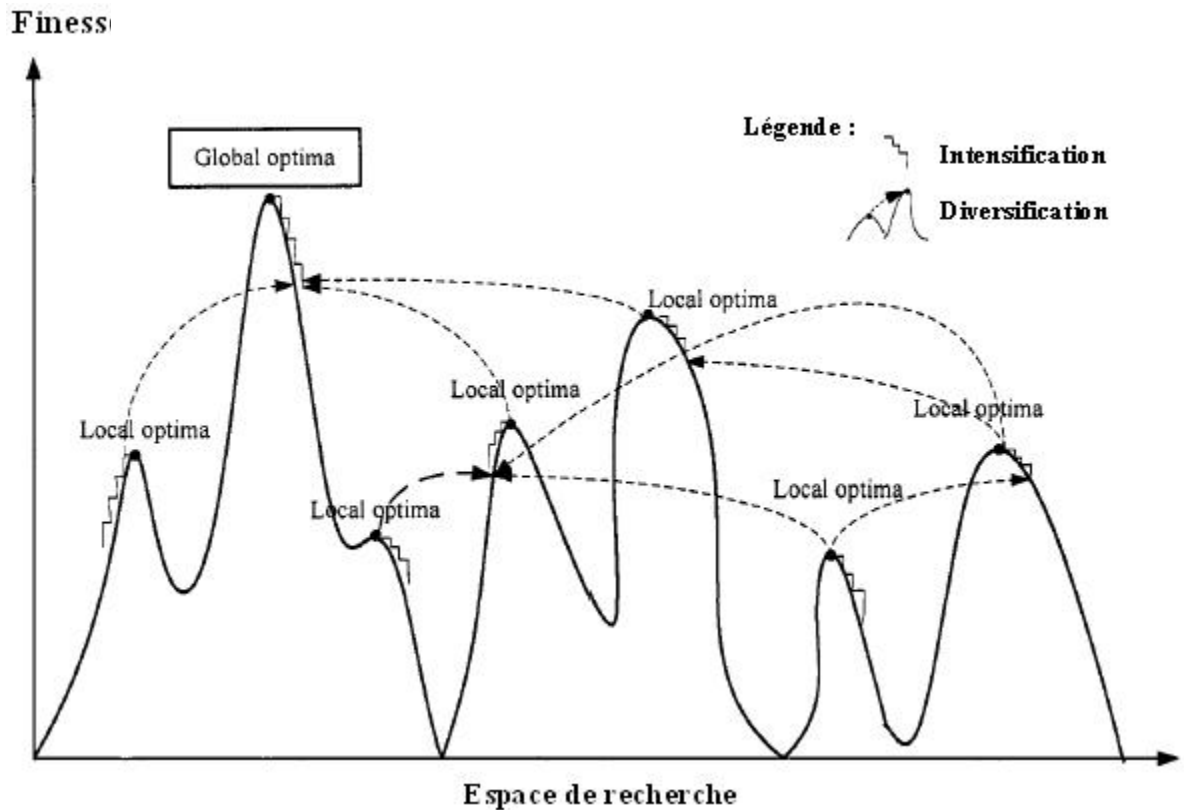


Figure 1.10 Processus de recherche (RT/RS)

1.6.5 Algorithme

La Figure 1.11 illustre l'algorithme général de la méthode hybride recuit simulé et recherche taboue, avec TL : la liste taboue, T : la température, S_p est la solution issue du processus de perturbation, S_L est la solution issue du processus de l'intensification, τ : le facteur de décroissement de la température est couramment choisi 0.99, la température initiale T_0 ainsi que la taille maximale de la liste taboue N_{max} sont à déterminer empiriquement.

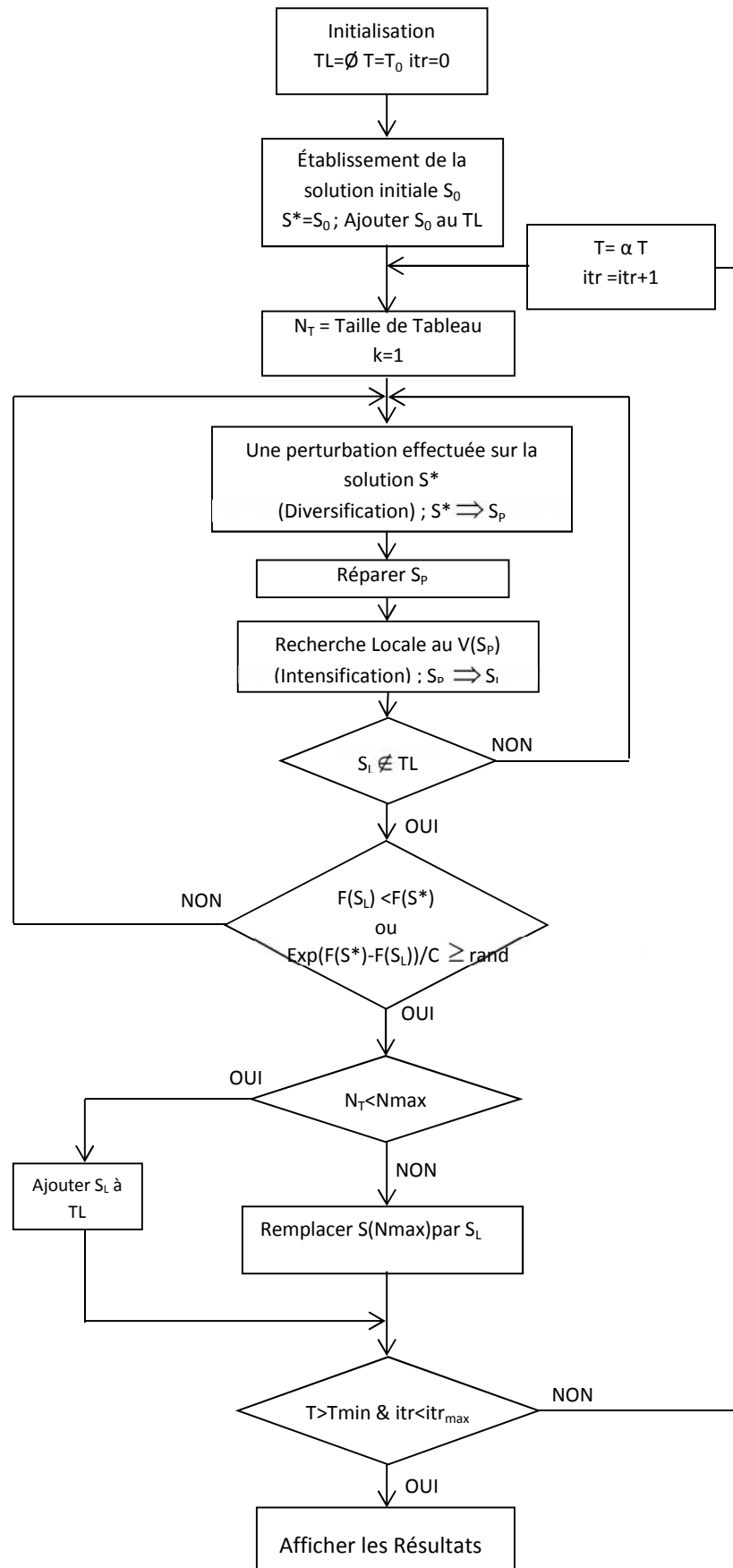


Figure 1.11 Algorithme de la méthode hybride RT/RS

1.7 Conclusion

Dans ce chapitre, nous avons exposé la formulation mathématique de la fonction objective et les différentes contraintes du problème d'engagement des turbines (UCP). Pour résoudre le problème de l'UCP, nous avons proposé d'utiliser deux métaheuristiques de base, le recuit simulé et la recherche taboue ainsi qu'une hybridation de ces deux métaheuristiques. Cette dernière accepte une solution plus mauvaise dans l'espoir qu'à partir de celle-ci on puisse trouver une solution encore meilleure et interdit de reprendre des solutions récemment visitées ce qui nous assure de ne pas boucler sur les mêmes solutions.

Chapitre 2

Reconfiguration des Réseaux de Distribution en Présence des DGs

Chapitre 2: Reconfiguration des Réseaux de Distribution en Présence des DGs

2.1 Introduction

Le réseau de distribution n'a pas été conçu, à l'origine, pour accueillir des unités de production mais pour acheminer, de manière unidirectionnelle, l'électricité qui provient du réseau de répartition jusqu'aux consommateurs moyenne et basse tension. L'objectif de ce chapitre est de présenter deux applications importantes intégrées dans la version actuelle logiciel, à savoir l'optimisation de l'emplacement et de la taille des générateurs dispersés (DGs) dans les réseaux de distribution, et la reconfiguration optimale des réseaux en présence des DGs.

Ce chapitre s'articule autour de quatre parties. Dans la première partie, on donnera des notions générales sur l'écoulement de puissance dans les réseaux de distribution et la technique de résolution du problème en utilisant la méthode ascendante/descendante et la stabilité de tension. Dans la deuxième partie du chapitre, on s'intéressera aux méthodes d'optimisation exploitant les algorithmes métaheuristiques les plus utilisés dans le domaine de l'électrotechnique et plus particulièrement l'algorithme génétique multi objectif (NSGA-II) employé dans toutes nos études optimales. La troisième partie du chapitre est consacré premièrement à l'optimisation de l'emplacement et de la taille des (DGs) dans les réseaux de distribution, et enfin à la reconfiguration optimale des réseaux en présence des DGs.

2.2 Notions de base

2.2.1 Ecoulement de puissance

Le calcul d'écoulement de puissance consiste à déterminer les grandeurs d'état associées aux éléments d'un réseau électrique. Ce calcul est une étape importante, car les informations obtenues constituent une base de départ pour les études d'optimisation, de

stabilité, de planification, etc. L'état électrique d'un nœud est caractérisé par deux grandeurs complexes, la puissance apparente S et la tension U , soit quatre grandeurs réelles: les puissances active P et réactive Q , le module de la tension U et son déphasage par rapport à une référence. A partir des grandeurs d'état des nœuds, la circulation des puissances ou des courants de chaque branche peut être déterminée.

Comparés aux réseaux de transmission d'énergie électrique, les réseaux de distribution sont caractérisés par leurs configurations radiales et un rapport de la résistance de branche à sa réactance (r/X) élevé. Ces derniers caractères font que leurs matrices expressions sont mal conditionnées et rendent donc, la solution de l'écoulement de puissance par les méthodes Newtoniennes ou leurs dérivées [32-36] impossible (divergences fréquentes de ces méthodes). Même si ces méthodes venaient à converger, la construction de la matrice de Jacobi prend un temps de calcul relativement long, ce qui ne permet pas leur utilisation en temps réel. De plus, dans l'étude de la rationalisation du transit de l'énergie réactive, le calcul de l'écoulement de puissance étant répété plus d'une fois, il devient utile et nécessaire même de réduire le temps de calcul du mieux que l'on peut. Afin de déterminer les puissances transitant dans ces lignes ainsi que les courants de branche et les tensions au niveau de leurs différents jeux de barres, la voie la plus indiquée dans ce cas est l'application des lois régissant les circuits électriques. Ce type de méthodes est basé sur le balayage en montée (backward sweep) et le balayage en descente (forward sweep) de la ligne [37].

Considérons un modèle de circuit équivalent d'une branche typique entre les nœuds i et $i+1$ d'un réseau de distribution radial, comme le montre la Figure 2.1.

Dans la Figure 2.1, $V_i \angle u_i$ et $V_{i+1} \angle u_{i+1}$ sont les modules de tension et les angles de phase de deux nœuds i et $i+1$, respectivement, $r_{i,i+1}, x_{i,i+1}$ sont la résistance et la réactance de la branche, respectivement, et le courant traversant la branche est $I_{i,i+1}$.

Le courant circulant dans la branche entre les nœuds p et q est donné par :

$$I_{i,i+1} = \frac{V_i \angle u_i - V_{i+1} \angle u_{i+1}}{r_{i,i+1} + jx_{i,i+1}} \quad (2.1)$$

ou

$$I_{i,i+1} = \frac{P_i - jQ_i}{V_i \angle -u_i} \quad (2.2)$$

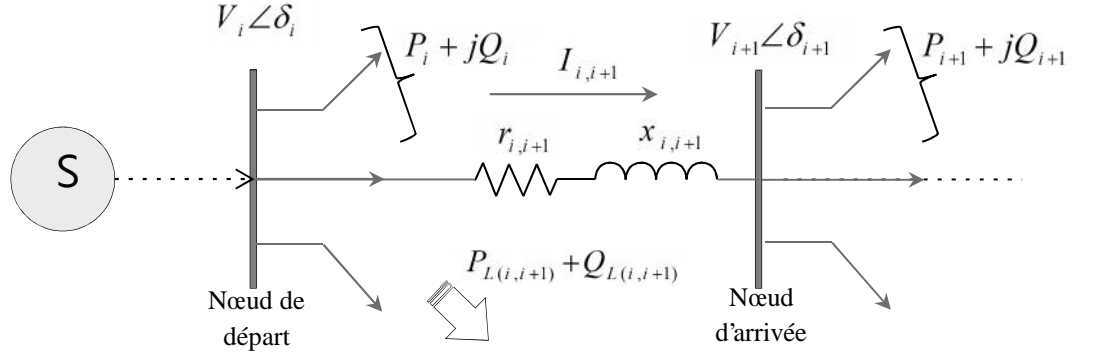


Figure 2.1 Schéma équivalent d'une branche

Les pertes actives et réactives dans la branche sont données par :

$$\begin{cases} P_{L(i,i+1)} = r_{i,i+1} \frac{(P_i^2 + Q_i^2)}{V_i^2} \\ Q_{L(i,i+1)} = x_{i,i+1} \frac{(P_i^2 + Q_i^2)}{V_i^2} \end{cases} \quad (2.3)$$

Les puissances active et réactive au début de la branche sont données par:

$$\begin{cases} P_i = P_{i+1} + P_{L(i,i+1)} \\ Q_i = Q_{i+1} + Q_{L(i,i+1)} \end{cases} \quad (2.4)$$

Avec :

- P_i est la somme des puissances actives de tous les nœuds derrière le nœud i , plus la puissance active consommée dans le nœud i , plus la somme des pertes actives de toutes les branches derrière le nœud i .
- Q_i est la somme des puissances réactives de tous les nœuds derrière le nœud q , plus la charge réactive consommée dans le nœud i , plus la somme des pertes réactives de toutes les branches derrière le nœud i .

De l'équation (2.1) et (2.2), on peut écrire:

$$I_{i,i+1} = \frac{V_i \angle u_i - V_{i+1} \angle u_{i+1}}{r_{i,i+1} + jx_{i,i+1}} = \frac{P_i - jQ_i}{V_i \angle -u_i} \quad (2.5)$$

i.e. :

$$V_i^2 - V_i V_{i+1} \angle u_{i+1} - u_i = (P_i - jQ_i)(r_{i,i+1} + jx_{i,i+1}) \quad (2.6)$$

Séparant la partie réelle et imaginaire :

$$\begin{cases} V_i V_{i+1} \cos(u_{i+1} - u_i) = V_i^2 - (P_i r_{i-1,i} + Q_i x_{i-1,i}) \\ V_i V_{i+1} \sin(u_{i+1} - u_i) = Q_i r_{i-1,i} - P_i x_{i-1,i} \end{cases} \quad (2.7)$$

A partir de l'équation de l'équation (2.7), on peut déduire :

$$V_{i+1} = \left\{ V_i^2 - 2(P_i r_{i+1} + Q_i x_{i+1}) + (r_{i+1}^2 + x_{i+1}^2) \frac{(P_i^2 + Q_i^2)}{V_i^2} \right\}^{1/2} \quad (2.8)$$

Et l'angle correspondant:

$$u_{i+1} = u_i + \tan^{-1} \left\{ \frac{Q_i r_{i,j+1} - P_i x_{i,j+1}}{V_i^2 - (P_i r_{i,j+1} + Q_i x_{i,j+1})} \right\} \quad (2.9)$$

La solution de l'écoulement de puissance dans les réseaux de distribution par balayage en montée et en descente est exécutée en suivant l'algorithme présenté sur la Figure 2.2.

Lire les données du réseau électrique.

Initialiser toutes les tensions des nœuds à 1.0 p.u et les pertes dans les branches à 0.0 p.u

Tant que le critère d'arrêt n'est pas satisfait faire

L'étape ascendante : Calculer pour toutes les branches, les pertes de puissances actives et réactives selon (2.3) ainsi que les puissances active et réactive au début de la branche selon (2.4).

L'étape descendante: Calculer la tension et la phase du nœud à l'arrivée de chaque branche selon (2.8) et (2.9).

FinTanque

Afficher les résultats

Figure 2.2 Algorithme de l'écoulement de puissance

2.2.2 La Stabilité de Tension

On définit la stabilité de la tension comme la capacité d'un réseau de maintenir une tension de barre constamment acceptable à chaque nœud du réseau, dans des conditions nominales de fonctionnement, après avoir subi une perturbation. L'état du réseau est dit

instable en tension lorsqu'une perturbation, un accroissement de la charge ou une modification de la condition du réseau entraîne une chute progressive et incontrôlable de la tension, aboutissant en un effondrement généralisé de la tension.

Beaucoup de critères ont été développés pour étudier le problème de stabilité de tension qui correspond à un point d'équilibre stable dans les réseaux électriques. Le critère de stabilité de tension classique le plus connue est celui de rapport de dérivée de la puissance réactive injectée par rapport au niveau de tension aux jeux de barre de contrôle ($\partial Q/\partial V$). Un autre critère ($\partial Q_G/\partial Q_L$) qui est basé sur la relation entre la puissance réactive générée $Q_G(V)$ et la puissance réactive demandée $Q_L(V)$ est utilisé souvent pour envisagé ce problème [38-42].

Dans notre travail, on s'intéresse à étudier la stabilité de tension en appliquant l'indice appelé (*SI Index*) proposé dans la référence [43]. L'indice *SI* donne une image totale de tensions au niveau des jeux de barres de charges, ces valeurs varient dans l'intervalle [0.1], le jeu de barres ayant la valeur de *SI* la plus basse sera le jeu de barre le plus critique dans ce système. L'indice de stabilité *SI* pour le nœud ($i+1$) est donnée par [43]:

$$SI_{i+1} = V_i^4 - 4 \cdot (P_i x_{i,j+1} - Q_i r_{i,j+1})^2 - 4 \cdot (P_i r_{i,j+1} + Q_i x_{i,j+1})^2 \cdot V_i^2 \geq 0 \quad (2.10)$$

avec $i = 2, 3, \dots, N_{bus}$

En fonctionnement stable, la valeur de *SI* doit être supérieure à zéro pour tous les nœuds. Le nœud ayant la valeur minimale de *SI* est le plus sensible à l'écroulement de tension.

2.2.3 Courbe d'effondrement de tension

La courbe d'effondrement de tension (Figure 2.3) montre la sensibilité et la variation de la tension du nœud critique en fonction de l'augmentation de la charge.

$$\begin{cases} P_{L,i} = P_{L0,i} (1 + \lambda) \\ Q_{L,i} = Q_{L0,i} (1 + \lambda) \end{cases} \quad (2.11)$$

pour $i = 2, 3, \dots, N_{bus}$

Avec $P_{L0,i}$ et $Q_{L0,i}$ sont, *respectivement*, les puissances actives et réactives du nœud i et λ est le facteur de charge.

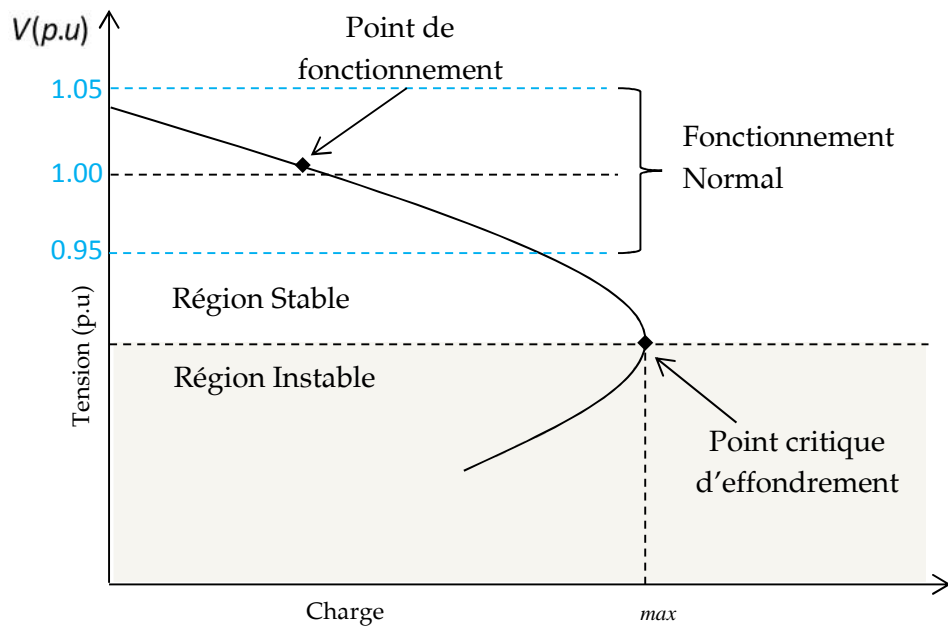


Figure 2.3 Courbe d'effondrement de tension

2.2.4 La production décentralisée

La dérégulation et l'ouverture du marché d'électricité ont créé un cadre favorable pour l'apparition et le développement de petits producteurs, connectés aux réseaux électriques dans une manière décentralisée. La production décentralisée consiste à générer localement dans les zones de consommation des quantités d'électricité relativement faibles par rapport aux unités classiques de production (thermiques, nucléaires, hydrauliques, etc.). Les principales différences de la production décentralisée par rapport aux grandes centrales de production consistent dans la localisation et la taille. Les sources décentralisées de petite taille sont connectées, en général, aux réseaux de distribution. Elles ne constituent pas seulement une réserve pour l'alimentation depuis les grands réseaux de transport ou de répartition, en pouvant être une substitution pour ceux-ci.

La production décentralisée basée sur l'utilisation de sources d'énergies primaires renouvelables et de la cogénération afin d'augmenter le rendement énergétique des installations de production, ce qui contribue à produire une énergie plus propre en développant une nouvelle catégorie de producteurs : la génération d'énergie dispersée (DG).

Les DG peuvent être classées en quatre grands types basés sur la capacité de fournir de la puissance active et réactive.

- a. **Type 1:** Ce type de DG est capable de fournir que la puissance active tels que les photovoltaïques, les micro-turbines, les piles à combustible, qui sont intégrés au réseau à l'aide des convertisseurs / onduleurs.

- b. Type 2:** DG capable de délivrer des puissances active et réactive. Les unités à base de machines synchrones (cogénération, turbines à gaz, etc) font partie de ce type.
- c. Type 3:** DG capable de fournir que la puissance réactive. Les compensateurs synchrones tels que les turbines à gaz sont un exemple de ce type fonctionnant avec facteur de puissance nul.
- d. Type 4:** DG capable de délivrer une puissance active en consommant de la puissance réactive. Principalement les générateurs à induction, qui sont utilisés dans des parcs éoliens, entrent dans cette catégorie.

2.3 Emplacement et dimensionnement optimal de DGs

Les DGs doivent être installées de façon à réduire les pertes de puissance en lignes et l'appel de puissance active et réactive au niveau du poste source et améliorer la stabilité de tension du réseau électrique. En plus et afin de ne pas avoir un effet inverse sur le plan de tension et les pertes de puissance, il faut que la puissance délivrée par chaque DG installée soit optimale pour équilibrer entre la puissance demandée et la puissance générée pendant les différentes périodes de charges. Donc notre problème consiste à optimiser aussi les tailles des DGs installées dans le réseau de distribution. Il ressort de l'analyse bibliographique [44-57] que de nombreuses tentatives ont été faites pour résoudre le problème d'optimisation reliées à la taille et le placement des DGs. La plupart des méthodes ont été proposées pour réduire les pertes de puissance active du réseau de distribution. Peu de travaux ont considéré la stabilité de tension qui est également une préoccupation majeure des systèmes d'énergie. Dans notre travail, on a proposé une méthode multi-objective NSGA-II qui prend en considération les deux objectifs (minimisation des pertes actifs et amélioration de la stabilité de tension).

2.3.1 Fonctions objectifs

a) Pertes de puissance actives

La première fonction objective consiste à minimiser de la somme des pertes de puissance active:

$$f_1 = P_{loss} = \sum_{i=1}^{N_{bus}-1} P_{L(i,i+1)} = \sum_{i=1}^{N_{bus}-1} r_{i,i+1} \frac{(P_i^2 + Q_i^2)}{V_i^2} \quad (2.12)$$

b) Stabilité de tension

La deuxième fonction objective consiste à améliorer de la stabilité de tension du réseau électrique:

$$f_2 = \frac{1}{1 + SI_{\min}} \quad (2.13)$$

Avec SI_{\min} est la valeur minimale de l'indice de stabilité SI de tous les jeux de barres.

2.3.2 Contraintes égalités

Ces contraintes sont l'image des lois physiques gouvernant le système électrique. Elles sont représentées par les équations non linéaires de l'écoulement de puissance. Il faut que la somme des puissances active et réactive injectées dans chaque jeu de barres soit égale à zéro.

$$\begin{cases} P_{Gi} - P_{Di} - V_i \sum_{j=1}^{N_{bus}} V_j \cdot Y_{ij} \cdot \cos(\theta_{ij} + u_j - u_i) = 0 \\ Q_{Gi} - Q_{Di} + V_i \sum_{j=1}^{N_{bus}} V_j \cdot Y_{ij} \cdot \sin(\theta_{ij} + u_j - u_i) = 0 \end{cases}, \quad (2.14)$$

$$i = 1, 2, \dots, N_{bus}$$

2.3.3 Contraintes inégalités

Les contraintes d'inégalités représentent généralement les limites de fonctionnement (limites physiques) du réseau.

- a. Limites de génération des DGs

$$0 \leq \sum_{i=1}^{N_{DG}} P_{DG_i} \leq \sum_{i=1}^{N_{bus}} P_{D_i} \quad (2.15)$$

- b. Limites de tension des jeux de barres

$$V_i^{\min} \leq V_i \leq V_i^{\max}, \quad i = 2, 3, \dots, N_{bus} \quad (2.16)$$

- c. Limites thermiques des lignes

$$|S_{li}| \leq S_{li}^{\max}, \quad i = 2, 3, \dots, N_{bus} \quad (2.17)$$

2.3.4 Manipulation des contraintes

Les techniques de pénalisation [58] permettent de prendre en compte les violations des contraintes sur les variables continues. Une des manières souvent utilisée dans de pareils cas consiste à pénaliser les fonctions objectifs (qui mesurent la qualité d'une solution) en réécrivant l'équation (II-9). La relation (II-29) permet en effet de s'éloigner des solutions dont les termes de pénalité sont non nuls.

$$\begin{aligned}
f_n^{Penalized} = f_n &+ \sum_{i=1}^{N_{bus}} K_v (V_i - V_i^{lim})^2 \\
&+ \sum_{i=1}^{N_{line}} K_s (S_{Li} - S_{Li}^{lim})^2 \\
&+ \sum_{i=1}^{N_{line}} K_D \left(\sum_{i=1}^{N_{DG}} P_{DG_i} - \sum_{i=1}^{N_{bus}} P_{Di} \right)^2
\end{aligned} \tag{2.18}$$

avec f_n est la $n^{\text{ème}}$ fonction objective, V_i^{lim} et S_{Li}^{lim} sont données par les expressions suivantes :

$$V_i^{lim} = \begin{cases} V_i^{\max} & \text{if } V_i > V_i^{\max} \\ V_i^{\min} & \text{if } V_i < V_i^{\min} \\ V_i & \text{if } V_i^{\min} \leq V_i \leq V_i^{\max} \end{cases} \tag{2.19}$$

$$S_{Li}^{lim} = \begin{cases} S_{Li}^{\max} & \text{if } S_{Li} > S_{Li}^{\max} \\ S_{Li}^{\min} & \text{if } S_{Li} < S_{Li}^{\min} \\ S_{Li} & \text{if } S_{Li}^{\min} \leq S_{Li} \leq S_{Li}^{\max} \end{cases} \tag{2.20}$$

K_v , K_s et K_D sont des facteurs de pénalité.

Pour le problème de la reconfiguration optimale, une fonction de pénalité est utilisée dans le cas de l'existence des mailles ou des charges isolés :

$$F_n^{Penalized} = K_m N_m + K_i N_i \tag{2.21}$$

où N_m est le nombre de mailles existantes, N_i est le nombre de charges isolées et K_m et K_i sont des facteurs de pénalité.

2.4 Reconfiguration du réseau de distribution en présence des DGs

Deux types d'interrupteur sont généralement connus dans le système électrique destinés pour la protection et la configuration du réseau. Les premiers sont les interrupteurs normalement fermés (sectionalizing switches) et les deuxièmes normalement ouverts (tie switches). Par le changement de l'état de ces interrupteurs, la configuration du réseau de distribution est changée et les charges sont transférées entre les lignes tant que la configuration radiale d'alimentation électrique reste maintenue. Cette implémentation est connue sous le nom de reconfiguration du système d'alimentation [59]. Plusieurs méthodes ont été proposée dans la littérature [57, 60-71] pour résoudre le problème de la reconfiguration (en présence ou en absence des DGs), dans a notre travail on a proposé une

méthode multi objectifs NSGA-II basée sur les techniques de la théorie des graphes pour résoudre le problème.

2.4.1 Ouverture et fermeture des interrupteurs (switches)

L'objectif principal de l'ouverture et la fermeture des interrupteurs (switches) et de réduire les pertes totales de puissance active et d'améliorer la stabilité de tension tout en maintenant les contraintes les contraintes de type égalité et inégalité (mentionnées précédemment) et assurant les deux conditions suivantes [65] :

- ✓ Le maintien de la structure radiale du réseau ;
- ✓ L'alimentation de toutes les charges connectées au réseau.

La fermeture de tous les interrupteurs dans le réseau de distribution radial va créer des boucles au sein du réseau déterminant les possibles éléments qui doivent être ouverts. Donc l'application de la structure radiale du réseau de distribution est faite par l'identification de ces boucles et s'assurer qu'un seul segment de ligne commutable soit toujours ouvert, ce qui ne permet que la génération de topologies admissibles délimitant ainsi l'espace de recherche à évaluer dans le processus de configuration.

Donc, chaque boucle identifiée détermine parmi tous ses éléments, l'élément à déconnecter pour former un réseau radial. La Figure 2.4 représente un réseau constitué de trois boucles où les composants de chacune d'elles sont déterminés comme suit [66]:

Boucle 1 = [L₂ L₄ L₅]

Boucle 2 = [L₁ L₃ L₄]

Boucle 3 = [L₅ L₆ L₇]

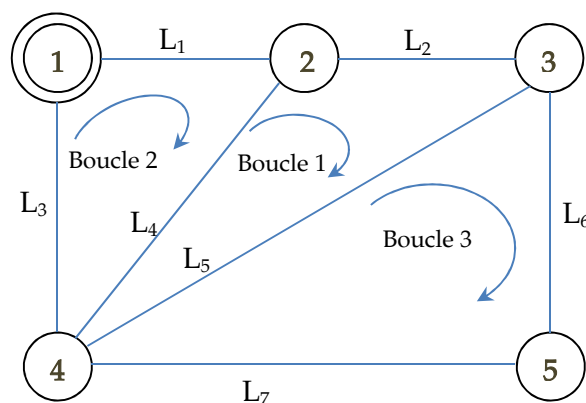


Figure 2.4 Réseau constitué de trois boucles

Ainsi, pour construire une topologie radiale faisable, les éléments qui vont être déconnectés doivent être élus de l'ensemble de vecteurs sans éléments communs de boucles

fondamentales. Les combinatoires qui peuvent être obtenues à partir de ces vecteurs donnent toutes les configurations possibles du système.

2.4.2 Réseau de distribution et théorie des graphes

La configuration de réseau de distribution est modélisé comme un graphe non orienté $G = (V, E)$. Chaque sommet de V représente un nœud et chaque arête e dans E indique une branches entre les deux nœuds u et v .

Tel que mentionné précédemment, les réseaux de distribution sont maillés mais exploités avec une configuration radiale. Le fonctionnement radial signifie généralement que chaque nœud est alimenté à partir d'une seule source et il n'y a pas de boucles sur le réseau. Dans la théorie des graphes, la configuration radiale d'un réseau de distribution est l'équivalent d'un arbre couvrant (spanning tree). Un arbre couvrant d'un graphe est défini comme un sous-ensemble des arêtes qui connecte tous les sommets sans former de cycle [72]. Donc, le rôle de la reconfiguration est de trouver le meilleur arbre couvrant qui optimise l'ensemble des objectifs et satisfait toutes les contraintes du problème.

a) Vérification de la configuration radiale et de la connectivité

Pour trouver la meilleure configuration d'un réseau de distribution en utilisant l'algorithme NSGA-II, la théorie des graphes peut être utilisée pour vérifier que chaque solution candidate à une configuration radiale afin d'assurer l'alimentation de toutes les charges et éviter les courts-circuits. Dans cette section, nous proposons d'appliquer la méthode algébrique de Kirchhoff basée sur la matrice d'incidence A pour vérifier la configuration radiale des solutions.

La matrice d'incidence A consiste à définir des relations d'incidence entre les arêtes et les nœuds. On définit la matrice $A = A(G) = [a_{ij}]$, de taille $n \times br$ dont les éléments sont définis par :

$$a_{ij} = \begin{cases} 1, & \text{si le nœud } v_i \text{ est une extrémité de } e_j \\ 0, & \text{sinon} \end{cases} \quad (2.22)$$

Dans [73], il est prouvé que le rang de La matrice A est égale à la différence entre le nombre de sommets n et le nombre de composantes connexes* k .

* Un graphe non orienté sera dit connexe si pour tout nœud il existe un chemin pour atteindre tout autre nœud.

$$\text{rang}(A) = n - k \quad (2.23)$$

Nous pouvons maintenant énoncer l'algorithme (2.2) pour vérifier si le réseau est radiale et ne contient pas des nœuds isolés, le processus et le suivant:

1. Vérifier la relation entre le nombre de nœuds et le nombre des arêtes:

$$n = n_b + 1 \quad (2.24)$$

2. Vérifier que le nombre des composantes connexes égal à 1:

$$k = \text{rang}(A) - n = 1 \quad (2.25)$$

Algorithme 2.2

Procédure etat =IsRadial(G)

A=IncidenceMatrix(G) ;

Si (rang(A) = n-1 **et** br=n-1)

etat =vrai;

Sinon

etat =faux;

Fin

Fin

b) Trouver le nombre de mailles et de charges isolées

Le nombre de mailles (cycles) et le nombre de charges isolées existantes devrait être calculé pour être utilisée ultérieurement dans la fonction de pénalité. Calculer le nombre de charges isolées peut se faire par algorithme de parcours en profondeur (ou DFS, Depth First Search) [73].

L'algorithme de parcours en profondeur permettra de trouver les charges connectées à la source d'alimentation (nœud 1), et à la fin de la procédure, chaque nœud marqué *non visité* est considérée comme *isolé*. L'algorithme 2.3 décrivent l'algorithme de parcours en profondeur utilisé pour le de calcul de nombre de mailles et de charges isolées.

Algorithme 2.3	
<pre> Procédure nbr_mailles_et_isolés(G) nœud(1:n).visite= faux // marquer tous les nœuds // comme non visité vc = 1 // nœud de démarrage Branches_connectées = 0 DepthFirstSearch(vc) Nbr_isolés=trouver(nœud.visité== faux). Nbr_mailles= Branches_connectées +1-(n- Nbr_isolés) FinProcédure </pre>	<pre> Procédure DepthFirstSearch(vc) nœud(vc). visité=vrais Pour (chaque w adjacent à vc) Faire Si (nœud (w).visité= faux) Alors nœud (w).visité= vrais DepthFirstSearch(w) Branches_connectées++ FinSi FinPour FinProcédure </pre>

2.5 Algorithme NSGA II

2.5.1 Principe

L'Algorithme Génétique Multi Objectif que nous avons utilisé dans notre travail est le NSGA-II (Non-dominated Sorting Genetic Algorithm II) présenté et amélioré par Deb et al. [74]. C'est l'un des algorithmes les plus utilisés et les plus cités dans la littérature. L'algorithme NSGA-II est un algorithme évolutif multi-objectif, établissant les rapports de dominance* entre les individus et offrant une méthode de tri particulièrement rapide des chromosomes. Cet algorithme utilise la mesure du surpeuplement autour des individus pour assurer la diversité dans la population. Le principe de cet algorithme est illustré à la Figure 2.6.

Au début, une population initiale P^0 est générée aléatoirement, puis elle subit un tri en utilisant le concept de la non-domination. Chaque solution se voit affecter une force, ou rang, égal à son niveau de non-dominance (1 pour le meilleur niveau, 2 pour le niveau suivant, etc...). L'étape de reproduction consiste en un tournoi pour la sélection des parents. Deux individus de la population sont choisis aléatoirement dans la population, le tournoi est basé sur une comparaison de la domination sous contraintes des deux individus.

Pour une génération t donnée, on crée $R^t = P^t \cup Q^t$ étant la population enfants de la population précédente P^t (générés à partir des parents à travers les opérateurs de croisement

* On dit que le vecteur $x = (x_1, x_2, \dots, x_k)$ domine le vecteur $y = (y_1, y_2, \dots, y_k)$ si :

- x est au moins aussi bon que y sur tous les objectifs et,
- x est strictement meilleur que y sur au moins un objectif.

et de mutation), R^t inclut les individus de P^t , ce qui assure le caractère élitiste de l'algorithme NSGA-II. La population R^t contient $2N_{pop}$ individus (elle est composée de N_{pop} parents et N_{pop} enfants). R^t subit ensuite un tri en utilisant le concept de la non-dominance de Pareto. Les individus sont regroupés dans des fronts de non-dominance tels que F_1 représente les individus de rang 1, F_2 les individus de rang 2, ... etc. L'objectif suivant est de réduire le nombre d'individus de $2N_{pop}$ dans la population R^t pour obtenir une population P^{t+1} de taille N_{pop} . Si la taille de F_1 est inférieure à N_{pop} , alors tous les individus de F_1 sont conservés. Il en est de même pour les autres fronts tant que le nombre d'individus conservés ne dépasse pas la taille N_{pop} . Si l'on prend l'exemple de la Figure 2.6, les fronts F_1 et F_2 sont intégralement conservés mais la conservation du front F_3 va entraîner un dépassement de la taille N_{pop} de la population P^{t+1} . Il faut alors procéder à une sélection des individus de F_3 à conserver.

Dans ce cas l'algorithme NSGA-II fait intervenir un mécanisme de préservation de la diversité de la population basé sur l'évaluation de la densité des individus autour de chaque solution, à travers une procédure de calcul de «Distance de proximité». Une faible valeur de la Distance de proximité pour un individu correspond à un individu «bien entouré». On procède alors à un tri décroissant selon cette distance de proximité pour retenir des individus du front F_3 et éliminer ainsi les individus des zones les plus denses. On complète de cette façon la population P^{t+1} . Les individus ayant des valeurs extrêmes pour les critères sont également préservés par ce mécanisme, permettant ainsi de conserver les bornes extérieures du front de Pareto.

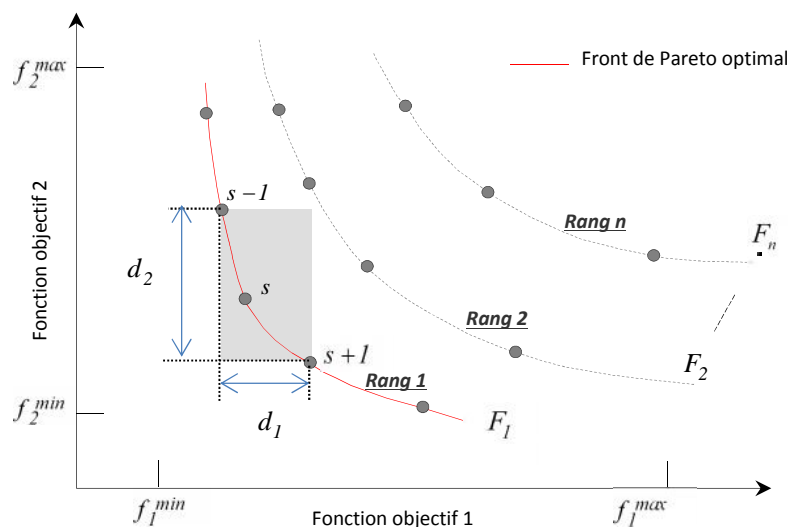


Figure 2.5 Classement des solutions dans les frontières de Pareto

À la fin de cette phase, la population P^{t+1} est créée. Puis une nouvelle population Q^{t+1} est générée par reproduction à partir de P^{t+1} . On poursuit itérativement la procédure décrite ci-dessus jusqu'à la satisfaction de critère d'arrêt fixé par l'utilisateur.

De point de vue général, le NSGA-II permet de maintenir l'élitisme et la diversité sans ajouter de paramètres supplémentaires, tout en utilisant un algorithme séduisant par sa simplicité avec un minimum de paramètres de réglage.

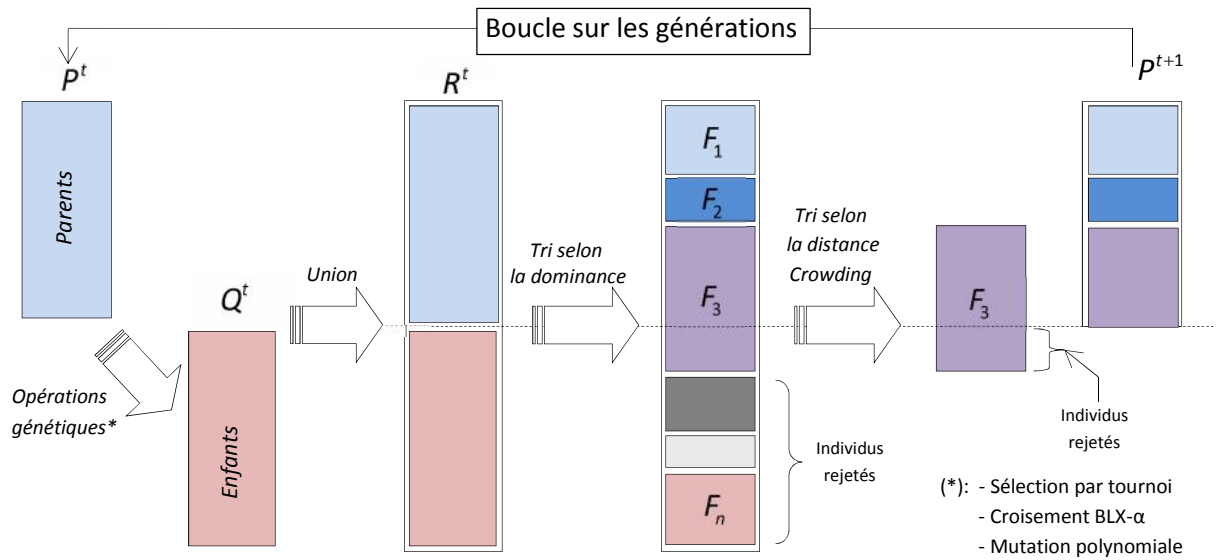


Figure 2.6 Principe de l'algorithme NSGA-II

2.5.2 Calcul de la distance de crowding

La distance de crowding d'une solution (s) (ou d'un individu) se calcule en fonction du périmètre formé par les points les plus proches de (s) sur chaque objectif. La Figure 2.5 montre une représentation à deux dimensions associée à la solution (s). Le calcul de la distance de crowding nécessite, avant tout, le tri des solutions selon chaque objectif, dans un ordre ascendant. Ensuite, pour chaque objectif, les individus possédant les valeurs limites (la plus petite et la plus grande valeur de fonction objectif) se voient associés une distance infinie (∞). Pour les autres solutions intermédiaires, on calcule une distance de crowding égale à la différence normalisée des valeurs des fonctions objectifs de deux solutions adjacentes. Ce calcul est réalisé pour chaque fonction objectif. La distance de crowding d'une solution est calculée en sommant les distances correspondantes à chaque objectif [74]:

$$d_s = \frac{d_1}{f_1^{\max} - f_1^{\min}} + \frac{d_2}{f_2^{\max} - f_2^{\min}} \quad (2.26)$$

avec f_j^{\max} et f_j^{\min} les minimums et maximums de la fonction objective j .

2.5.3 Choix de la meilleure solution de compromis

Parmi ces individus du front de Pareto optimal on choisit à l'aide de la logique floue le meilleur compromis. Ceci en modélisant chaque solution suivant chaque objectif par une fonction d'appartenance \sim_i défini par [75]:

$$\sim_i = \begin{cases} 1 & \text{si } f_i \leq f_i^{\min} \\ \frac{f_i^{\max} - f_i}{f_i^{\max} - f_i^{\min}} & \text{si } f_i^{\min} \leq f_i \leq f_i^{\max} \\ 0 & \text{si } f_i \geq f_i^{\max} \end{cases} \quad (2.27)$$

Pour chaque solution non dominée, nous normalisons la fonction d'appartenance associée par l'expression suivante :

$$\sim^k = \frac{\sum_{i=1}^m \sim_i^k}{\sum_{k=1}^D \sum_{i=1}^m \sim_i^k} \quad (2.28)$$

où D est le nombre de solutions non dominés et m est le nombre des fonctions objectifs. Le meilleur compromis est la solution dont la fonction d'appartenance normalisée est maximale.

2.6 Application de l'algorithme NSGA-II sur le problème l'optimisation de l'emplacement et de la taille des DGs

Le vecteur d'état correspondant à l'optimisation de l'emplacement et de la taille des DGs est donné comme suit :

$$X = [L_1 \ L_2 \ \dots \ L_{n_{DG}} \ P_1 \ P_2 \ \dots \ P_{n_{DG}} \ Q_1 \ Q_2 \ \dots \ Q_{n_{DG}}] \quad (2.29)$$

Où $P_1, P_2, \dots, P_{n_{DG}}$ et $Q_1, Q_2, \dots, Q_{n_{DG}}$: sont les puissances actives et réactives injectées par les DGs insérées respectivement aux emplacements $L_1, L_2, \dots, L_{n_{DG}}$; n_{DG} est le nombre des DGs installées. Si les DGs installées génèrent uniquement de la puissance active, c'est-à-dire, on ne prend pas en compte la puissance réactive générée, la taille du vecteur de contrôle devient $1 \times (2 \times n_{DG})$.

Les étapes de la méthode sont les suivants:

Etape 1: Générer aléatoirement une population initiale $P^0 = [X_1 \ X_2 \ \dots \ X_i \ \dots \ X_{N_{pop}}]$ prise de l'espace de recherche :

$$P_0 = [x_1 \ x_2 \ \dots \ x_{N_{pop}}] \quad (2.30)$$

$$L_i = \text{round}(2 + \text{rand}(N_{bus} - 2)) \quad (2.31)$$

$$P_i = P_{DG}^{\min} + \text{rand} \times (P_{DG}^{\max} - P_{DG}^{\min}) \quad (2.32)$$

$$Q_i = P_{DG}^{\min} + \text{rand} \times (P_{DG}^{\max} - P_{DG}^{\min}) \quad (2.33)$$

avec $\text{round}(\bullet)$ retourne la valeur d'un nombre arrondi à l'entier le plus proche, et rand indique un nombre aléatoire distribué uniformément entre 0 et 1.

Initialiser le compteur d'itération $t = 0$.

Etape 2: Faire le calcul de l'écoulement de puissance pour chaque individu et d'évaluer les fonctions d'objectif (f_1 et f_2) en utilisant les équations (2.12) et (2.13).

Etape 3: Vérifier les contraintes (2.15), (2.16) et (2.17) compte tenu des résultats de l'écoulement de puissance. Si les contraintes ne sont pas satisfaites, ajouter les fonctions de pénalité aux fonctions objectifs suivant l'équation (2.18).

Etape 4: Générer une nouvelle population d'enfants Q^t à partir de P^t en appliquant les opérateurs génétiques (Sélection par tournoi, croisement BLX- et mutation polynomiale [76]).

Etape 5: Combiner les populations de parents et d'enfants et créer $R^t = P^t \cup Q^t$.

Etape 6: Appliquer un tri non-domine sur R^t et identifier les différents fronts F_i , $i = 1, \dots, \text{etc.}$

Etape 7: Créer une nouvelle population $P^{t+1} = \emptyset$ Initialiser $i = 1$. Tant que $|P^{t+1}| + |F_i| < N_{pop}$ (la population n'est pas pleine), on fait $P^{t+1} = P^t + F_i$ et $i = i + 1$.

Etape 8: Lorsque toutes les solutions de F_i ne peuvent être intégrées à P^{t+1} on fait un tri sur les distances de crowding et on inclut à P^{t+1} les $N_{pop} - |P^{t+1}|$ solutions les plus éparpillées en utilisant la distance de crowding dans l'ensemble F_i .

Etape 9: Créer la population d'enfants Q^{t+1} à partir de P^{t+1} en utilisant les opérateurs génétiques (Sélection par tournoi, croisement BLX- et mutation polynomiale). Incrémenter le compteur des générations ($t = t + 1$).

2.7 Application de l'algorithme NSGA-II sur le problème de la reconfiguration optimale

Le vecteur d'état correspondant à la reconfiguration du réseau est donné comme suit :

$$X = [x_1, x_2, \dots, x_{NL}]_{(1 \times NL)} \quad (2.34)$$

Où :

x_1, x_2, \dots, x_{NL} sont les interrupteurs sélectionnés à être ouverts pour une nouvelle configuration pris des vecteurs formés par les boucles créés par la fermeture de tous les interrupteurs et que chacun d'eux peut être un interrupteur normalement fermé (sectionalizing switch) ou un interrupteur normalement ouvert (tie switch); NL est le nombre de vecteurs ou de boucles formées par la fermeture des interrupteurs (ties switches) comportant les branches destinées à être ouvertes afin de reconfigurer le réseau en lui gardant sa structure radiale et en assurant l'alimentation de toutes les charges connectées.

Les étapes de la méthode sont les suivants:

Etape 1: Identifier les boucles du réseau formées par la fermeture des tie switches.

Etape 2: Générer aléatoirement une population initiale $P^0 = [X_1 \ X_2 \ \dots \ X_i \ \dots \ X_{N_{pop}}]$ prise de l'espace de recherche imposé par les boucles formées. Initialiser le compteur d'itération $t = 0$.

Etape 3: Pour chaque individu de P^t , vérifier si le réseau est radiale et ne contient pas des nœuds isolés (algorithme 2.2). S'il satisfait, passer à l'étape suivante. Sinon, passez à l'étape 6.

Etape 4: Faire le calcul de l'écoulement de puissance pour chaque individu et d'évaluer les fonctions d'objectif (f_1 et f_2) en utilisant les équations (2.12) et (2.13).

Etape 5: Vérifier les contraintes (2.16) et (2.17) compte tenu des résultats de l'écoulement de puissance. Si les contraintes ne sont pas satisfaites, ajouter les fonctions de pénalité aux fonctions objectifs suivant l'équation (2.18). Passez à l'étape 7.

Etape 6: Calculer le nombre de mailles et de charges isolées (algorithme 2.3); ajouter les fonctions de pénalité aux fonctions objectifs suivant l'équation (2.21).

Etape 7: Générer une nouvelle population d'enfants Q^t à partir de P^t en appliquant les opérateurs génétiques (Sélection par tournoi, croisement BLX- et mutation polynomiale [76]).

Etape 8: Combiner les populations de parents et d'enfants et créer $R^t = P^t \cup Q^t$.

Etape 9: Appliquer un tri non-domine sur R^t et identifier les différents fronts F_i , $i = 1, \dots, etc$.

Etape 10: Créer une nouvelle population $P^{t+1} = \emptyset$ Initialiser $i = 1$. Tant que $|P^{t+1}| + |F_i| < N_{pop}$ (la population n'est pas pleine), on fait $P^{t+1} = P^t + F_i$ et $i = i + 1$.

Etape 11: Lorsque toutes les solutions de F_i ne peuvent être intégrées à P^{t+1} on fait un tri sur les distances de crowding et on inclut à P^{t+1} les $N_{pop} - |P^{t+1}|$ solutions les plus éparpillées en utilisant la distance de crowding dans l'ensemble F_i .

Etape 12: Créer la population d'enfants Q^{t+1} à partir de P^{t+1} en utilisant les opérateurs génétiques (Sélection par tournoi, croisement BLX- et mutation polynomiale). Incrémenter le compteur des générations ($t = t + 1$).

2.8 Conclusion

Dans le présent chapitre, deux applications relatives aux réseaux de distribution ont été présentés, à savoir l'optimisation de la taille et de l'emplacement des DGs, et la reconfiguration optimale des réseaux en présence des DGs. L'algorithme d'optimisation multi-objectif NSGA-II a été utilisé pour les deux applications afin d'optimiser deux fonctions objectifs considérées dans ce travail, à savoir la minimisation des pertes actifs et l'amélioration de la stabilité de tension tout en satisfaisant des contraintes de type égalité et inégalité.

Des techniques issues de la théorie des graphes ont été utilisées pour la reconfiguration optimale afin minimiser l'espace de recherche, assurer l'alimentation de toutes les charges connectées au réseau et maintenir la topologie radial des réseaux électriques de distributions.

Chapitre 3

Modélisation Orientée Objet des Réseaux Electriques

Chapitre 3: Modélisation Orientée Objet des Réseaux Electriques

3.1 Introduction

De nombreuses applications ont été développées ces dernières années en utilisant la Programmation orientée objet (POO). Cet engouement pour la POO se justifie largement aujourd'hui. En effet, les systèmes informatiques (logiciels) sont de plus en plus hétérogènes et plus complexes, de plus ils ont une durée de vie assez longue dans laquelle ils subissent des adaptations et des modifications (maintenance). La POO facilite la réutilisation, la flexibilité des programmes par la définition des classes et objets, et la protection des données par la notion d'encapsulation. On peut ajouter aussi d'autres méthodes à la classe de base par le principe d'héritage, et la combinaison et l'hybridation entre les différentes méthodes en utilisant l'association entre les objets des classes.

Il s'agit dans ce chapitre de présenter la POO et les différents mécanismes utilisés par cette approche et de mettre en évidence leurs intérêts dans le processus de développement des applications logicielles. Nous décrivons d'abord les principes et les mécanismes de base de l'orienté objets, ensuite les différentes méthodes qui existent et les différents modèles de représentation objets du standard UML (Unified Modeling Language). Enfin nous présentons les modèles orientés objets développés et utilisés dans cette thèse pour le développement des applications réseaux électriques.

3.2 Notions sur la modélisation orientée objet

En informatique, la modélisation consiste tout d'abord à décrire un problème, puis décrire la solution de ce problème ; ces activités s'appellent respectivement l'analyse et la conception. Un modèle est une description abstraite d'un système ou d'un processus qui permet de comprendre et de simuler. La mise en œuvre d'une approche orientée objet, permet de concevoir un programme comme un ensemble d'entités (*objets*) indépendantes, qui interagissent et communiquent par l'intermédiaire de messages en vue de réaliser une opération spécifique [77].

3.2.1 Notions de classe et d'objet

On définit « l'objet » comme une entité atomique du monde réel composée d'un état et d'un comportement. L'objet est donc une capsule qui contient ce que fait l'objet; les noms employés à l'intérieur n'interfèrent pas avec l'extérieur. Une classe d'objets décrit un groupe d'objets ayant des attributs similaires, des comportements identiques, des relations communes avec les autres objets. Il s'agit d'une description abstraite en termes de données et de comportements d'une famille d'objets. Les généralités sont contenues dans la classe et les particularités sont contenues dans les objets. Les objets informatiques sont construits à partir de la classe, par un processus appelé : *instanciation*.

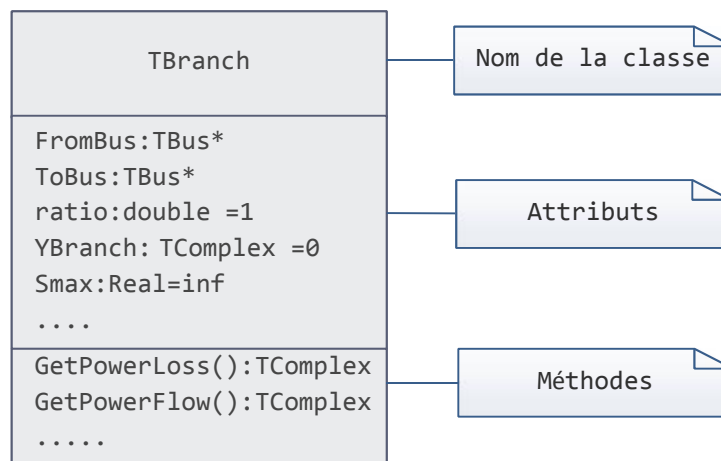


Figure 3.1 Exemple de représentation graphique d'une classe

Chaque classe est représentée sous la forme d'un rectangle divisé en trois compartiments (Figure 3.1). Le premier compartiment contient le nom de la classe, le second les attributs et le dernier les opérations. Par défaut, les attributs sont cachés et les opérations sont visibles. Les compartiments peuvent être supprimés pour alléger les diagrammes. La représentation graphique d'une classe permet de faire apparaître le niveau de visibilité des

attributs et méthodes au moyen des caractères +, # et qui correspondent respectivement aux niveaux *public*, *protégé*, et *privé*. En Delphi chaque classe possède deux méthodes implicites qui peuvent être surchargées : le *constructeur* d'objet et le *destructeur* d'objet. Le constructeur est une fonction qui initialise toutes les variables de la classe. Le destructeur est appelé implicitement pour libérer la mémoire allouée par un objet de cette classe.

3.2.2 Principes de la programmation orientée objet

Dans cette section, nous présentons certaines propriétés essentielles de l'approche orientée objet. En effet, seules les caractéristiques présentant un intérêt incontestable durant la phase d'analyse et de conception du programme, sont exposées ci-dessous.

a) Concept d'encapsulation

L'encapsulation consiste à masquer des attributs et des méthodes de l'objet vis-à-vis des autres objets. En effet, certains attributs et méthodes ont pour seul objectif des traitements internes à l'objet et ne doivent pas être exposés aux objets extérieurs. *Encapsulés*, ils sont appelés les attributs et méthodes privés de l'objet. La définition de l'encapsulation se fait au niveau de la classe. Le Delphi implémente l'encapsulation en utilisant les mots réservés : *public*, *private* ou *protected*.

b) Concept de polymorphisme

Le polymorphisme permet de définir plusieurs formes pour une méthode commune à une hiérarchie d'objets. Il est donc possible de déclencher des opérations différentes en réponse à un même message. C'est à l'exécution que l'on détermine quelle forme appeler suivant la classe de l'objet courant.

c) Concept d'héritage

L'héritage est une technique offerte par les langages de programmation pour construire une classe à partir d'une ou plusieurs autres classes. Il met en œuvre les principes de généralisation et de spécialisation en partageant explicitement les attributs et méthodes communs au moyen d'une hiérarchie de classes. La classe fille (sous-classe) hérite des caractéristiques (attributs et méthodes) de sa classe mère (super-classe), mais elle se distingue par ses caractéristiques propres.

d) Différents Types des Méthodes (Opérations)

d.1) Constructeurs et Destructeurs

Les constructeurs servent à construire l'objet en mémoire. Un constructeur va donc se charger de mettre en place les données, d'associer les méthodes avec les champs et de créer le *diagramme d'héritage* de l'objet. Un constructeur est une méthode spéciale qui crée et initialise des instances d'objet. La déclaration d'un constructeur ressemble à celle d'une procédure en commençant par le mot *constructor* dans Delphi.

Le destructeur est le pendant du constructeur : il se charge de détruire l'instance de l'objet. La mémoire allouée pour le *diagramme d'héritage* est libérée. Certains compilateurs peuvent également se servir des destructeurs pour éliminer de la mémoire le code correspondant aux méthodes d'un type d'objet si plus aucune instance de cet objet ne réside en mémoire. Dans Delphi, on utilise le mot *destructor*.

d.2) Méthodes Virtuelles et Dynamiques

Les méthodes dynamiques ou virtuelles, à la différence des méthodes statiques, peuvent être surchargées dans les classes dérivées. Quand une méthode surchargée est appelée, c'est le type réel (à l'exécution) du type de classe ou d'objet utilisé dans l'appel de la méthode, et non pas le type déclaré de la variable, qui détermine l'implémentation activée. Dans Delphi, on utilise les mots *virtual* ou *dynamic*.

d.3) Méthodes Abstraites

Une méthode abstraite est une méthode virtuelle ou dynamique n'ayant pas d'implémentation dans la classe où elle est déclarée. Son implémentation est déléguée à une classe dérivée. Les méthodes abstraites doivent être déclarées en spécifiant la directive *abstract* après *virtual* ou *dynamic*.

3.2.3 Avantages de la POO

Bien que la POO requière de la part du concepteur un changement d'état d'esprit par rapport à l'approche algorithmique classique, elle présente néanmoins de nombreux avantages dont on cite [2]:

- **Modularité** : Pour une conception judicieuse d'applications de grandes envergures, il est recommandé de diviser un programme en un ensemble de modules complètement indépendants faciles à maintenir. La structure modulaire

fournit donc des mécanismes pour un remplacement facile d'algorithmes spécifiques, en rendant possible de tester de nouveaux algorithmes et/ou comparer la performance de codes distincts.

- **Réutilisation des codes** : Les classes produites peuvent être regroupées en bibliothèques et être réutilisées.
- **Lisibilité** : L'interface (documentée) permet d'avoir un mode d'emploi clair et précis de l'utilisation d'une classe, qui est d'autant plus clair que l'implantation des classes est cachée.
- **Extensibilité** : Les classes sont définies en terme de services. Dès lors, un changement de représentation interne de données ou une modification de celles-ci n'altère pas la façon dont les autres classes les utilisent.

3.3 Langage unifié de modélisation (UML)

Le but d'UML est de décrire les systèmes complexes en utilisant des concepts orientés objets. UML est bien structuré pour être utilisable par des machines mais il est également très graphique, permettant d'offrir un langage commun de description et de représentation universel. Ainsi les échanges de données sont plus faciles ce qui conduit à une réutilisation de plus en plus importante [8]. Les diagrammes proposés dans UML sont classés selon deux aspects :

- L'aspect statique du système qui est décrite par le diagramme de classes (et le diagramme d'objets).
- L'aspect dynamique du système qui s'exprime via les diagrammes de collaboration (ou d'interaction), les diagrammes séquentiels, et les diagrammes d'activité.

3.3.1 Modèle statique (Diagramme de classes)

Le diagramme de classes permet de représenter la structure des classes qui forment l'architecture du système. C'est un diagramme de base à tous les autres diagrammes, il décrit le système en termes de classes et de relations entre classes. Les relations entre les classes sont exprimées graphiquement par des chemins entre les rectangles avec des textures des décorations sur les chemins et leurs extrémités. Normalement, la valeur des attributs d'un objet n'est pas accessible directement par un autre objet, les attributs sont masqués à l'intérieur d'un objet. L'interaction entre les objets s'opère en activant différentes opérations déclarées dans l'interface. Ainsi, les opérations de l'interface d'une classe sont accessibles à

d'autres classes. Plusieurs niveaux de protections sont généralement disponibles. Par exemple Delphi offre trois niveaux d'accessibilité [78]:

- **Niveau privé -** : Seuls les objets de même classe et les fonctions et les objets amis peuvent y accéder (c'est le niveau de protection le plus élevé).
- **Niveau protégé #** : Seuls les objets de même classe et les objets de classes dérivées peuvent y accéder (c'est le niveau de protection intermédiaire).
- **Niveau publique +** : L'effet de l'encapsulation est éliminé. Les opérations et les attributs placés dans ce niveau sont accessibles par tous (c'est le niveau de protection le plus faible).

a) Représentation d'Association

La relation d'association exprime une abstraction du lien entre les objets. L'association entre deux classes possède un rôle et une nature de l'association. L'association normale unidirectionnelle est une association dirigée entre classes représentée par une flèche. L'association bidirectionnelle est une association entre les deux sens de classes, est représentée par une ligne droite qui relie les deux classes associées. La Figure 3.2 illustre l'association entre les différents éléments du réseau électrique [40].

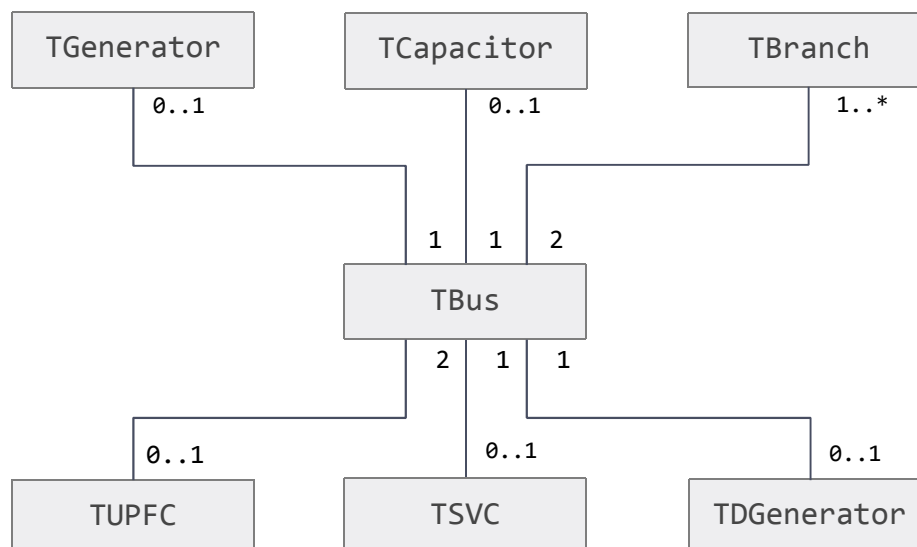


Figure 3.2 Association entre les classes du réseau électrique

La multiplicité est utilisée pour indiquer le nombre d'instances d'une classe liées à une instance d'une autre classe. Le Tableau 3.1 donne les différentes valeurs utilisées pour indiquer la multiplicité.

Tableau 3.1 Valeurs de la multiplicité

Notation	Description
1	Exactement un et un seul
0..1	Zéro ou un
x..y	Au minimum x et au maximum y
*	plusieurs
0..*	Zéro ou plus
1..*	Un ou plus
n	Nombre spécifié

b) Représentation d'Agrégation

L'agrégation est une autre manière d'association avec un lien de composition. Il existe deux types d'agrégations:

- L'agrégation forte : Ou composition, la notation UML utilise une flèche dont la pointe est un losange plein pour représenter l'agrégation forte (composition). Si la classe entière disparaît la classe partie disparaît aussi.
- L'agrégation par référence: Ou agrégation simple, l'UML utilise une flèche dont la pointe est un losange pour représenter l'agrégation par référence. Si la classe entière disparaît la classe partie n'est pas disparue. Le losange est du côté de la classe composite (classe entière).

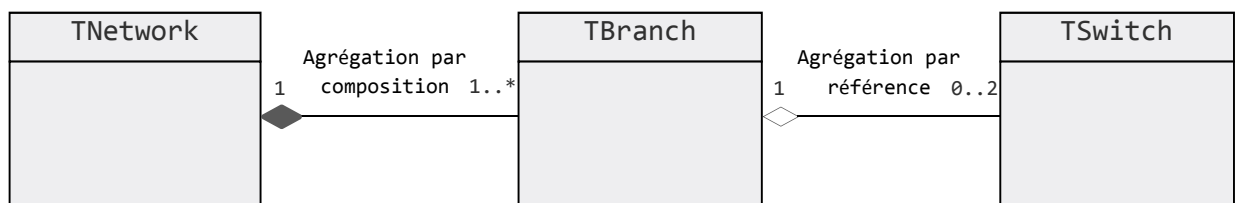


Figure 3.3 Agrégation par référence et agrégation par composition

c) Représentation d'Héritages

L'héritage est la propriété qui fait bénéficier à une sous-classe de la structure et du comportement de sa sur-classe. L'héritage provient du fait qu'une sous-classe est un sous-ensemble de sa sur-classe. Ses instances sont également instances de sa sur-classe. En conséquence, elles bénéficient de la structure et du comportement définis dans cette sur-classe,

en plus de la structure et du comportement introduits au niveau de la sousclasse. En UML, l'héritage se représente par une flèche blanche généralement tournée vers le haut. La Figure 3.4 illustre un exemple de représentation d'héritage.

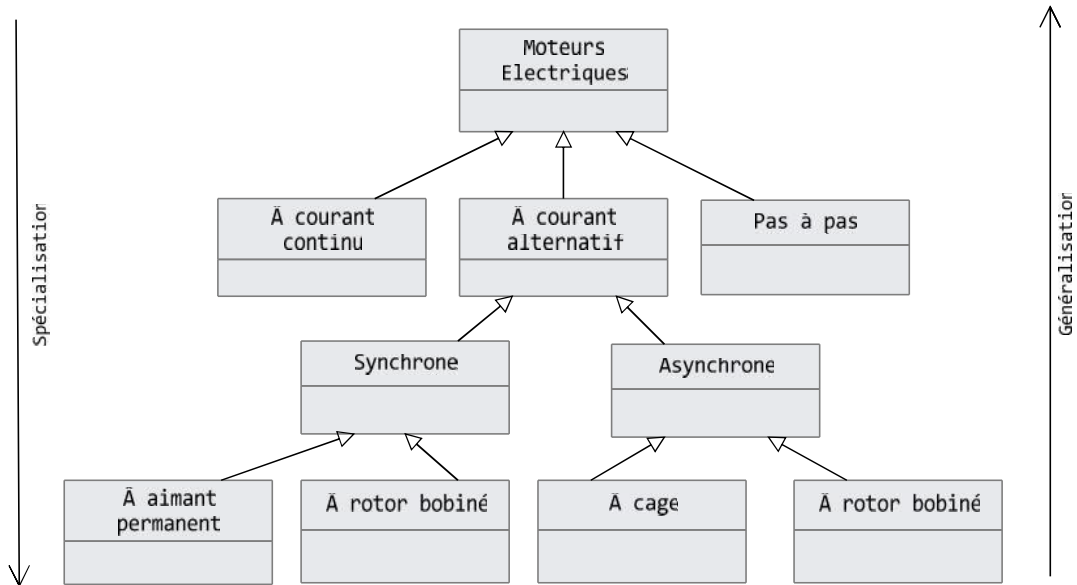


Figure 3.4 Représentation de l'héritage

d) Packages

Pour les systèmes logiciels formés d'un très grand nombre de classes, il convient de regrouper celles-ci en entités logiques distinctes. En langage UML, ces entités logiques s'appellent des *packages* ou *paquets*. Chaque package est muni d'une interface formée de ses classes publiques qui lui permet de communiquer sa fonctionnalité aux autres packages qui peuvent l'importer. Un package peut également importer des éléments d'un modèle compris dans d'autres packages. En langage UML, un package est représenté graphiquement par une *icône de dossier*. Les packages sont également dotés d'un niveau de *visibilité*, et peuvent avoir des *relations* entre eux. Ces relations sont de trois types: la *dépendance*, le *raffinement*, et la *généralisation*. Un package dépend d'un autre package s'il importe des éléments de celui-ci. Il en est un raffinement s'il décrit la même partie du modèle mais à un niveau différent d'abstraction. La généralisation prend la même signification que pour les classes. Les liens et relations entre les packages sont illustrés à la par une *flèche pointillée* (allant dans le sens de la dépendance, de la généralisation ou du raffinement). Les classes d'un package ne sont pas accessibles à l'extérieur du package inconditionnellement.

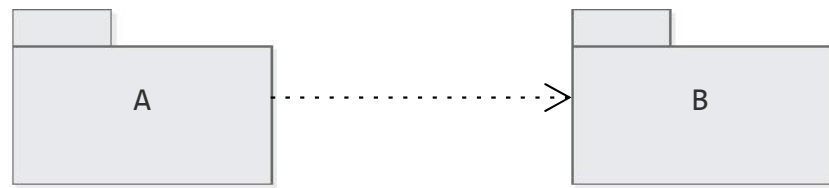


Figure 3.5 Package A importe les services du package B

3.3.2 Modèle Dynamique

Le modèle statique permet de modéliser les principaux éléments formant le système, tandis que le modèle dynamique a pour but de modéliser l'évolution de chaque élément en fonction des messages qu'il reçoit et les interactions entre ces éléments dans le temps et dans l'espace. La représentation graphique de ce modèle est décrite principalement par des diagrammes d'états, diagrammes d'activités, diagrammes séquentiels et le diagramme de collaboration.

a) Diagramme d'Etats

Le diagramme d'états montre les états que peuvent prendre les objets et comment les événements extérieurs (messages reçus, temps écoulé, etc.) peuvent provoquer un changement d'état. Le but de diagramme d'états est de fournir une description détaillée de différents états d'une classe au comportement dynamique en identifiant les transitions entre les états.

b) Diagramme d'Activités

Le diagramme d'activités sert à décrire le travail effectué dans les opérations d'une classe à un niveau plus élevé. Il se rapproche conceptuellement aux organigrammes introduits en génie logiciel.

c) Diagramme Séquentiel

Le diagramme séquentiel s'intéresse à la manière dont plusieurs objets interagissent entre eux. Il se base sur la séquence des messages envoyés entre les objets. Les diagrammes séquentiels possèdent un axe vertical qui présente le temps et un axe horizontal qui présente l'ensemble des objets en interaction dans un processus. Ces derniers permettent de visualiser le programme lors de son exécution. Quand celui-ci s'exécute en effet, ce sont

les objets qui s'agitent, en se sollicitant mutuellement par l'envoi de messages, et ce sont précisément ces envois de messages qui constituent l'essentiel de ces diagrammes.

d) Diagramme de Collaboration

Le diagramme de collaboration s'intéresse également à la modélisation dynamique de l'interaction entre les objets participant à la réalisation d'un point fonctionnel, cependant il vise principalement à représenter et à illustrer les liens entre les objets et comment ils interagissent, le temps étant ici une variable secondaire. Généralement, les diagrammes de collaboration sont plus utiles que les diagrammes séquentiels.

3.4 Modélisation orientée objet des réseaux électriques

La phase de modélisation est une étape très difficile, c'est celle qui fait le plus appel au « sens de l'ingénieur », elle suppose que l'on ait préalablement fourni une réponse claire à ce qu'on veut représenter et avec quel objectif. Dans le domaine de réseaux électriques, on constate que l'application de la MOO a beaucoup d'avantages vue que la structure physique d'un réseau électrique est adaptable à une structure de classes [2-7].

La MOO d'un réseau électrique ne vise pas seulement la construction d'une bonne structure de classes de ses éléments, mais également des méthodes d'analyse (applications ou fonctions de calcul) de ce système électrique et son GUI. Il serait bien normal de penser que les méthodes d'analyse devraient être implémentées comme méthodes (fonctions membres) dans la structure représentative du système. Cependant, cette vision limite la flexibilité de la structure. Actuellement la MOO fait sortir trois grandes structures de classes : la structure des éléments physiques du système électrique, les méthodes d'analyse appliquées au système ou sur ses éléments et une troisième structure utilitaire concerne les facilités de calcul (matrices, algorithmes, etc.).

La définition des structures de classes est le point de départ dans un projet orienté objets. Le modèle objet statique qui représente les éléments du monde réel et ses relations est la base des prochaines étapes du projet [2]. Par conséquent, une attention particulière doit être adressée à cette phase du projet qui n'est pas une tâche facile. En effet, dans des problèmes complexes, le cas des réseaux électriques, la possibilité d'avoir plusieurs représentations de classes est largement évidente. Donc le choix d'une forme ou d'une autre pour la représentation de la structure, selon la philosophie de la MOO, peut impliquer des dissimilitudes dans l'exécution des applications bien déterminées.

D'une manière générale, la conception d'un bon logiciel orienté objets n'est pas une tâche facile. Un bon logiciel doit également avoir de bonnes performances sur le plan temps

de calcul. Les travaux les plus récents [79-82] ont donné des résultats satisfaisant pour des réalisations orientées objets (en utilisant le langage de programmation C++ et Delphi). Ces améliorations dans le temps de calcul sont dues aux développements de compilateurs qui supportent l'orienté objets et qui ont évolué suffisamment ces dernières années. D'un autre coté une grande évolution sur le plan matériel a permis de disposer de machines de grande capacité de traitement de données.

3.4.1 Structure objet du réseau électrique

a) Classe de base: TPowerComponents

Le modèle du réseau électrique est structuré de la même façon que le réseau électrique réel. Les éléments électriques de puissance qui composent le réseau électrique à savoir les générateurs, les transformateurs, les lignes de transmission, les charges et les batteries de condensateur vont être modélisés sous forme d'objets. Les objets similaires sont regroupés dans des classes qui héritent les fonctionnalités et les données des superclasses. Ils sont représentés par une classe de base et un ensemble de sous-classes héritées d'elle. La Figure 3.6 illustre la mise en œuvre de la classe «TPowerComponents» qui a été définie comme la classe de base pour toutes les classes du réseau électrique. La classe «TPowerComponents» a une partie privée qui englobe les attributs utilisées pour décrire le fonctionnement électrique de l'élément et une partie publique qui contient les méthodes d'accès aux données ; ces méthodes font naturellement la coordination avec les autres classes et communiquent avec l'environnement externe de la classe. Un constructeur est une fonction qui initialise toutes les variables de la classe et un destructeur pour libérer la mémoire allouée pour cette classe.

b) Classe: TBus

La classe TBus modélise les jeux de barres d'un réseau électrique. Ces éléments servent de points de connexion pour la formation des éléments de base du réseau électrique. Cette classe joue un rôle primordial dans la conception de la hiérarchie de classes proposée. Figure 3.7 montre les détails de cette classe.

```

type
TPowerComponents = class (TObject)                               {class name}
private                                                         {private part of the class}
  ID      : integer;
  x, y, Rotation: integer;
  FOnDbClick : TNotifyEvent;
  ListOfComponents : TList;
  .....
public                                                         {public part of the class}
  ComponentName:string;
  property Name : String read  ComponentName;
  property ComponentID : integer read  ID;
  property OnDbClick : TNotifyEvent
            read FOnDbClick write FOnDbClick;
  constructor Create(AOwner: TComponent); override; virtual;
  destructor Destroy; override;                               {object destructor}
  .....
end;

```

Figure 3.6 Une partie de la class TPowerComponents

Outre les attributs et méthodes hérités de la classe de base, les principaux attributs et méthodes de *Tbus* sont:

- *Name*: Nom du jeu de barres
- *BusType* : indique le type de jeu de barres (générateur, de charge, de référence)
- *Area*: Numéro de la zone
- *Voltage (p.u), Angle (°)* : Module et angle de tension
- *RealLoad (MW), ReactiveLoad (MVar)*: Puissance active demandée
- *Limits* : Une matrice qui contient toutes les limites physiques du jeu de barres
- *GetLimits()*: Recevoir la liste des limites du jeu de barres à partir de GUI
- *ConnectedComponents*: Liste des composants liés au jeu de barres
- *GetVoltage(),GetAngle()*:Recevoir le module et l'angle de tension à partir de GUI ou après le calcul de l'écoulement de puissance (suivant le type du jeu de barres).
- *GetConnected ()*: Recevoir la liste des composants liés au jeu de barres à partir de GUI
- *Create, Destroy*: Constructeur et destructeur de l'objet

c) Classe: TBranch

La classe *TBranch* représente la ligne de transmission du réseau électrique, la ligne de transmission doit être connectée entre deux jeux de barres. Le modèle algébrique considéré est le circuit en . La Figure 3.8 montre les détails de cette classe.

Les principaux attributs et méthodes de cette classe sont :

```

type
  TBus = class (TPowerComponents)                                {class name}
  FbusNo:integer;
  FBustype:integer=0;
  FArea:integer=1;
  FVoltage:double=1;
  FAngle:double=0;
  FRealLoad:double=0;
  FReactiveLoad:double=0;
  FLimits : array of array of double;
  ....
  private                                                         {private part of the class}
    ConnectedComponents: TList;
    function GetConnectedComponents(): TList;
    function GetLimits(): Tmatrix;
  ....
  public                                                           {public part of the class}
    ComponentName:string;
    property Name : String read ComponentName;
    property busNo : integer read FbusNo;
    property Bustype : integer read FBustype;
    property area : integer read Farea;
    property realLoad : double read FrealLoad;
    property reactiveLoad : double read FreactiveLoad;
    property Limits[Index: Integer]: double read GetLimits;
    procedure GetVoltage();
    procedure GetAngle();
    constructor Create(AOwner: TComponent); override;
    destructor Destroy; override;                               {object destructor}
  ....
end;

```

Figure 3.7 Une partie de la class TBus

- *Name*: Nom de la branche
- *FromBus, ToBus*: Les numéros des nœuds de départ et d'arrivées de la branche.
- *YBranch*: Admittance de la ligne
- *Ratio*: Rapport de transformation de la branche
- *TieBranch*: Indique si la branche est normalement ouverte
- *ActiveBranch*: Indique si la branche est en service
- *GetFromBus (), GetToBus()*: Recevoir les numéros des nœuds de départ et du jeu de barres à partir de GUI
- *GetPowerFlow()*: Détermine les puissances écoulées dans la branche après l'exécution de l'écoulement de puissance
- *GetPowerLoss ()*: Détermine les pertes après l'exécution de l'écoulement de puissance
- *Create, Destroy*: Constructeur et destructeur de l'objet

```

type
  TBranch = class (TPowerComponents)                               {class name}
    FFromBus:integer;
    FToBus:integer;
    FYBranch:TComplex=0;
    FRatio:double=1;
    FLimits: double=inf;
    FPowerLoss:TComplex=0;
    FPowerFlow:TComplex=0;
    FTieBranch: boolean=false;
    FActiveBranch: boolean=true;
    ....
  private                                                         {private part of the class}
    function GetFromBus(): TBus;
    function GetToBus(): TBus;
    ....
  public                                                           {public part of the class}
    property FromBus : integer read FFromBus;
    property ToBus : integer read FToBus;
    property Ratio : double read FRatio;
    property Limits : double read FLimits;
    function GetPowerLoss(): TComplex;
    function GetPowerFlow(): TComplex;
    procedure IsActive();
    constructor Create(AOwner: TComponent); override;
    destructor Destroy; override;                               {object destructor}
    ....
end;

```

Figure 3.8 Une partie de la class TBranch

d) Classe: TShuntElement

La classe TShuntElement est une classe abstraite, qui sert de base pour tous les éléments en dérivation dans le réseau électrique tels que le générateur, la charge et le compensateur (éléments connectés à un jeu de barres seulement). Ce sont les éléments considérés dans ce travail. Cette classe est illustrée par la Figure 3.9.

Outre les attributs et méthodes hérités de la classe de base, les principaux attributs et méthodes de *TShuntElement* sont:

- *ConnectedBus*: Numéro du jeu de barres de connexion
- *GetConnectedBus()*: Détermine le jeu de barres de connexion à partir du GUI (pointeur vers l'objet jeu de barres de connexion)
- *Create, Destroy*: Constructeur et destructeur de l'objet

```

type
  TShuntElement = class (TPowerComponent)           {class name}
    FConnectedBus:integer;
    FActiveElement: boolean=true;
    FVoltage: double=1.0000;
    ....
  private                                           {private part of the class}
    function GetConnectedBus(): TBus;
    ....
  public                                           {public part of the class}
    procedure IsActive();
    constructor Create(AOwner: TComponent); override;
    destructor Destroy; override;                 {object destructor}
    ....
end;

```

Figure 3.9 Une partie de la class TShuntElement

e) Classe: TCompensateur

Le modèle algébrique du compensateur shunt est une impédance en série avec une capacité. Cette classe représente les compensateurs d'énergie réactive pouvant être reliés à un jeu de barres dans un réseau électrique. Il est produit par des batteries de condensateurs et/ou des réacteurs. Cette classe est illustrée par la Figure 3.10.

L'attribut principal de la classe Compensator est sa puissance réactive, représentée par Q_{sh} , sa valeur sera positive si le compensateur est une batterie de condensateurs, négative si c'est un réacteur. Cette puissance réactive sera injectée au jeu de barres auquel le compensateur est relié.

- Q_{sh} : puissance réactive ;
- $GetQ_{sh}()$: méthodes d'accès au attribut Q_{sh} ;
- $Create, Destroy$: Constructeur et destructeur de l'objet

```

type
  TCompensateur = class (TShuntElement)           {class name}
    FQsh:double =0.00;
    ....
  private                                           {private part of the class}
    ....
  public                                           {public part of the class}
    property Qsh : double read FQsh;
    procedure GetQsh ();
    constructor Create(AOwner: TComponent); override;
    destructor Destroy; override;                 {object destructor}
    ....
end;

```

Figure 3.10 Une partie de la class TCompensateur

f) Classe: TGenerator

Le générateur est un élément typiquement câblé en dérivation dans un réseau électrique, il est responsable de la génération de la puissance au jeu de barres auquel il est connecté. La classe *TGenerator* est illustrée par la Figure 3.11.

Les principaux attributs et méthodes de *TGenerator* sont :

- *RealPower, ReactivePower*: puissances active et réactive du générateur
- *Slack*: indique si le générateur est un générateur balancier
- *Committed* : Puissances active et réactive minimales et maximales
- *mode*: réactances transitoires et subtransitoires directes et en quadrature
- *UnitConstraints* : fonctions d'accès aux attributs Pg, Qg et V en lecture
- *CostCoefficients*: fonctions d'accès aux attributs Pg, Qg et V en lecture
- *GetUnitConstraints()* , *GetCostCoefficients()*: Recevoir les coefficients à partir de GUI
- *GetStatus()*: Indique si le générateur est en service
- *GetPower()*, *GetCost()*: Calcul de la puissance délivrée et du coût de production
- *Create, Destroy*: Constructeur et destructeur de l'objet

```

type
TGenerator = class (TShuntElement)                                {class name}
  FRealPower:double=0;
  FReactivePower:double=0;
  FSlack:boolean= false;
  FCommitted:boolean= true;
  Fmode:integer=1;
  FPowerLimits : array of array of double;
  FUnitConstraints : array of array of double;
  FCostCoefficients : array of array of double;
  ....
private                                                            {private part of the class}
  function GetUnitConstraints(): Tmatrix;
  function GetCostCoefficients (): Tmatrix;
  ....
public                                                            {public part of the class}
  property FRealPower: double read FRealPower;
  property Slack: boolean read FSlack;
  procedure GetStatus ();
  function GetPower (): Tcomplex;
  function GetCost (): double;
  constructor Create(AOwner: TComponent); override;
  destructor Destroy; override;                                {object destructor}
  ....
end;

```

Figure 3.11 Une partie de la class TGenerator

La figure 3.12 montre la conception globale des classes de base du réseau électrique et leurs sous-classes. Quatre classes de base sont identifiées et dérivées de la classe de base *TPowerComponents*. La classe «TBus» (jeu de barres) représente une classe essentielle parce

que tous les éléments électriques de puissance sont connectés à un ou deux jeux de barres. La classe «TBranch» représente les éléments électriques qui sont liés à deux jeux de barres alors que ceux qui sont liés à un seul jeu de barres sont groupés dans la classe «TShuntElement». En utilisant la notion d'héritage, de nouvelles classes qui représentent le reste des éléments du réseau électrique sont dérivées de ces classes. Par exemple, la classe TBranch est la classe de base des lignes électriques «TBranch» et des transformateurs «Transformer». Les attributs et les opérations associées à la classe «TBranch» tel que le jeu de barres de départ, le jeu de barres d'arrivée, la résistance, la réactance et la susceptance sont hérités par les sous-classes de la ligne et du transformateur. Aussi, la classe « Line » elle-même est une classe de base pour les lignes aériennes «OverHeadLine» et les câbles souterrains «Cable».

3.4.2 Modélisation orientée objet des applications (UCP comme exemple)

La classe *TUnitCommitment* est une classe de base qui contient les données et les méthodes relative à la résolution du problème de l'Unit Commitment (UCP) telle que la création de la solution initiale, le codage binaire de la solution et le calcul de la fonction objectif. Elle contient également des fonctions de préparation des données, qui fournit l'interface entre le solveur de l'UCP et l'utilisateur. En héritant de la classe de base *TUnitCommitment*, on peut modifier ou ajouter des techniques de résolution tel que la méthode hybride Recherche Taboue/ Recuit Simulé (RT/RS) sans avoir à réécrire les méthodes communes qui ont été définies dans la classe de base.

La classe *TOptimalPowerFlow* est également conçu pour être la classe de base pour tous les algorithmes d'écoulement de puissance optimale. La classe *TNewtonOPF* est dérivée de la classe de base *TOptimalPowerFlow* pour résoudre le problème d'écoulement de puissance optimale à l'aide de la méthode de Newton. Le solveur linéaire [83] est adopté dans la classe de base de l'écoulement de puissance optimale pour résoudre l'équation linéaire creux de forme $[A][X]=[b]$. La fig. 3.12 montre la représentation UML du solveur de l'UCP.

Tous les objets connectés dans l'interface utilisateur graphique (GUI) seront répertoriés dans trois catégories : *listOfBranches*, *listOfBuses*, *listOfGenerators* et *listOfCapacitors* de type TList, ces listes seront mises à jour après toute modification au cours de la phase de conception du réseau électrique dans le GUI.

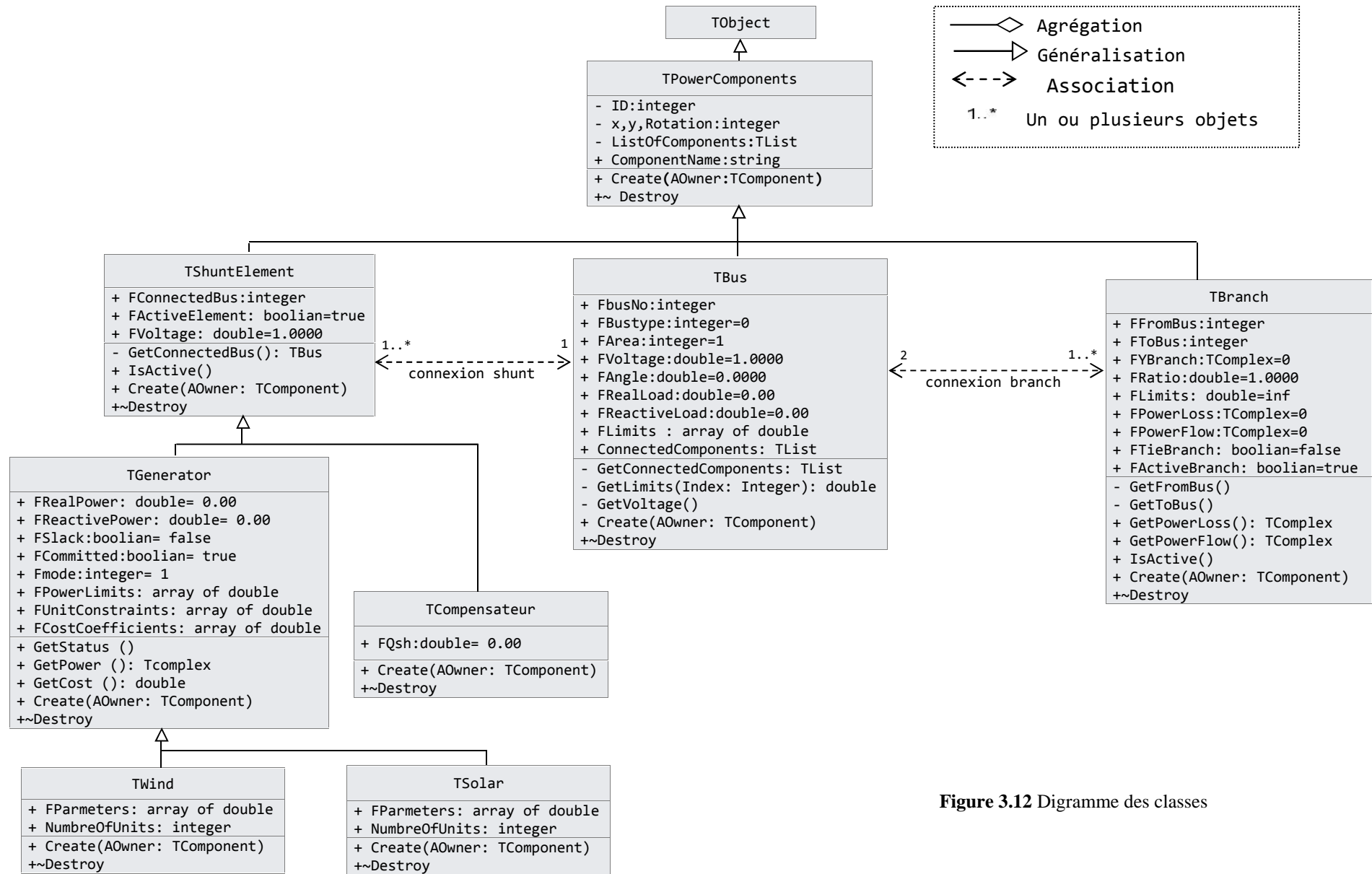


Figure 3.12 Digramme des classes

La classe *TUnitCommitment* invoque la fonction *getListOfObjects()* pour obtenir la liste des générateurs à partir de la classe GUI. La même fonction *getListOfObjects()* est redéfini (overridden) dans la classe *TOptimalPowerFlow* pour obtenir les quatre listes *listOfBranches*, *listOfGenerators*, *listOfCapacitors*, et *listOfBuses* à partir de la classe GUI. Toutes les données de ces objets seront utilisées pour calculer la matrice admittance, la matrices Hessienne et Jacobienne alors que la classe de Unit commitment utilise seulement les données des générateurs.

Si l'utilisateur décide d'exécuter l'application « Unit Commitment » sans tenir compte des contraintes d'écoulements de puissance optimale, la classe *TLambdaEDP* sera utilisé pour calculer le Dispatching économique à l'aide de la méthode Lambda [11] à chaque heure. Dans ce cas, les pertes de puissance et les contraintes du réseau ne seront pas considérées.

La description des fonctions membres des classes sont résumées dans le Tableau 3.2. La Figure 3.12 montre le diagramme d'activités pour la résolution du problème de l'Unit Commitment par la méthode RT/RS.

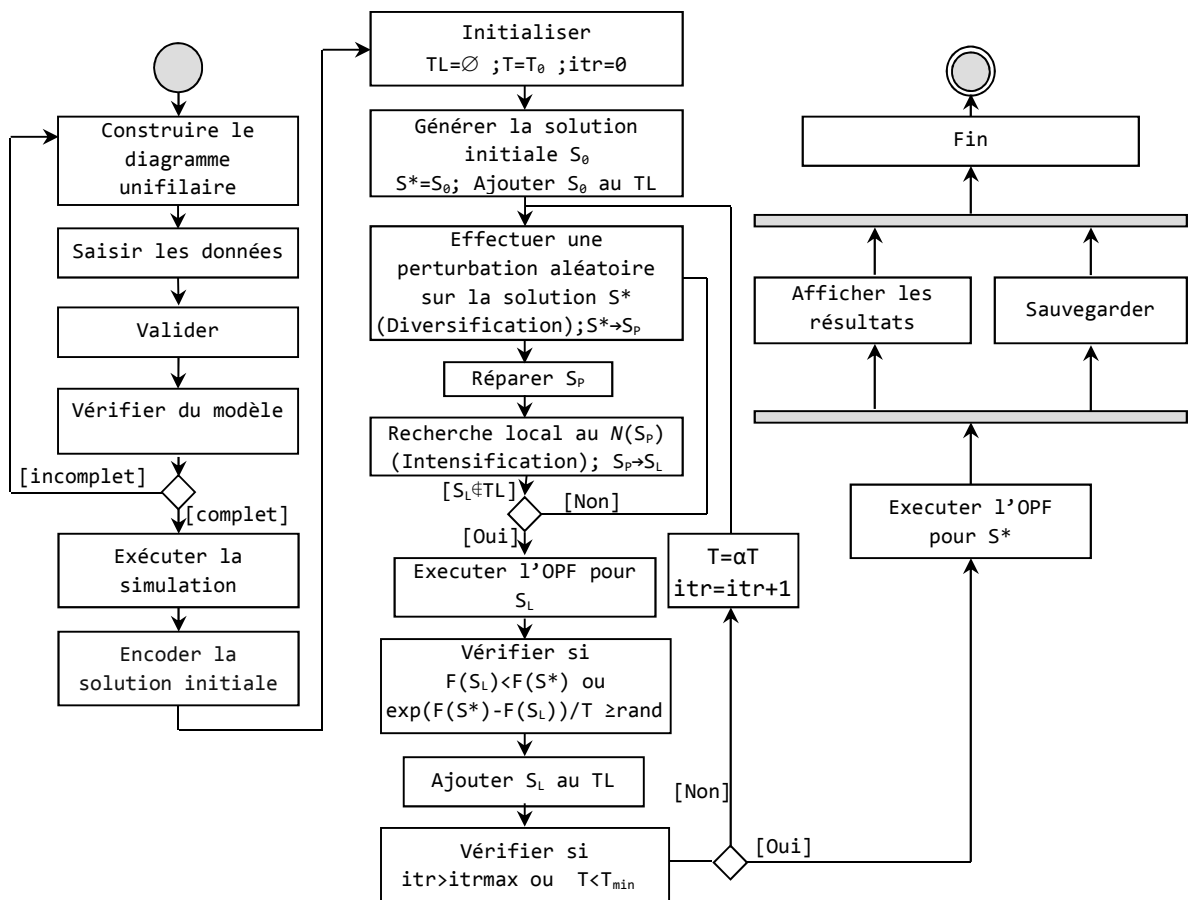
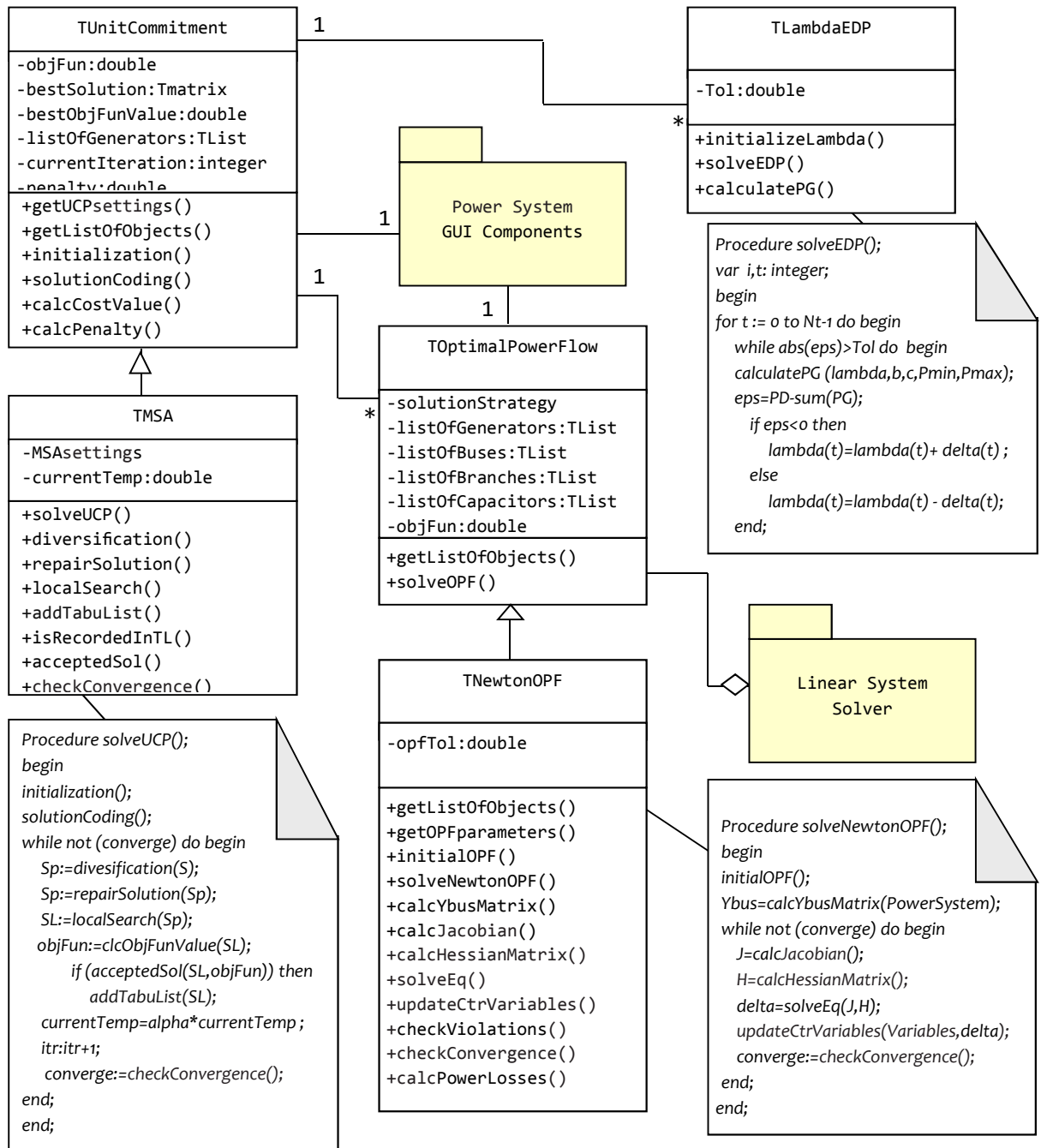


Figure 3.12 Diagramme d'activités pour la résolution de l'UCP



Modèle objet proposé pour l'application UCP

Tableau 3.2 Principales fonctions utilisées dans l'application UCP

Classe	Opération	Description
TUnitCommitment	getUCPsettings()	Recevoir les paramètres d'optimisation de l'UCP à partir de GUI
	getListOfObjects()	Recevoir la liste des composants du réseau électrique à partir de GUI

	<i>initialization()</i>	<i>Calculer la solution initiale en utilisant la méthode de la Liste de Priorité</i>
	<i>solutionCoding()</i>	<i>Encoder en binaire la solution initiale</i>
	<i>calcCostValue()</i>	<i>Calculer le coût total de production</i>
	<i>calcPenalty()</i>	<i>Calculer les pénalités dans le cas des violations des contraintes d'inégalité.</i>
	<i>clcObjFunValue()</i>	<i>Calculer la valeur de la fonction objectif</i>
<i>TMSA</i>	<i>solveUCP()</i>	<i>Exécuter la méthode RT/RS pour la résolution du problème de l'UCP</i>
	<i>diversification()</i>	<i>Effectuer une perturbation aléatoire sur la solution actuelle</i>
	<i>repairSolution()</i>	<i>Réparer la solution infaisable</i>
	<i>localSearch()</i>	<i>Effectue une recherche locale dans le voisinage de la solution actuelle</i>
	<i>addTabuList()</i>	<i>Ajouter la solution actuelle à la liste taboue</i>
	<i>isRecordedInTL()</i>	<i>Vérifier que la solution actuelle n'est pas enregistrée dans la liste taboue</i>
	<i>acceptedSol()</i>	<i>Vérifier que la solution actuelle est acceptée</i>
	<i>checkConvergence()</i>	<i>Vérifier le critère de convergence</i>
<i>TOptimalPowerFlow</i>	<i>solveOPF()</i>	<i>Exécuter le calcul de l'écoulement de puissance optimal (OPF)</i>
<i>TNewtonOPF</i>	<i>getOPFsettings()</i>	<i>Recevoir les paramètres de l'OPF à partir de GUI</i>
	<i>initialOPF()</i>	<i>Initialiser l'OPF</i>
	<i>solveNewtonOPF()</i>	<i>Exécuter la méthode Newton pour la résolution du problème de OPF</i>
	<i>calcYbusMatrix()</i>	<i>Calculer les éléments de la matrice admittance</i>
	<i>calcJacobian()</i>	<i>Calculer les éléments de la matrice Jacobienne</i>
	<i>calcHessianMatrix()</i>	<i>Calculer les éléments de la matrice Hessienne</i>
	<i>solveEq()</i>	<i>Résoudre le système d'équations en utilisant le solveur linéaire</i>
	<i>updateCtrVariables()</i>	<i>Mise à jour des variables de contrôle</i>
	<i>checkViolations()</i>	<i>Vérifier les violations sur les contraintes d'inégalité</i>
	<i>calcPowerLosses()</i>	<i>Calculer les pertes dans le réseau électrique</i>
<i>TLambdaEDP</i>	<i>initializeLambda()</i>	<i>Initialiser la valeur de Lambda</i>
	<i>solveEDP()</i>	<i>Exécuter le calcul de dispatching économique par la méthode de Lambda</i>
	<i>calculatePG()</i>	<i>Calculer les puissances délivrées par les générateurs</i>

3.5 Module de l'interface utilisateur graphique (GUI)

La principale interaction entre l'utilisateur et la simulation est accomplie à travers une interface utilisateur graphique (GUI) conçue en employant la programmation orientée objet. L'avantage principal de cette technique est la gestion cohérente et normalisée, grâce aux symboles, aux menus et aux zones de dialogue graphiques que l'on active généralement par un simple clic de la souris. Pour exploiter les programmes d'analyse des réseaux électriques, on utilise des fenêtres comme environnement de travail. Les phénomènes physiques ainsi que

les données d'un réseau électrique sont mieux assimilées si l'information est représentée sous forme graphique contrairement à la forme numérique.

Pour que l'outil développé soit une plateforme pour l'étude des réseaux électriques, il est important que la GUI puisse fournir à l'utilisateur des accès faciles à toutes les informations concernant le réseau électrique.

La figure 2.1 montre la communication entre les principales parties qui sont :

- Un éditeur graphique est spécialement développé pour visualiser les diagrammes unifilaires des réseaux électriques avec fenêtres de boites de dialogue. Il utilise des symboles graphiques pour représenter les éléments du réseau électrique tels que les jeux de barres, les lignes de transmission, les charges, les générateurs, etc.
- Une base de données visuelle est développée pour que l'utilisateur puisse faire entrer et modifier les données avec souplesse sur écran. Les données sont liées au diagramme unifilaire et aux applications à exécuter.
- Les applications qui simulent le fonctionnement d'un réseau électrique.

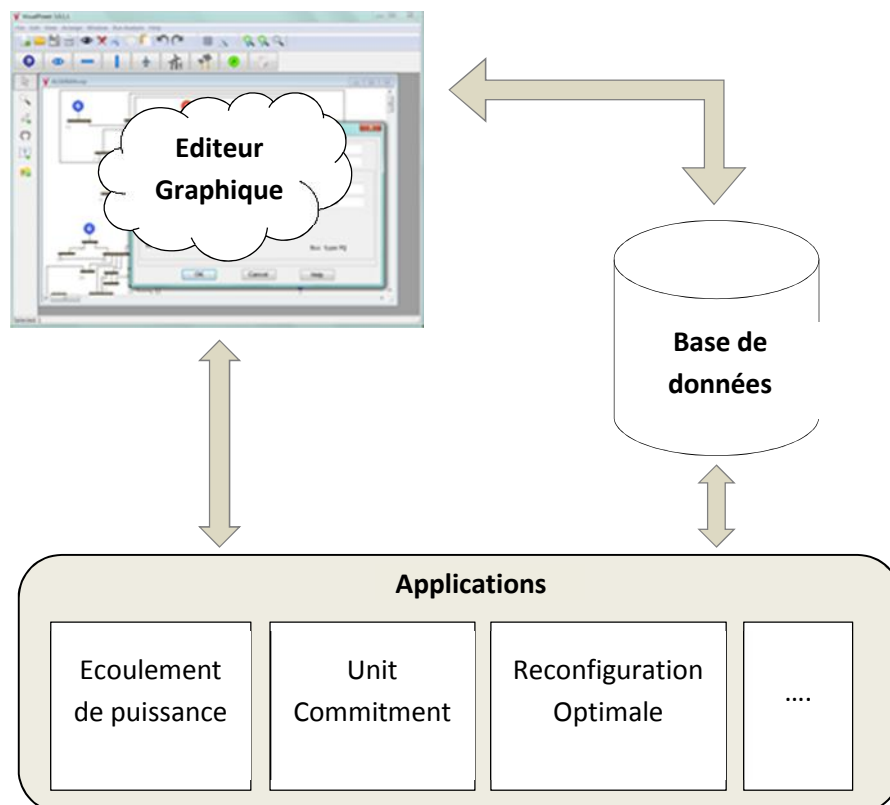


Figure 3.13 Structure générale

3.6 Conclusion

Dans ce chapitre, on a exposé le principe d'application de la technique orientée objet sur les systèmes de puissance électrique de point de vue conception et programmation. Les objets peuvent nous aider à représenter le réseau électrique dans l'environnement informatique, sans nécessité de le déformer ou le décomposer. Le réseau électrique est modélisé en utilisant deux principaux concepts de l'objet: la classification et la spécialisation. L'interaction entre l'utilisateur et la simulation est réalisée à travers un GUI sous un environnement Windows et les données numériques sont stockées et traitées dans une base de données orientée objets sous Windows. Dans le prochain chapitre, on présentera une description générale du logiciel développé.

Chapitre 4

Description du Logiciel Développé

Chapitre 4: Description du Logiciel Développé

4.1 Introduction

Dans ce chapitre, nous présentons une description concise, du logiciel *VisualPower* développé pour visualiser, analyser et optimiser les réseaux électrique, les phases d'analyse et de conception avec UML ont été déjà présentées dans le chapitre précédent. La présentation du logiciel sous environnement Windows permet l'interprétation rapide des résultats de simulations avec une communication interactive entre les utilisateurs et les processus de simulation. En effet, la représentation graphique des données et des résultats rend l'interprétation physique des phénomènes très complète et plus compréhensible.

4.2 Présentation du logiciel *VisualPower*

VisualPower est notre programme 32-bit interactif pour simuler, analyser et optimiser les performances des réseaux électriques en utilisant la technique orientée objet. Il a été développé sous l'environnement de programmation DELPHI. Ce programme peut résoudre plusieurs problèmes dans le système de puissance. Il est développé pour les systèmes d'exploitation Microsoft® Windows 7 (figure 4.1).

Sa puissante interface graphique hautement interactive ainsi que son environnement intuitif rendent l'utilisation du logiciel très conviviale et efficace. On peut construire, dans son canevas, directement le diagramme unifilaire du réseau électrique à tester et inclure interactivement les données des différents éléments constituant le réseau électrique à travers des boîtes de dialogues liées aux bases de données. Les Applications font entrer les données dans un format graphique, trouvent les solutions, et affichent les résultats sur le même graphique.

VisualPower est structuré sous forme modulaire afin de permettre l'intégration future d'autres programmes d'analyse et de calcul des réseaux électriques. La version actuelle comporte six applications informatiques implémentées comme modules indépendants qui sont :

- ✓ Ecoulement de puissance (EP)
- ✓ Ecoulement de puissance optimale (EPO)
- ✓ Dispatching économique (DE)
- ✓ Unit Commitment (UC)
- ✓ Optimisation de la Taille et de l'Emplacement des DGs (OTEDG)
- ✓ Reconfiguration Optimale des Réseaux de Distribution (RORD)

4.3 Représentation graphique des réseaux électriques

Les réseaux sont présentés sous forme de schémas unifilaires. L'utilisateur peut choisir les données et les résultats à afficher sur ces schémas. On peut graphiquement ajouter un élément, l'effacer, le délocaliser, changer son orientation ou introduire ses propriétés. Comme on peut connecter les éléments nécessaires (générateurs, transformateurs, charges, lignes électriques, éléments shunts, jeux de barres, et autres) pour représenter correctement un réseau électrique.

Une fois cette étape est terminée, il est utile de sauvegarder les configurations du réseau électrique. Pour enregistrer un réseau sur lequel on travaille, qu'il soit nouveau ou déjà existant, on clique sur le bouton Enregistrer dans la barre d'outils Fichiers, le projet s'enregistre dans le disque dur sous forme d'un fichier qui porte l'extension (*.VSP), et cela pour permettre une utilisation ultérieure de ce projet (Figure 4.2).

L'utilisation du logiciel comporte les étapes principales suivantes :

- ✓ Conception graphique du schéma unifilaire du réseau à étudier par la création des objets physiques du réseau ;
- ✓ Spécifier les données des objets ;
- ✓ Choisir une application à exécuter et spécifier ses propres données ;
- ✓ Affichage des résultats dans la fenêtre d'exécution de l'application choisie.

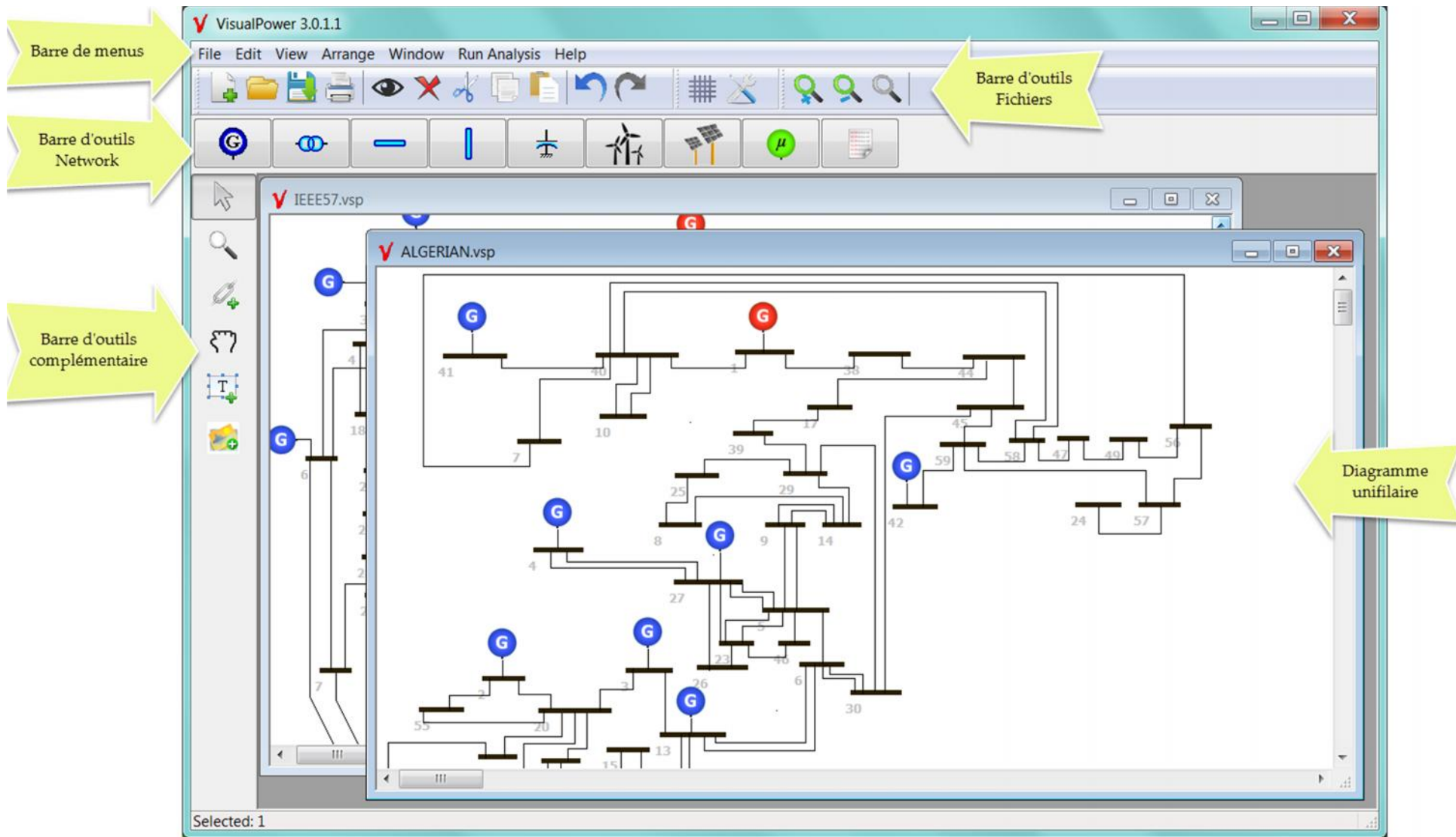


Figure 4.1 Vue générale de l'éditeur graphique

La barre d'outils Fichiers (Figure 4.3) regroupe les raccourcis relatifs à un projet à savoir : Créer un nouveau projet, Ouvrir un projet existant, Enregistrer un projet qui a déjà un nom, Nommer ou renommer un projet, Imprimer les données du réseau actuel, Imprimer le schéma unifilaire du réseau, l'aperçu du projet et le zoom. Elle contient aussi des boutons de raccourcis pour couper, copier et coller un élément graphique sur le canevas.

La palette des composants nécessaires pour la construction du schéma unifilaire du réseau existe dans les barres d'outils « Network » du logiciel de la Figure 4.4, cette palette contient les 8 composants les plus importants avec toutes les orientations possibles sous forme de sous palettes afin de bien tracer le schéma unifilaire.

Pour placer un composant sur le canevas du logiciel, il suffit de le sélectionner dans la palette puis cliquer sur le canevas pour le déposer. Dessiner un élément du réseau c'est suffisant à le créer. En cliquant sur un élément du réseau, une boîte de dialogue est ouverte, permettant de modifier les données et présentant les résultats. Les données sont définies en unités physiques (MW, kV, ...); ou en " per unit". Après l'enregistrement du projet, l'utilisateur peut exécuter application parmi les applications disponible sur le menu « Run Analysis » dans la barre de menus (Figure 4.5).

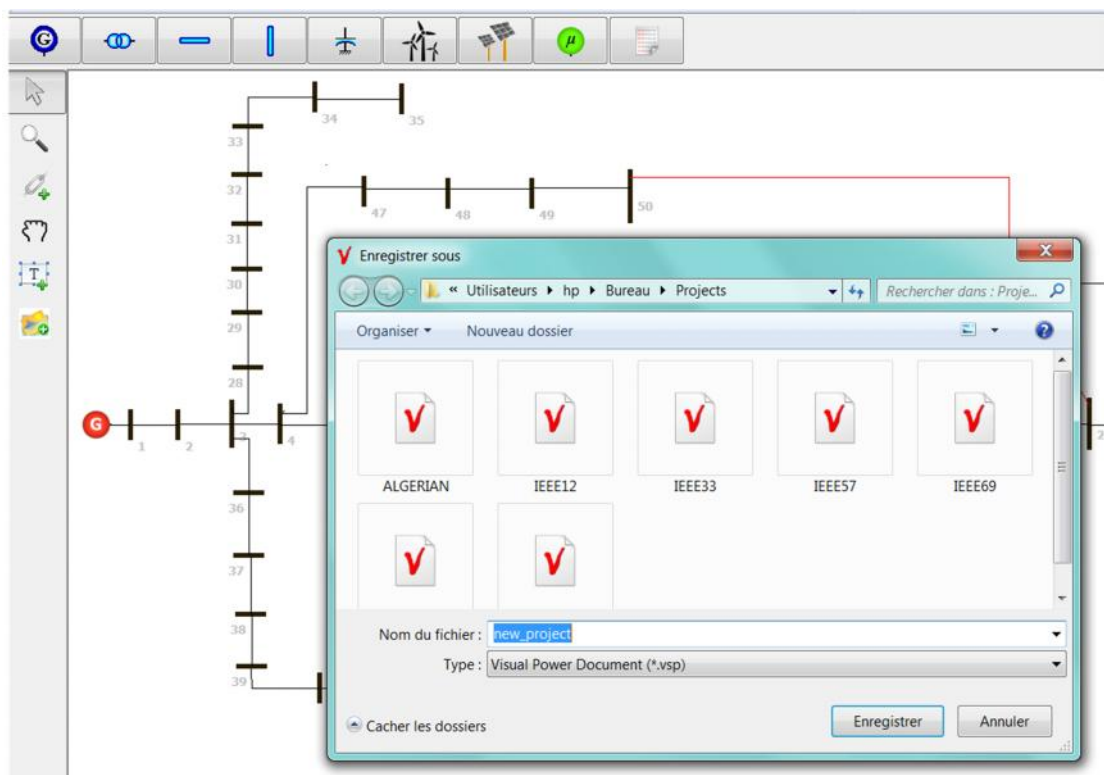


Figure 4.2 Enregistrement du projet actuel

Les paragraphes suivants décrivent les constitutions des bases de données (détails) des différents composants du réseau à l'étude et les applications disponibles dans la version actuel

du logiciel (On se limite dans cette thèse de présenter seulement les applications discuté dans les chapitres précédents).

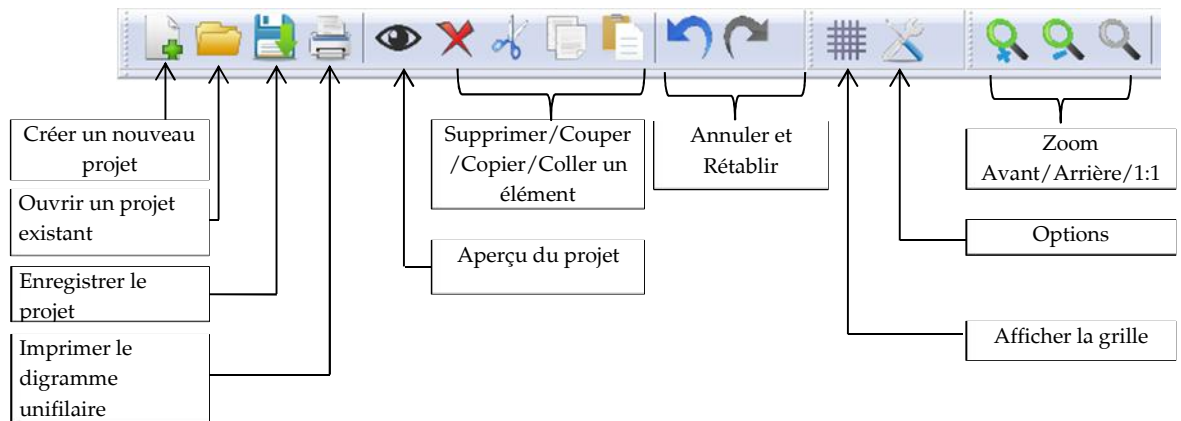


Figure 4.3 Barre d'outils Fichiers

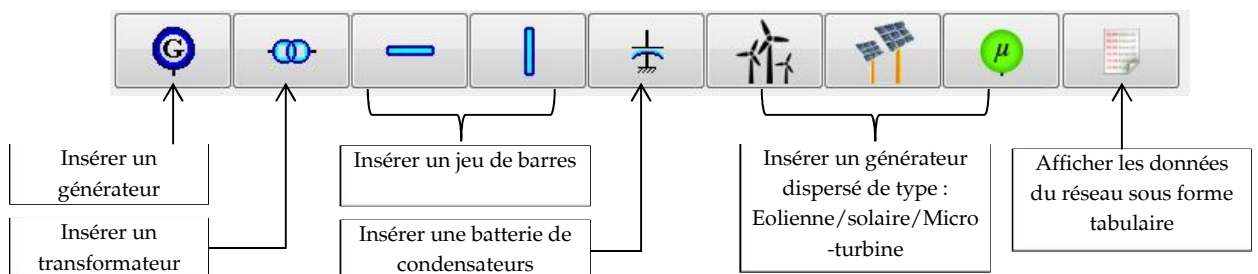


Figure 4.4 Palette des composants électriques

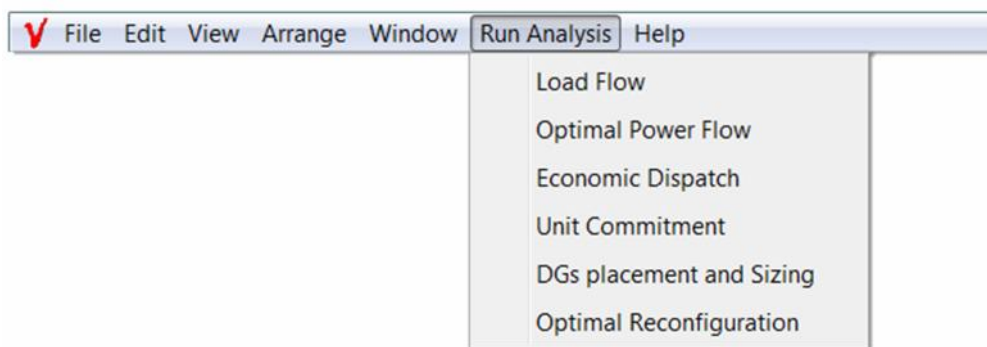


Figure 4.5 Les applications proposées dans la version actuelle

4.3.1 Données du jeu de barres

Un jeu de barres constitue un nœud de connexion des équipements électriques dans un réseau électrique. Tous les composants électriques sont connectés à un ou à deux jeux de barres. Le nom « *TBus* » est l'identificateur choisi tout au long de la conception pour la classe des jeux de barres. Quand l'utilisateur double-clique sur le symbole de *TBus* au moment de sa

création, la boîte de dialogue est ouverte et l'utilisateur peut spécifier les paramètres de cet élément (Figure 4.6). Le Tableau 4.1 résume les paramètres du modèle d'un jeu de barres.

Figure 4.6 La boîte de dialogue de Bus

Tableau 4.1 Données du jeu de barres

<i>Champ</i>	<i>Description</i>	<i>Type</i>	<i>Unité</i>
<i>Name</i>	Nom du jeu de barres	String	-
<i>Area</i>	Numéro de la zone	Integer	-
<i>baseKV</i>	Tension de base	Integer	kV
<i>Pd (MW)</i>	Puissance active demandée	Float	MW
<i>Qd (MVar)</i>	Puissance réactive demandée	Float	p.u
<i>Gs (p.u)</i>	Conductance totale	Float	p.u
<i>Bs (p.u)</i>	Susceptance totale	Float	p.u
<i>MaxVm (p.u)</i>	Tension maximale admissible	Float	p.u
<i>MinVm (p.u)</i>	Tension minimale admissible	Float	p.u

4.3.2 Données de la ligne de transport

L'objet « ligne électrique », qui est identifié par le nom « *TBranch* », doit être connecté entre deux jeux de barres (Les jeux de barres doivent être créés en premier lieu). Le modèle en π est considéré dans ce logiciel. Le Tableau 4.2 montre l'ensemble des variables liées au

modèle de la ligne électrique en indiquant seulement les paramètres spécifiques de la classe *TBranch*. Après avoir cliqué sur le symbole de *TBranch*, l'utilisateur doit positionner le curseur sur le premier jeu de barres et glisser la souris pour atteindre le deuxième jeu de barres et relâcher la souris. La Figure 4.7 montre la boîte de dialogue associée à cet élément.

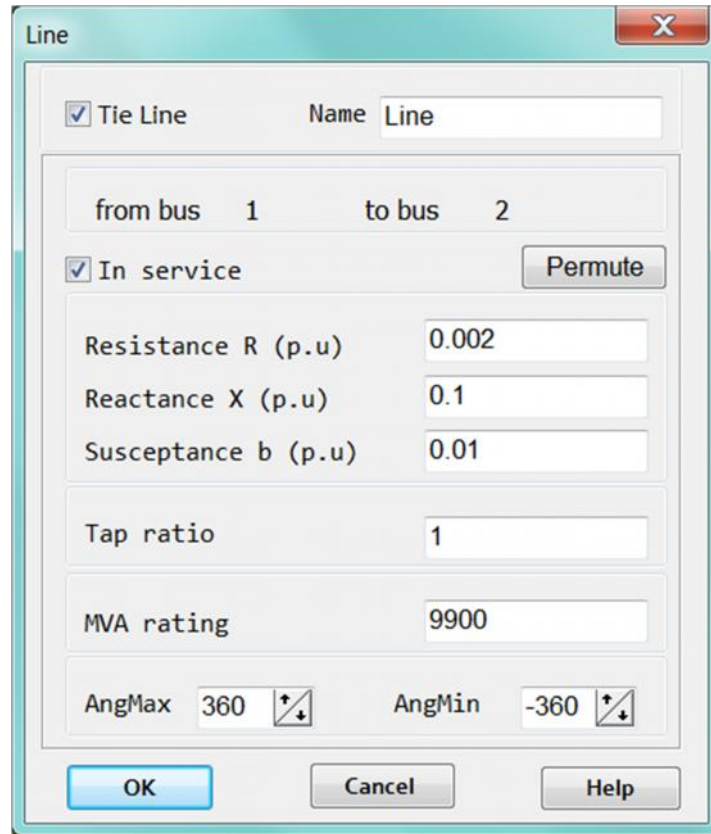


Figure 4.7 La boîte de dialogue de ligne de transport

Tableau 4.2 Données de ligne de transport

<i>Champ</i>	<i>Description</i>	<i>Type</i>	<i>Unité</i>
<i>Name</i>	Nom de la ligne	String	-
<i>Tie Line</i>	Si la ligne est normalement ouverte	Boolean	-
<i>In service</i>	Si la ligne est en service	Boolean	-
<i>Resistance R (p.u)</i>	Résistance totale de la ligne	Float	p.u
<i>Reactance X (p.u)</i>	Réactance totale de la ligne	Float	p.u
<i>Tap ratio</i>	Rapport de transformation	Float	p.u
<i>MVA rating</i>	Capacité maximale de la ligne	Float	p.u
<i>AngMax</i>	Décalages maximal angles	Integer	Degrés
<i>AngMin</i>	Décalages minimal angles	Integer	Degrés

4.3.3 Données du générateur

Le générateur est un élément typiquement câblé en dérivation dans un réseau électrique, il est responsable de la génération de la puissance au jeu de barres auquel il est connecté. Les figures (4.8, 4.9 et Figure 4.10) montrent la boîte de dialogue associée à cet élément.

Generator1 - Linked to Bus: 2

Name: Generator1

Slack Generator In Service

Costs | Power and Voltage Control | Unit Commitment

Cost coefficients

a (\$/MWh ²)	0.002
b (\$/MWh)	10
c (\$)	0

OK Cancel Help

Figure 4.8 La boîte de dialogue du générateur: Coûts

Generator1 - Linked to Bus: 2

Name: Generator1

Slack Generator In Service

Costs | Power and Voltage Control | Unit Commitment

Power Control

MW Output	0
Min. MW output	0
Max. MW output	1000
mBase (MW)	100

Voltage Control

Mvar Output	0
Min. Mvar output	-1000
Max. Mvar output	1000
Voltage (p.u)	1

OK Cancel Help

Figure 4.9 La boîte de dialogue du générateur: puissances et contrôle de tension

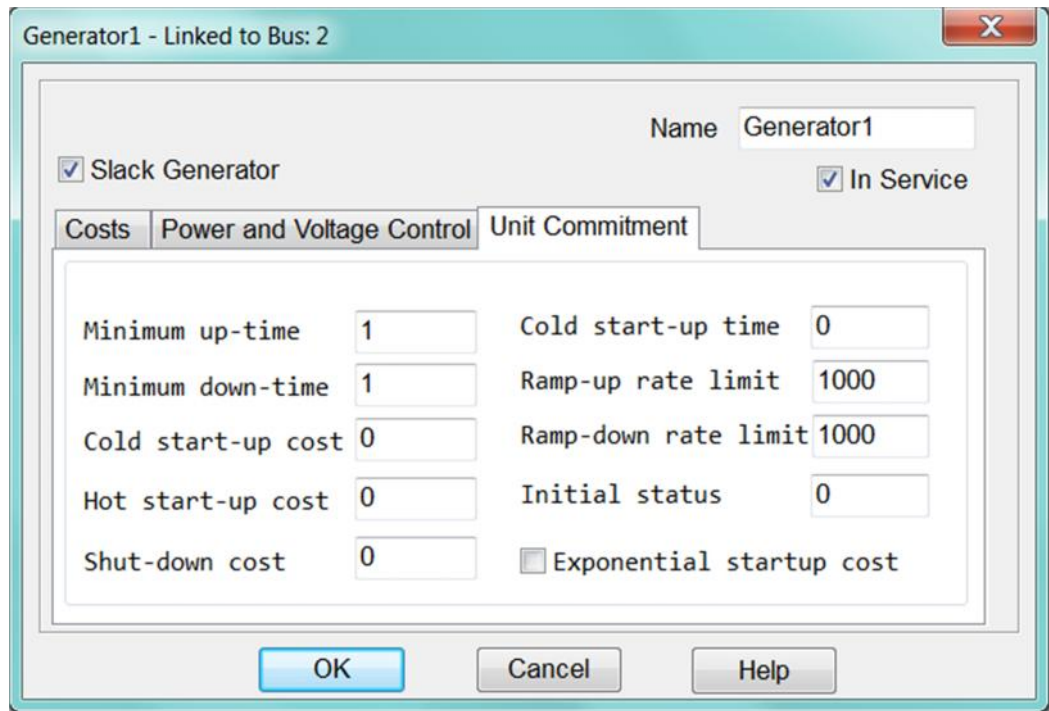


Figure 4.10 La boite de dialogue du générateur: Unit Commitment

Tableau 4.3 Données du générateur

<i>Champ</i>	<i>Description</i>	<i>Type</i>	<i>Unité</i>
<i>Name</i>	Nom du générateur	String	-
<i>Slack Generator</i>	<i>S'il est un générateur balancier</i>	Boolean	-
<i>In service</i>	Si le générateur est en service	Boolean	-
<i>a, b, c</i>	Les coefficients du coût quadratique	Float	\$
<i>MW output</i>	Puissance active délivrée	Float	MW
<i>MVar output</i>	Puissance réactive délivrée	Float	MVar
<i>Min. MW output</i>	Limite min. de Puissance active délivrée	Float	MW
<i>Max. MW output</i>	Limite max. de Puissance active délivrée	Float	MW
<i>Min. MVar output</i>	Limite min. de Puissance réactive délivrée	Float	MVar
<i>Max. MVar output</i>	Limite max. de Puissance réactive délivrée	Float	MVar
<i>mBase (MW)</i>	Puissance de base	Float	MW
<i>Voltage (p.u)</i>	Niveau de tension	Float	p.u
<i>Min. up time</i>	Temps minimal de marche	Integer	h
<i>Min. down time</i>	Temps minimal d'arrêt	Integer	h
<i>Cold start-up cost</i>	Coût de redémarrage à froid	Float	\$
<i>Hot start-up cost</i>	Coût de redémarrage à chaud	Float	\$
<i>Shut-down cost</i>	Coût d'extinction	Float	\$

<i>Cold start-up time</i>	Durée de redémarrage à froid	Float	\$
<i>Ramp-up rate limit</i>	Élévation maximale de la puissance	Float	MW
<i>Ramp-down rate limit</i>	Chute maximale de la puissance	Float	MW
<i>Initial status</i>	état initial	Float	h
<i>Exponential startup cost</i>	Si la fonction du cout de redémarrage est de forme exponentielle	Float	\$

4.3.4 Données de générateur d'énergie dispersé (DG)

Le générateur d'énergie dispersé nécessite un seul jeu de barres pour se connecter dans le réseau de distribution. Dans la présente version de ce logiciel on a considéré deux types de DGs, à savoir les unités de génération éolienne (WTGU) et les unités de génération photovoltaïque (PV). L'utilisateur a le choix d'introduire directement les puissances active et réactive injectée dans le réseau ou de laisser au logiciel de calculer ces puissances en fonction des paramètres internes des DGs (Annexe C). Les figures (Figure 4.11) et (Figure 4.12) montrent la boîte de dialogue associée à cet élément dans les deux cas.

Figure 4.11 La boîte de dialogue d'un DG de type photovoltaïque

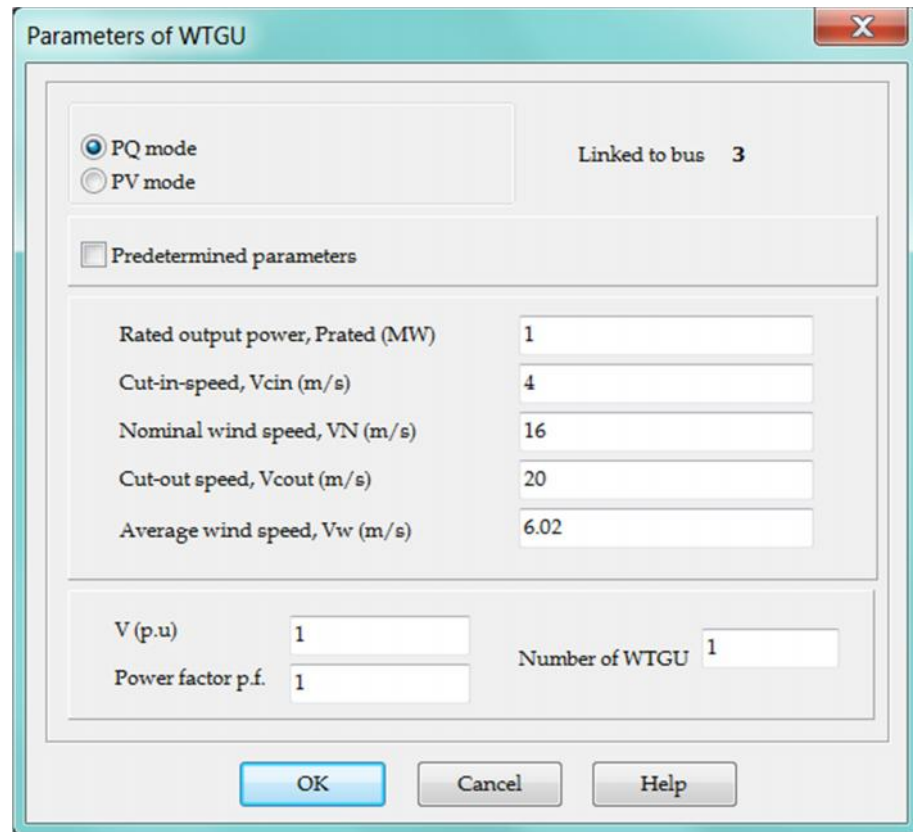


Figure 4.12 La boîte de dialogue d'un DG de type éolienne

4.4 Utilisation du logiciel

L'utilisation du logiciel comporte les étapes principales suivantes :

- Conception graphique du schéma unifilaire du réseau à étudier par la création des objets physiques du réseau ;
- Spécifier les données des objets ;
- Choisir une application à exécuter et spécifier ses propres données ;
- Affichage des résultats dans la fenêtre d'exécution de l'application choisie.

Chaque objet réseau crée possède une liste de jeux de barres, une liste de lignes, une listes de transformateurs, une liste de générateurs et une liste de générateurs dispersés. Ces listes sont construites par une simple liaison à la base de données.

En général, les données introduites dans les phases précédentes sont utilisées d'une manière ou d'une autre par toutes les applications du logiciel. En choisissant une application précise à exécuter, des données supplémentaires sont nécessaires.

4.4.1 Exécution de l'application UCP

Pour l'exécution de cette application, une fenêtre de simulation est générée (Figure 4.13), sur la même fenêtre les résultats sont visualisés.

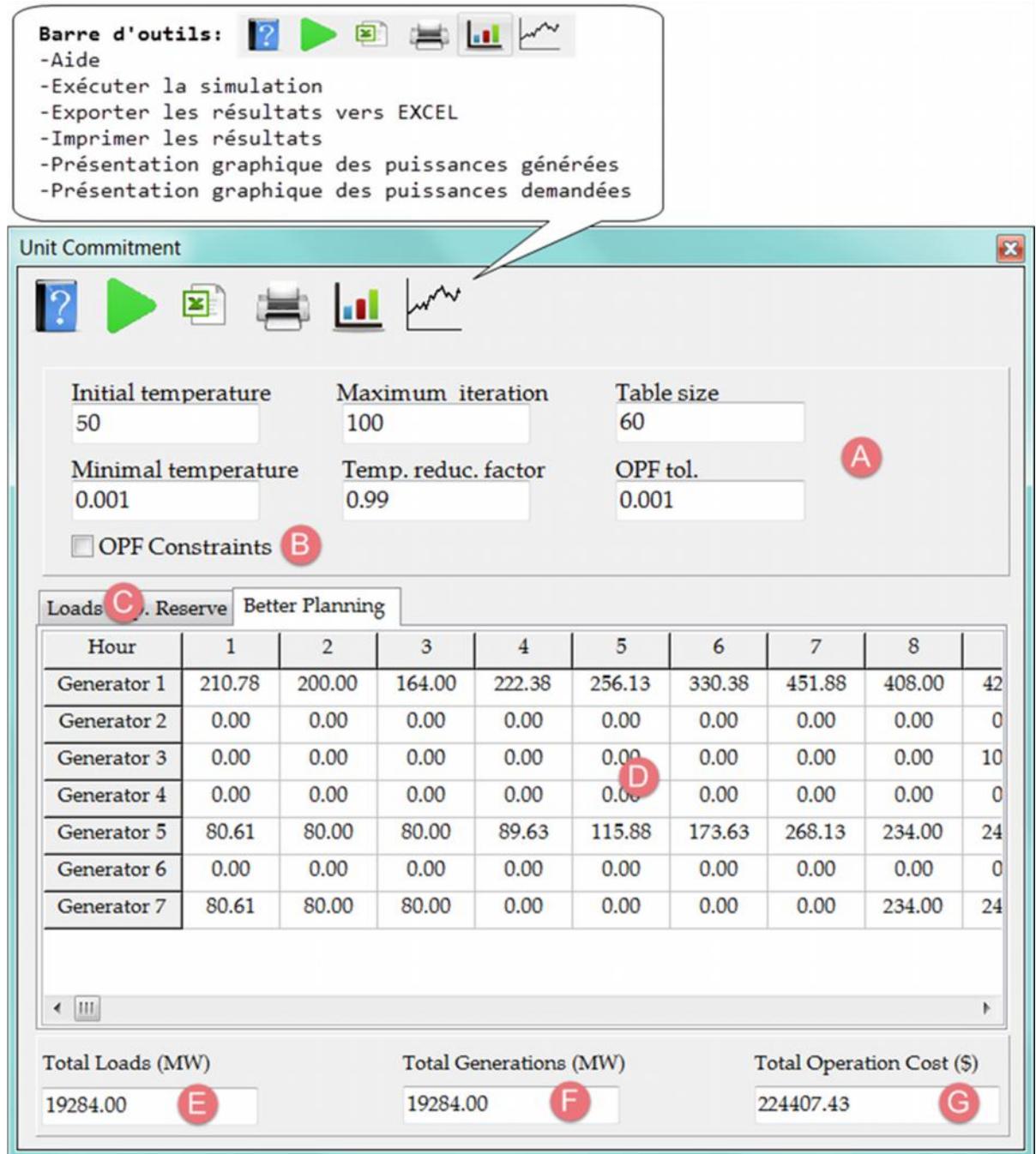
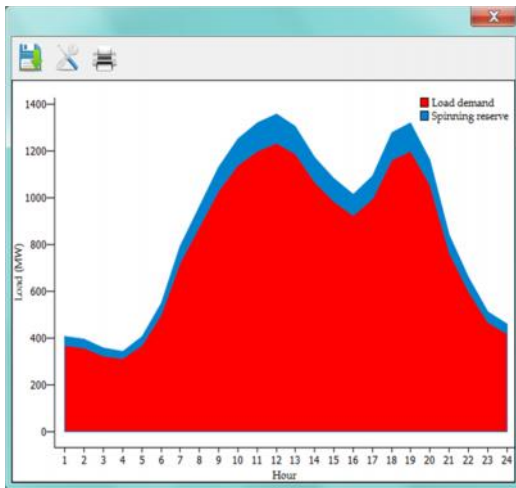


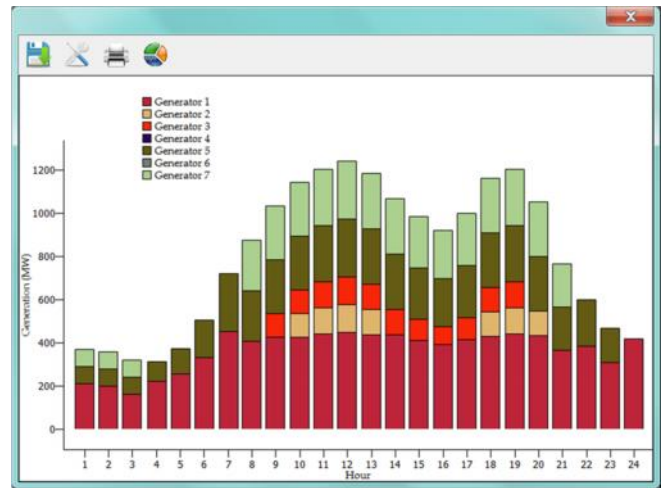
Figure 4.13 Fiche de l'interface de l'application UCP

Pour l'exécution du programme la méthode d'optimisation RT/RS pour résoudre le problème de l'UCP, l'utilisateur doit choisir:

- Les paramètres de la méthode d'optimisation RT/RS dans la partie **A** de la fenêtre (la température initiale, le facteur de réduction de température, le nombre d'itérations et la taille de la liste tabou).
- La considération (ou la non-considération) des contraintes d'écoulement de puissance optimale dans la case (OPF Constraints) dans la partie **B**.
- Les charges et les réserves tournantes durant les 24 heures en MW dans la partie **C**.



(a)



(b)

Hour	1	2	3	4	5	6	7	8	9
Generator 1	210.78	200	164	222.38	256.13	330.38	451.88	408	42
Generator 2	0	0	0	0	0	0	0	0	10
Generator 3	0	0	0	0	0	0	0	0	0
Generator 4	0	0	0	0	0	0	0	0	0
Generator 5	80.61	80	80	89.63	115.88	173.63	268.13	234	24
Generator 6	0	0	0	0	0	0	0	0	0
Generator 7	80.61	80	80	0	0	0	0	234	24

(c)

Figure 4.14 Représentation graphique et tabulaire de l'UCP

- (a) Charges et réserves tournantes
- (b) Puissances délivrées par les générateurs
- (c) Résultat tabulaire exporté au format Excel

Les résultats de l'Unit Commitment sont affichées sur la même fenêtre, ces résultats sont les puissances délivrées par les générateurs en MW dans la partie **D**, la puissance totale demandée en MW dans la partie **E**, la puissance totale générée en MW dans la partie **F** et le coût total de production en \$ dans la partie **G** de la fenêtre. La Figure 4.14 montre les représentations graphiques et tabulaires des résultats générées par le programme. L'utilisateur peut consulter un fichier d'aide intégré indépendamment. Les résultats peuvent être imprimés, enregistrés ou exportés vers Microsoft Excel.

4.4.2 Exécution de l'application OTEDG

De la même manière que l'application précédente, l'exécution de l'application OTEDG (Optimisation de la taille et de l'emplacement des DGs) engendre une fenêtre de simulation (Figure 4.15), sur la même fenêtre les résultats sont visualisés une fois les données propres de l'application courants de l'OTEDG sont spécifiées et l'application est exécutée.

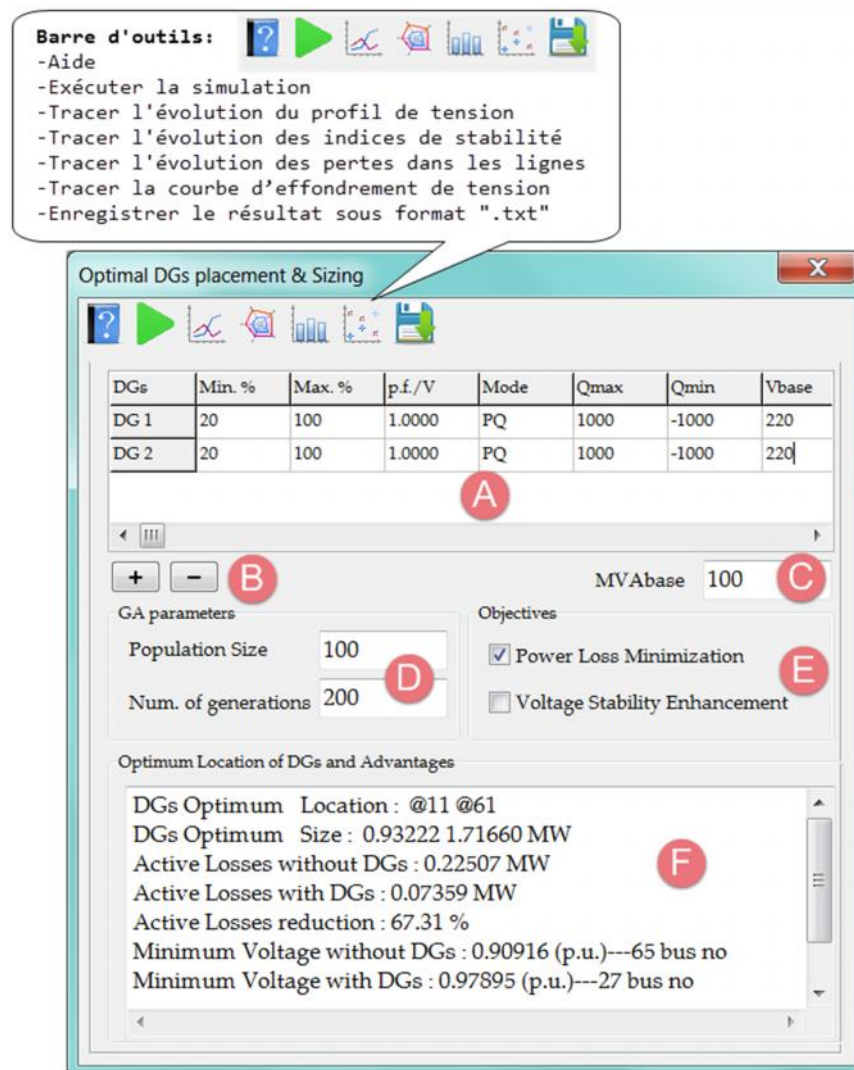
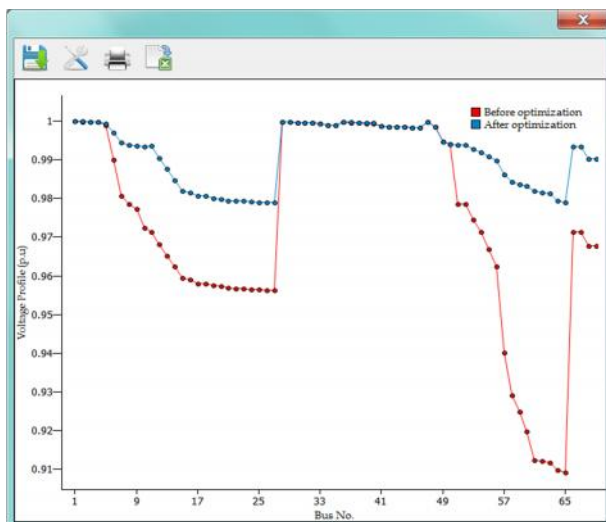


Figure 4.15 Fiche de l'interface de l'application OTEDG

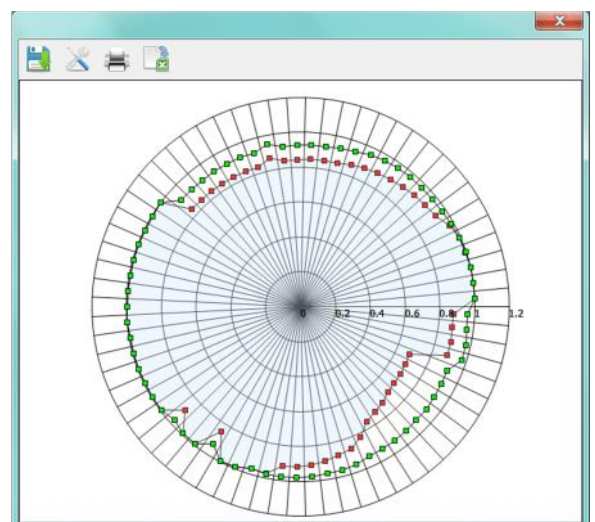
Pour l'exécution du programme de l'OTEDG, l'utilisateur doit choisir:

- Les paramètres des générateurs dispersés tels que les limites de puissances actives délivrées, le facteur de puissance, le niveau de tension, le mode de fonctionnement (PV ou PQ), les limites des puissances réactives délivrées et la tension de base dans la partie **A** de la fenêtre.
- Le nombre des générateurs dispersés en cliquant sur le bouton + ou – dans la partie **B**.
- La puissance de base dans la partie **C**.
- Les paramètres de l'algorithme génétique (pour le cas mono-objectif), ou de l'NSGA-II (pour le cas bi-objectifs) dans la partie **D**.
- Les fonctions objectifs (minimisation des pertes actives ou amélioration de la stabilité de tension, ou les deux en même temps) dans la partie **E**.

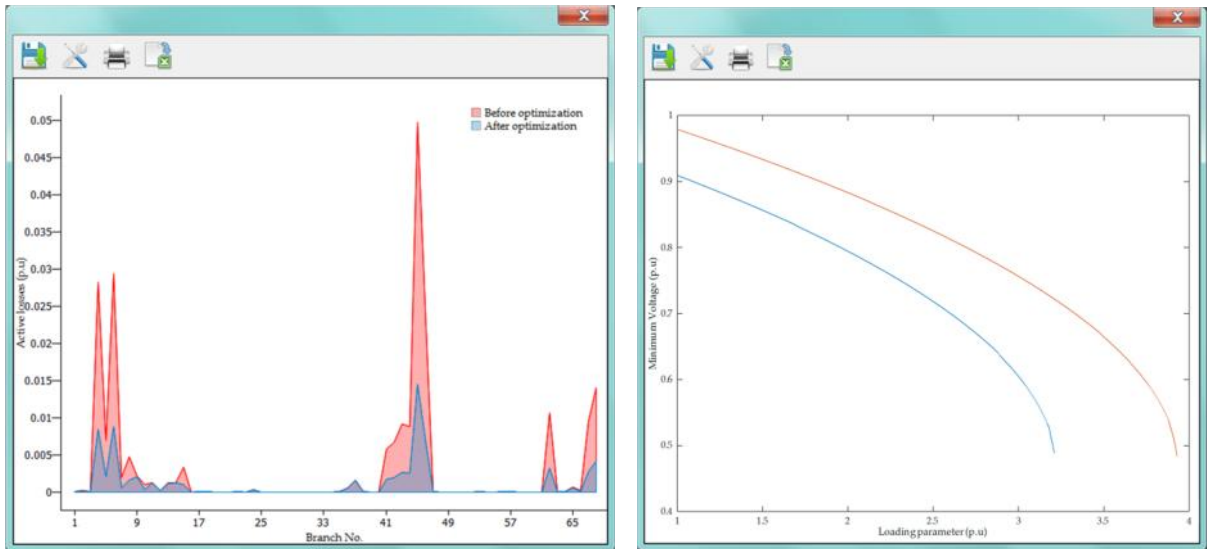
Les résultats de l'OTEDG sont affichées dans la partie **D** de la même fenêtre, ces résultats sont les emplacements et les tailles optimales des DGs, les pertes actifs avant et après l'emplacement des DGs, le taux de réduction des pertes, les tensions minimales avants et après l'emplacement des DGs avec les numéros des nœuds correspondants. La Figure 4.16 montre les représentations graphiques et tabulaires des résultats générées par le programme. L'utilisateur peut consulter un fichier d'aide intégré indépendamment. Les résultats peuvent être imprimés, enregistrés ou exportés vers Microsoft Excel.



(a)



(b)



(c)

(d)

Figure 4.16 Représentation graphique et tabulaire de l'OTEDG

- (a) Niveau de tension à chaque jeu de barres
- (b) Indice de stabilité à chaque jeu de barres
- (c) Pertes actifs à chaque ligne
- (d) Tension minimale en fonction du facteur de charge

4.4.3 Exécution de l'application RORD

De la même manière que les deux applications précédentes, l'exécution de l'application RORD (Reconfiguration Optimale des Réseaux de Distribution) engendre une fenêtre de simulation (Figure 4.17), sur la même fenêtre les résultats sont visualisés une fois les données propres de l'application courants de la RORD sont spécifiées et l'application est exécutée.

Pour l'exécution du programme de la RORD, l'utilisateur doit choisir:

- Les paramètres de l'algorithme génétique (pour le cas mono-objectif), ou de l'NSGA-II (pour le cas bi-objectifs) dans la partie **A**.
- Les fonctions objectifs (minimisation des pertes actives ou amélioration de la stabilité de tension, ou les deux en même temps) dans la partie **B**.
- La puissance de base dans la partie **C**.

Les résultats de la RORD sont affichées dans la partie **D** de la même fenêtre, ces résultats sont : la configuration initiale et la configuration optimale (les branches mises hors service), le taux de réduction des pertes, les tensions minimales avants et après la reconfiguration avec les numéros des nœuds correspondants. La Figure 5.17 montre les représentations graphiques et tabulaires des résultats générées par le programme. L'utilisateur

peut consulter un fichier d'aide intégré indépendamment. Les résultats peuvent être imprimés, enregistrés ou exportés vers Microsoft Excel.

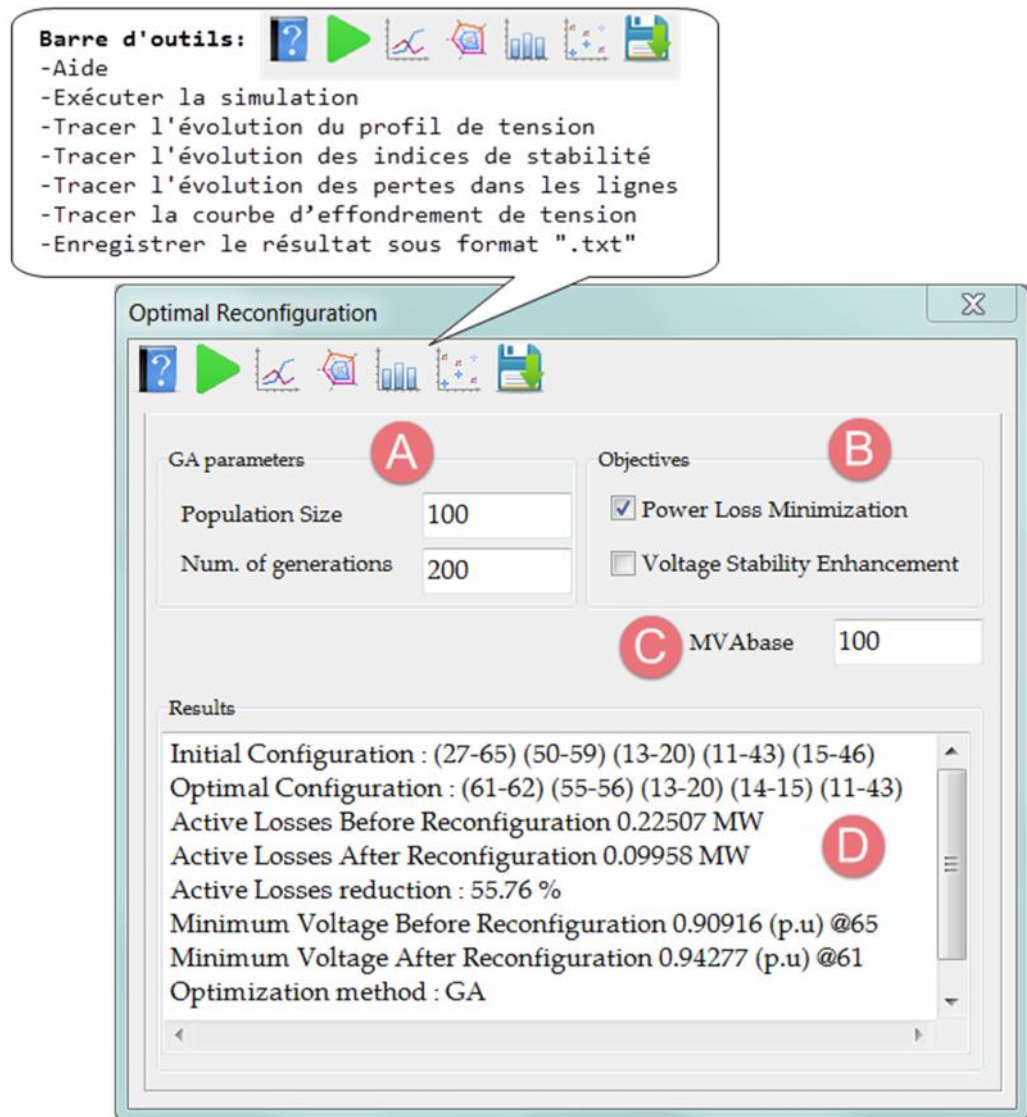


Figure 4.17 Fiche de l'interface de l'application RORD

4.5 Conclusion

Dans ce chapitre, on a exploré l'atout de la programmation orientée objet dans la simulation et l'analyse des systèmes électriques de puissance à l'aide de notre logiciel *VisualPower*. Ce logiciel peut être utilisé facilement dans le cadre de formation de l'ingénieur électricien. Il a aussi des possibilités sophistiquées nécessaires pour l'ingénieur professionnel.

Afin de valider précisément nos propositions, il nous faut à présent les mettre en œuvre de manière pratique. Nous devons donc réaliser des tests sur des réseaux électriques et apprécier les résultats obtenus. Cette mise en œuvre et les résultats qui en découlent font l'objet du chapitre 5.

Chapitre 5

Tests et Applications aux Réseaux Electriques

Chapitre 5: Tests et Applications aux Réseaux Electriques

5.1 Introduction

Dans la première partie de ce chapitre, nous avons appliqué la méthode hybride RT/RS avec l'approche orientée objet au problème d'engagement des turbines optimal avec les contraintes d'écoulement de puissances UC-EPO présentée antérieurement. La méthode d'écoulement de puissance optimal de Newton a été utilisée pour satisfaire les contraintes d'égalités et d'inégalités et pour déduire toutes les variables d'état dans les réseaux électriques. Nous avons élaboré des tests sur le réseau modèle IEEE 59-nœuds avec 7 générateurs et le réseau algérien de l'année 1997 qui comprend 10 générateurs, les résultats de simulation sont comparés avec d'autres résultats obtenus dans la littérature.

Dans la deuxième partie de ce chapitre, nous avons utilisé la méthode d'optimisation multi-objective NSGA-II pour deux applications sur les réseaux électriques de distribution. La première application est la détermination optimale de la taille et de l'emplacement de la production décentralisée. La seconde application est la reconfiguration optimale des réseaux électrique de distribution. L'objectif est de minimiser les pertes actifs et d'améliorer la stabilité de tension. Nous avons élaboré des tests sur le réseau modèle IEEE 33-nœuds, les résultats de simulation sont comparés avec ceux trouvés dans la littérature.

Toutes les approches proposées ont été développées sur notre logiciel *VisualPower* et ce par utilisation d'un ordinateur doté d'un processeur Intel® Core™ i5, 2.50 GHz et une mémoire vive de 8 GO.

5.2 Application de la méthode RT/RS au problème de l'UC

La méthode RT/RS a été appliquée pour résoudre le problème de l'unit commitment sur deux réseaux test et pour deux cas différents (*sans et avec les* contraintes d'écoulement de puissance optimal). Les deux réseaux test étudiés sont: le réseau standard IEEE 30- nœuds et le réseau algérien de 59-nœuds. Dans notre étude, taille maximale de la liste taboue RT et la

température initiale de recuit simulé RS ont été choisis respectivement égaux à 100 et 60, alors que le nombre maximal d'itérations égal à 1000.

5.2.1 Réseau IEEE 57-nœuds

Le réseau électrique test IEEE 57- nœuds, présenté en annexe A, est un réseau standard qui représente une portion du système de puissance électrique Américain (in the U.S. Midwest) pour Décembre 1961. Ce réseau est constitué de 57 jeux de barres et 7 générateurs liées aux jeux de barres n°=1,2,3,6,8,9 et 12 injectant leurs puissances à un système alimentant 50 charges (PQ bus) à travers 80 lignes de transport. La puissance demandée est montrée dans la Figure 5.1.

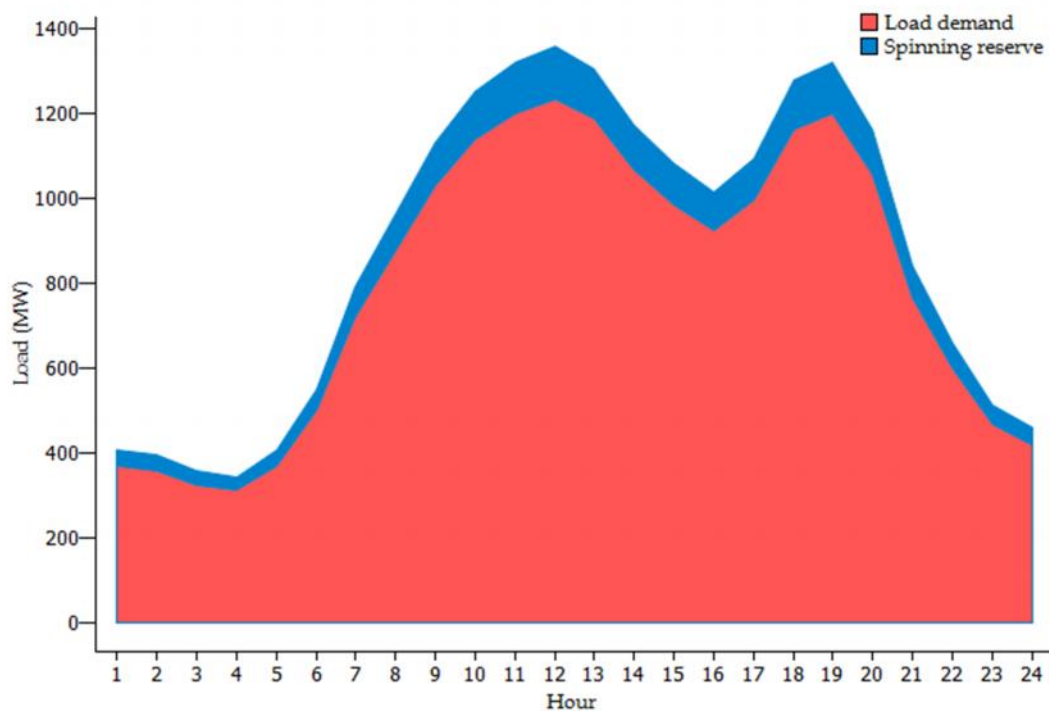


Figure 5.1 Profil de la charge du réseau IEEE 57 nœuds sur un intervalle de 24 h

a) UC traditionnel

Dans le premier cas, le problème de l'unit commitment a été résolu par la méthode RS/RT sans tenir compte aux contraintes d'écoulement de puissance optimal. Le Tableau 5.1 présente la planification optimale des 7 générateurs du système. Le coût total de production est trouvé \$225,111.58.

b) UC avec les contraintes de l'EPO

Dans le deuxième cas, le problème de l'unit commitment a été résolu par la méthode RS/RT en tenant compte aux contraintes de l'écoulement de puissances optimale tels que les

limites des tensions des nœuds de charge et les limites des lignes de transmission. Le Tableau 5.2 présente la planification optimale des 7 générateurs. Les puissances délivrées par les unités de production sont présentés sur la Figure 5.2. Le cout total de production est trouvé \$230,497.52, on remarque que le coût a augmenté par rapport au premier cas à cause de l'ajout des contraintes au problème d'optimisation.

Tableau 5.1 Meilleure planification du système IEEE 57-nœuds pour le cas 1

unit	Hour (Demand)											
	1(372)	2(360)	3(324)	4(312)	5(372)	6(504)	7(720)	8(876)	9(1032)	10 (1140)	11(1200)	12(1236)
1	210,78	200	164	222,38	256,13	330,38	346,96	408	426,27	425,95	440,54	490,17
2										109,07	120,42	150
3									109,32	109,07	120,42	
4												
5	80,61	80	80	89,63	115,88	173,63	186,52	234	248,21	247,96	259,31	297,91
6												
7	80,61	80	80				186,52	234	248,21	247,96	259,31	297,91
unit	Hour (Demand)											
	13(1188)	14(1068)	15(984)	16(924)	17(996)	18(1164)	19(1200)	20(1056)	21(768)	22 (600)	23(468)	24(420)
1	473,07	483,13	450,26	393,87	390,92	431,78	476,67	478,43	346,17	354,35	282	420
2	145,72				81,83	113,61	148,52					
3				84,12	81,83	113,61			50	53,38	50	
4												
5	284,61	292,43	266,87	223,01	220,71	252,5	287,41	288,78	185,91			
6												
7	284,61	292,43	266,87	223,01	220,71	252,5	287,41	288,78	185,91	192,27	136	
Total operation cost (\$)											225,111.58	

Tableau 5.2 Meilleure planification du système IEEE 57-nœuds pour le cas 2

Unité	Heure (charge)											
	1(372)	2(360)	3(324)	4(312)	5(372)	6(504)	7(720)	8(876)	9(1032)	10 (1140)	11(1200)	12(1236)
1	196.68	192.26	166.65	214.45	247.88	320.65	322.89	378.83	388.08	440.17	465.91	488.09
2										49.57	55.76	49.16
3									105.48	68.79	98.85	121.52
4												
5	90.31	86.47	80	102.18	130.27	193.87	204.15	256.14	279.91	300.13	300.34	300.42
6												
7	88.6	84.72	80				202.76	254.96	276.27	300.11	300.34	300.42
Unité	Heure (charge)											
	13(1188)	14(1068)	15(984)	16(924)	17(996)	18(1164)	19(1200)	20(1056)	21(768)	22 (600)	23(468)	24(420)
1	465.12	385.26	378.02	362.07	381.16	450.9	471.28	323.81	318.14	257.68	210.02	243.94
2	49.38	49.87				49.45	49.33	150.27	50	50	50	50
3	94.48	49.87	98.3	84.94	100.98	83.93	100.12					
4												
5	300.31	300.07	262.04	245.88	265.34	300.08	300.34	300.13	207.14	151.57	108.21	
6												
7	300.31	300.07	262.07	245.76	265.35	300.08	300.34	300.13	204.8	149.02	105.68	135.62
Coût total de production (\$)											230,497.52	

Afin de montrer l'efficacité de l'algorithme, une comparaison en terme de coût de production total a été effectuée avec une autre méthode présentée dans la littérature, à savoir la méthode hybride Relaxation Lagrangienne/Algorithme Génétique (LRGA) [84]. Le

Tableau 5.3 résume les résultats obtenus. On constate bien que la méthode proposée a donné le meilleur résultat comparativement à cette technique.

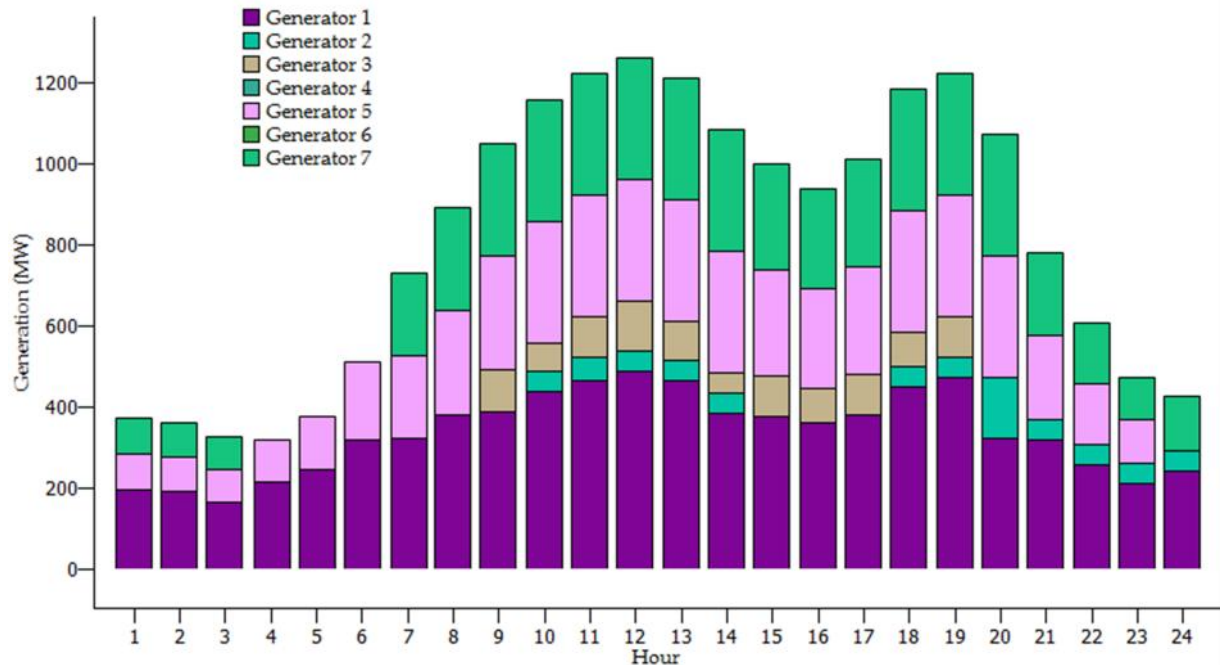


Figure 5.2 Les valeurs optimales des puissances générées (IEEE 57-nœuds)

Tableau 5.3 Comparaison entre LR/GA et RT/RS (IEEE 57-bus)

		Algorithmes	
		LR/GA [84]	RT/RS
Coût total de production (\$)	<i>Cas 1</i>	225,338.49	225,111.58
	<i>Cas 2</i>	230,878.90	230,497.52

5.2.2 Réseau Algérien 59-nœuds

En Algérie, la plus grande partie de l'électricité est d'origine thermique (98.3%), le reste se répartissant entre les centrales hydro-électriques ou à diesel*. Le transport de l'énergie s'effectue par transformation du courant triphasé produit à des tensions variables, en une tension unique de 220 kV, distribuée avec des lignes haute tension. Des sources d'énergie renouvelables telles que le vent et le soleil produisent de l'énergie électrique dans les sites isolés de l'Algérie. Elles représentent cependant des quantités négligeables.

* D'autres informations sur le réseau Algérien peuvent être consultées au niveau du site officiel de Sonelgaz : <http://www.sonelgaz.dz>

Nous avons testé le logiciel *VisualPower* pour résoudre le problème de l'unit commitment en utilisant la méthode RT/RS sur le réseau algérien (de l'année 1997). Ce réseau est composé de 59 jeux de barres, 10 générateurs, 49 charges (PQ bus) et 83 lignes de transmission (Annexe A). Le schéma unifilaire du réseau est présenté dans la Figure 5.3. Les données techniques et économiques des générateurs sont données dans le Tableau 5.4.

Tableau 5.4 Données techniques et économiques des générateurs du réseau test Algérien

Bus N°	$P_{i \min}$ (MW)	$P_{i \max}$ (MW)	r_i (\$/h)	S_i (\$/MWh)	χ_i (\$/MWh ²)	Coût de Redémarrage (\$)	Coût d'arrêt (\$)	MUT (h)	MDT (h)	\dagger_i (h)
1	8	72	0	1.50	0.0085	26	26	1	1	2
2	10	70	0	2.50	0.0170	17	17	2	2	2
3	30	510	0	1.50	0.0085	500	500	5	5	4
4	20	400	0	1.50	0.0085	500	500	5	5	4
13	15	150	0	2.50	0.0170	90	90	2	2	2
27	10	100	0	2.50	0.0170	55	55	2	2	2
37	10	100	0	2.00	0.0030	55	55	2	2	2
41	15	140	0	2.00	0.0030	90	90	2	2	2
42	18	175	0	2.00	0.0030	90	90	2	2	2
53	30	450	0	1.50	0.0085	500	500	5	5	4

a) UC traditionnel

Dans le premier cas, le problème de l'unit commitment a été résolu par la méthode RT/RT sans tenir compte aux contraintes d'écoulement de puissance optimal. Le tableau (5.6) présente la planification optimale des 10 générateurs du système. Le coût total de production est trouvé \$225,111.58.

b) UC avec les contraintes de l'EPO

Dans le deuxième cas, le problème de l'Unit Commitment a été résolu par la méthode RT/RT en tenant compte aux contraintes de l'écoulement de puissances optimale tels que les limites des tensions des nœuds de charge et les limites des lignes de transmission. Le tableau (5.7) présente la planification optimale des 10 générateurs. Les puissances délivrées par les unités de production sont présentés sur la Figure 5.4. Le cout total de production est trouvé \$230,497.52, on remarque que le coût a augmenté par rapport au premier cas à cause de l'ajout des contraintes au problème d'optimisation.

Tableau 5.5 Comparaison entre PSO et RT/RS (Algerian 59-bus)

		Algorithmes	
		PSO [85]	RT/RS
Coût total de production (\$)	Cas 1		47,958.52
	Cas 2	73,881.69	66,153.69

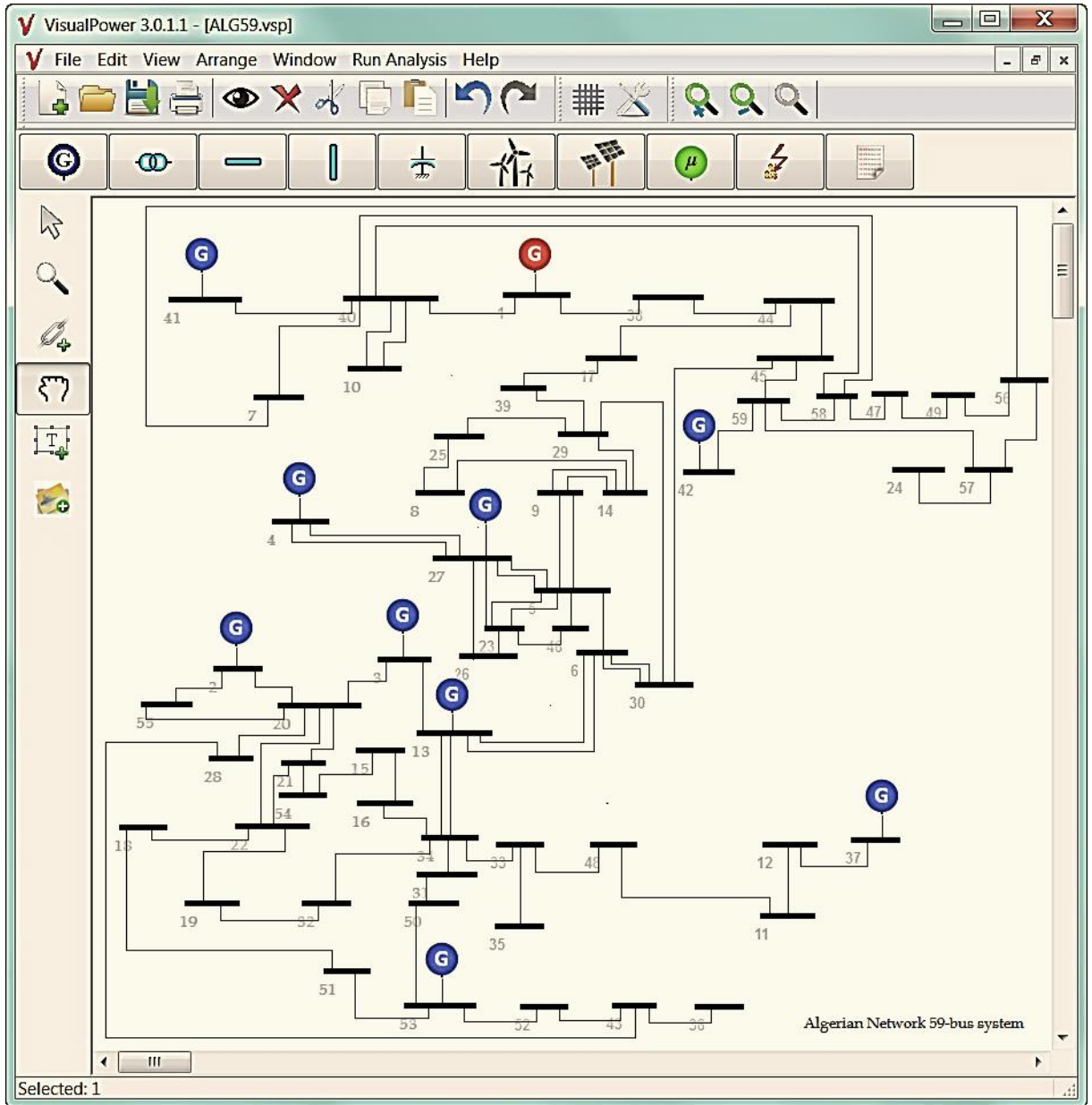


Figure 5.3 Schéma unifilaire du réseau de production et transport Algérien (1997)

Tableau 5.6 Meilleure planification du système Sonelgaz 59-nœuds pour le cas 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24		
1	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	
2			10	10	14.66	13.03	13.37	17.76	25.05	28.34			17.84	20.5	20.17	18.61	15.42	13.48				39.34	32.62	61.29		
3	81.59	78.42	76.46	76.35	88.14	84.88	85.57	94.35	108.92	115.5	129.61	130.27	94.5	99.83	99.16	96.05	89.67	85.78	108.94	120.94	146.94	137.5	124.07			
4	81.59	78.42	76.46	76.35									94.5	99.83	99.16	96.05	89.67	85.78							181.41	
5	15	15	15	15	15	15	15	17.76	25.05	28.34	35.39	35.73	17.84	20.5	20.17	18.61	15.42	15	25.06	31.06	44.06	39.34	32.62			
6	11.38	10	10	10	14.66	13.03	13.37	17.76	25.05	28.34	35.39	35.73	17.84	20.5	20.17	18.61	15.42	13.48	25.06	31.06	44.06	39.34	32.62	61.29		
7	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
8	140	138.87	133.31	132.98	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	140	
9	147.84	138.87	133.31	132.98	166.4	157.17	159.11	175	175	175	175	175	175	175	175	175	170.73	159.71	175	175	175	175	175	175	175	
10	81.59	78.42	76.46	76.35	88.14	84.88	85.57	94.35	108.92	115.5	129.61	130.27	94.5	99.83	99.16	96.05	89.67	85.78	108.94	120.94	146.94	137.5	124.07			
Coût de Production Total																										47,958.52 \$

Tableau 5.7 Meilleure planification du système Sonelgaz 59-nœuds pour le cas 2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24		
1	26.35	25.1	24.66	24.33	6.87	7.04	6.96	6.6	6.17	5.98	5.9	5.89	5.8	5.59	5.41	5.52	6.06	6.24	6.38	6.11	5.49	5.39	5.72	6.08		
2	8.58	8.75	8.8	8.79	8.84	9.01	8.93	8.56	8.11	9.65	13.71	12.83	15.42	21.77			8.22	8.19	8.33	8.06	29.07	33.05	17.82	8.02		
3	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30		
4	186.53	175.78	172.84	97.25	97.35	86.8	244.32	267.64	149.4	165.21	312.73	314.22	170.63	176.76	356.99	351.24	308.06	143.85	133.85	298.58	325.82	327.89	172.98	156.07		
5	26.21	25.11	24.73	28.79																						
6				51.76	100.91	100.78			101.48	101.64			101.78	101.95				101.42	101.31				101.84	101.56		
7	29.55	28.05	27.58	27.04	28.32	26.79	25.55	28.46	31.39	33.04	32.03	32.93	34.84	36.52	36.53	35.51	31.72	30.93	30.02	31.32	37.37	38.07	35.48	32.11		
8	44.76	42.46	41.55	47.49	87	83.43	76.69	85.73	102.52	106.76	103.58	103.87	110.44	114.53	110.33	107.68	100.13	100.89	97.93	98.33	111.57	113.55	112.01	104.47		
9	61.58	55.87	53.56	68.42	50.12	47.59	24.8	34.02	64.2	68.48	52.63	53.35	70.03	71.69	73.46	70.89	50.92	62.68	60.05	46.66	59.32	61.25	70.62	66.02		
10	345.04	347.45	348.27	347.73	312.39	309.6	286.1	288.59	312.25	307.33	286.62	286.47	311.96	316.82	248.44	246.35	283.5	312.13	311.95	293.19	293.29	294.65	313.83	312.24		
Coût de Production Total																										66,153.69 \$

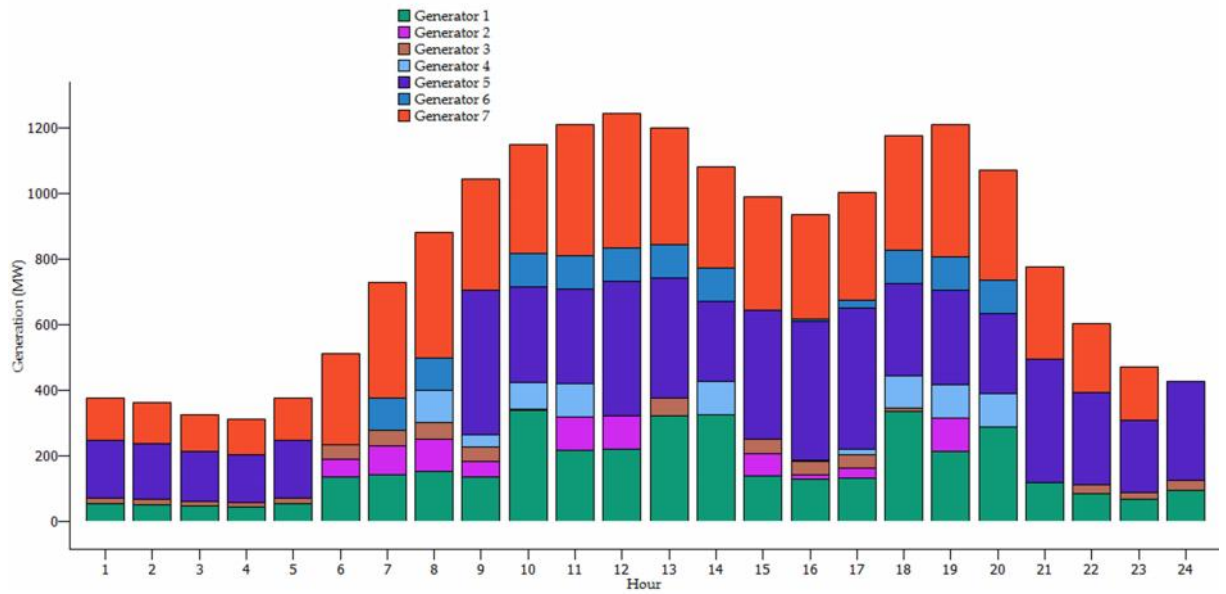


Figure 5.4 Les valeurs optimales des puissances générées (Sonelgaz 59-nœuds)

5.3 Optimisation de la taille et de l'emplacement des DGs dans les réseaux de distribution

Dans cette partie, l'algorithme NSGA-II est appliqué sur des réseaux de distribution de différentes tailles: IEEE 12-nœuds [86], 33-nœuds [87], 69-nœuds [88] et 118-nœuds [89]. Les données des réseaux tests sont présentées dans l'Annexe B. Notre objective est de trouver l'emplacement et les tailles optimales des générations dispersées (DG) dans ces réseaux afin de minimiser les pertes actives de transmission et d'améliorer la stabilité de la tension en utilisant l'indice (*SI_index*). Les paramètres de simulation de la méthode NSGA-II sont représentés dans le Tableau 5.8.

Tableau 5.8 Paramètres de simulations adoptées pour le NSGA-II

	Taille de la population	Max. N. de générations	Mutation
12-nœuds	50	100	0.05
33-nœuds	50	100	0.05
69-nœuds	100	200	0.05
118-nœuds	130	300	0.05

Le Tableau 5.9 montre l'efficacité de l'algorithme proposé sur les performances du système en comparaison avec le cas de base (sans DG). Pour trouver la capacité de charge maximale P_{max} on a réalisé un test d'augmentation de la charge sur les réseaux test avec un facteur d'augmentation compris entre [1 :10] jusqu'à ce que l'effondrement de tension est observée (Figure 5.14). Les résultats obtenus sont comparée avec ceux trouvés dans la littérature, à savoir, la méthode d'analytique [44], la recherche coucou (CS) [48], et l'optimisation par essaim de particules (PSO) [50]. Les positions et les et tailles optimales des DGs sont présentés dans le Tableau 5.10.

Discussion des résultats

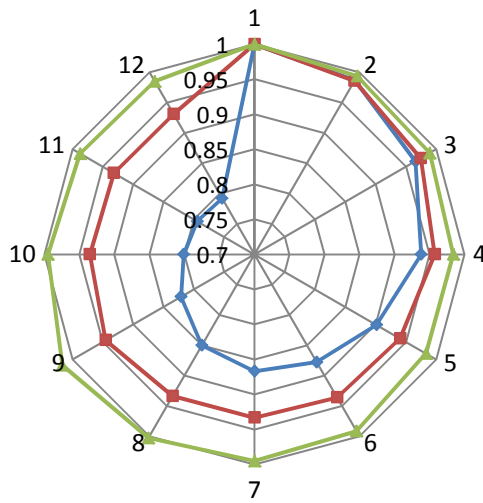
Comme on peut le voir à partir de la comparaison des colonnes (2) et (7) du Tableau 5.9, les pertes des puissances actives totales de tous les réseaux de test sont réduits de façon significative après l'installation des DGs. En outre, il peut être observé que la stabilité de la tension est améliorée, et la capacité maximale de charge des réseaux est augmentée après l'installation des DGs. Par conséquent, plus de charges peuvent être ajoutés sans causer l'effondrement de la tension.

La Figure 5.5 montre les indices de stabilité de la tension avant et après l'installation des DGs. On remarque bien qu'après l'installation DG, les indices de stabilité sont considérablement améliorés pour tous les réseaux test.

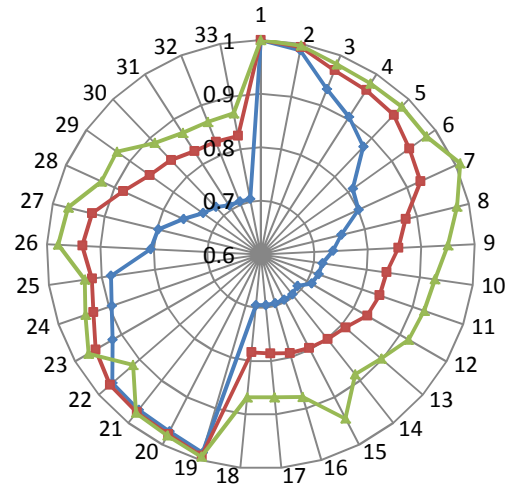
Le profil de tension de tous les réseaux test sont représentés les figures (5.6-5.9). Comme on peut le voir, les niveaux de tension à tous les nœuds pour tous les réseaux test sont améliorés et placés dans une marge acceptable.

Les figures (5.10-5.13) montrent les pertes actives dans les branches avant et après l'installation des DGs. On remarque aussi qu'après l'installation DG, les pertes sont considérablement réduites pour tous les réseaux test.

On peut voir dans le Tableau 5.10 et la Figure 5.5 que l'algorithme proposé atteint les mêmes ou de meilleurs résultats que d'autres méthodes publiées par d'autres auteurs. La Figure 5.14 démontre que la taille optimale et le meilleur emplacement de la DG ont amélioré la capacité maximale de charge pour tous les réseaux test.

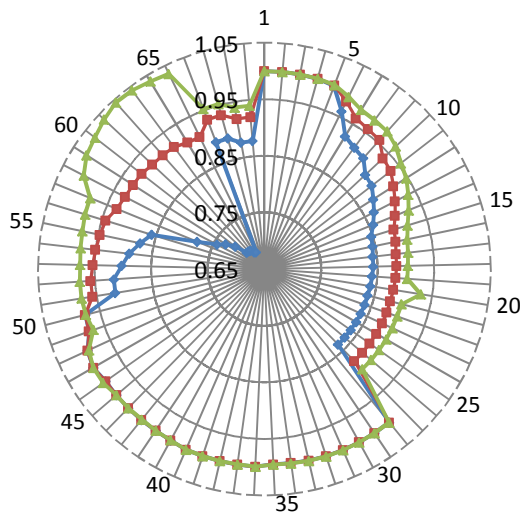


(a) 12-nœuds

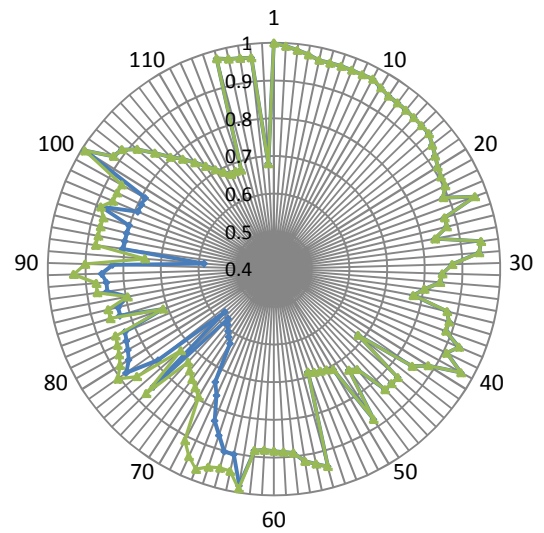


(b) 33-nœuds

—◆— Without DG —■— Analytical Method —▲— Proposed Method



(c) 69-nœuds



(d) 118-nœuds

—◆— Without DG —■— Analytical Method —▲— Proposed Method

Figure 5.5 Indice de stabilité à chaque jeu de barres avant et après l'installation du DG

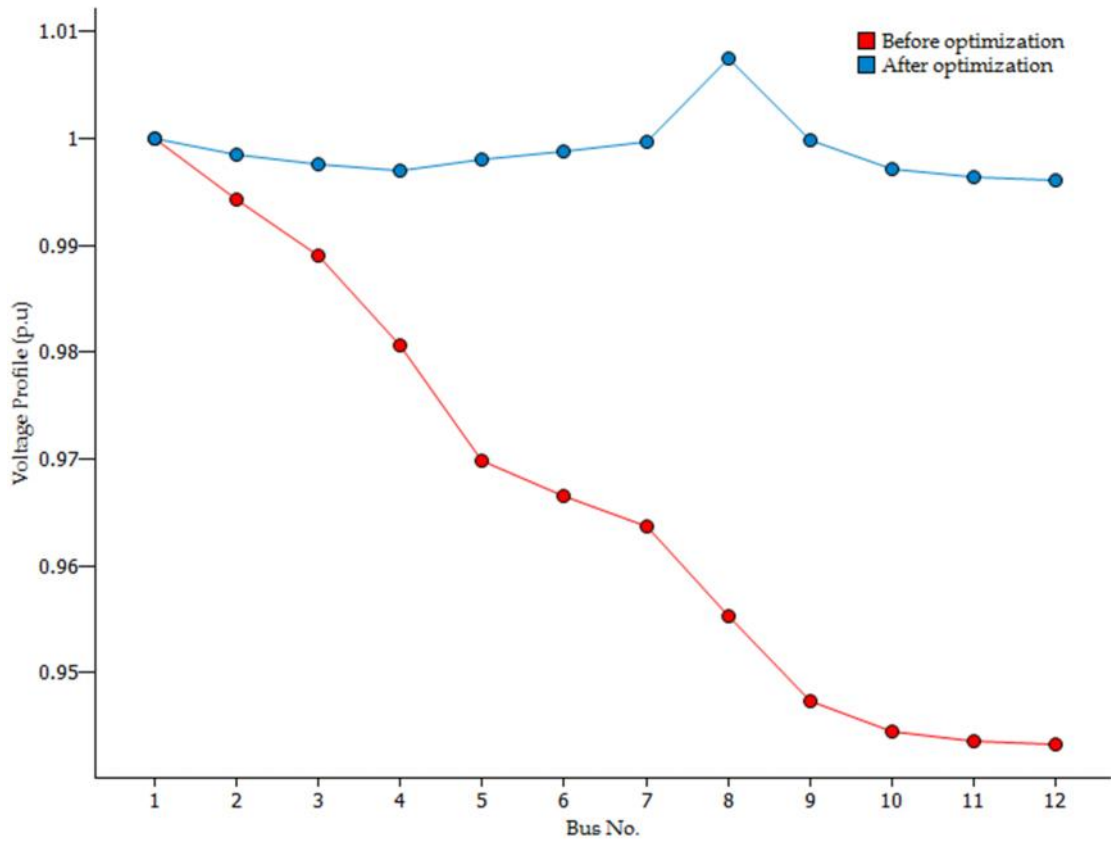


Figure 5.6 Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 12-nœuds avant et après l'installation du DG

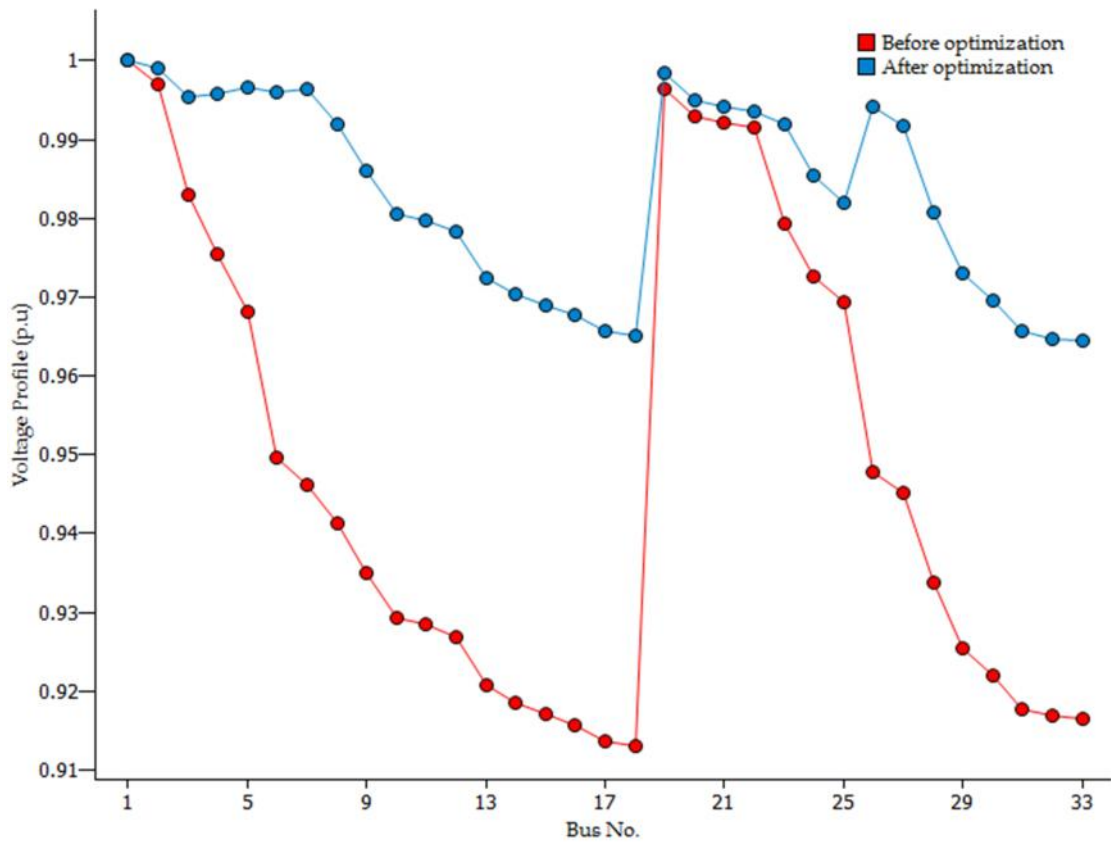


Figure 5.7 Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 33-nœuds avant et après l'installation du DG

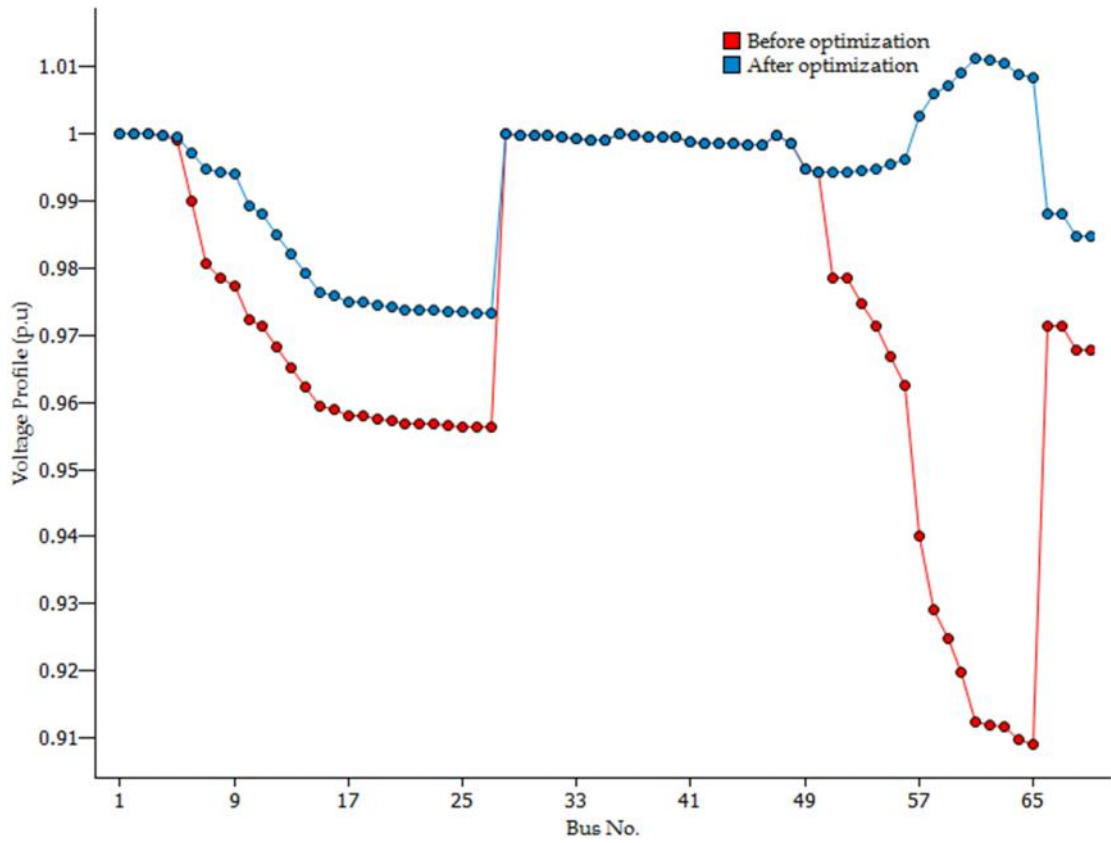


Figure 5.8 Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 69-nœuds avant et après l'installation du DG

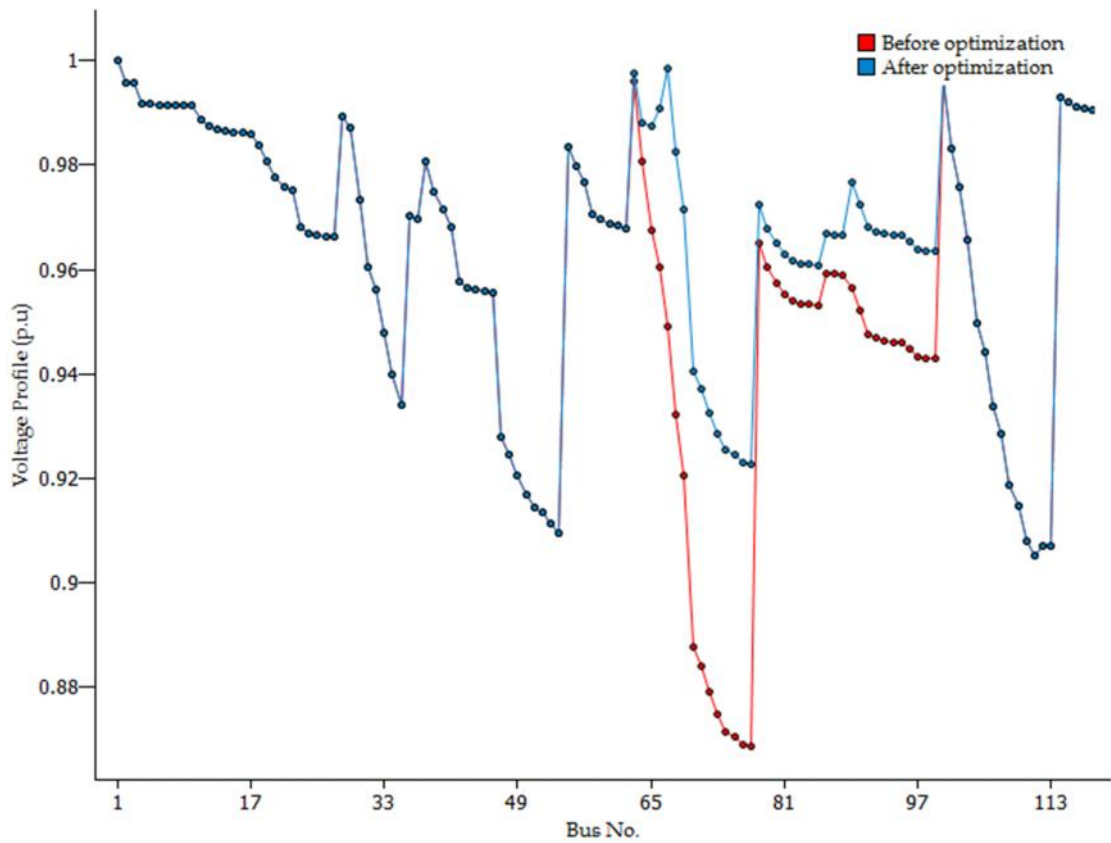


Figure 5.9 Niveau de tension à chaque jeu de barres du réseau de distribution IEEE 118-nœuds avant et après l'installation du DG

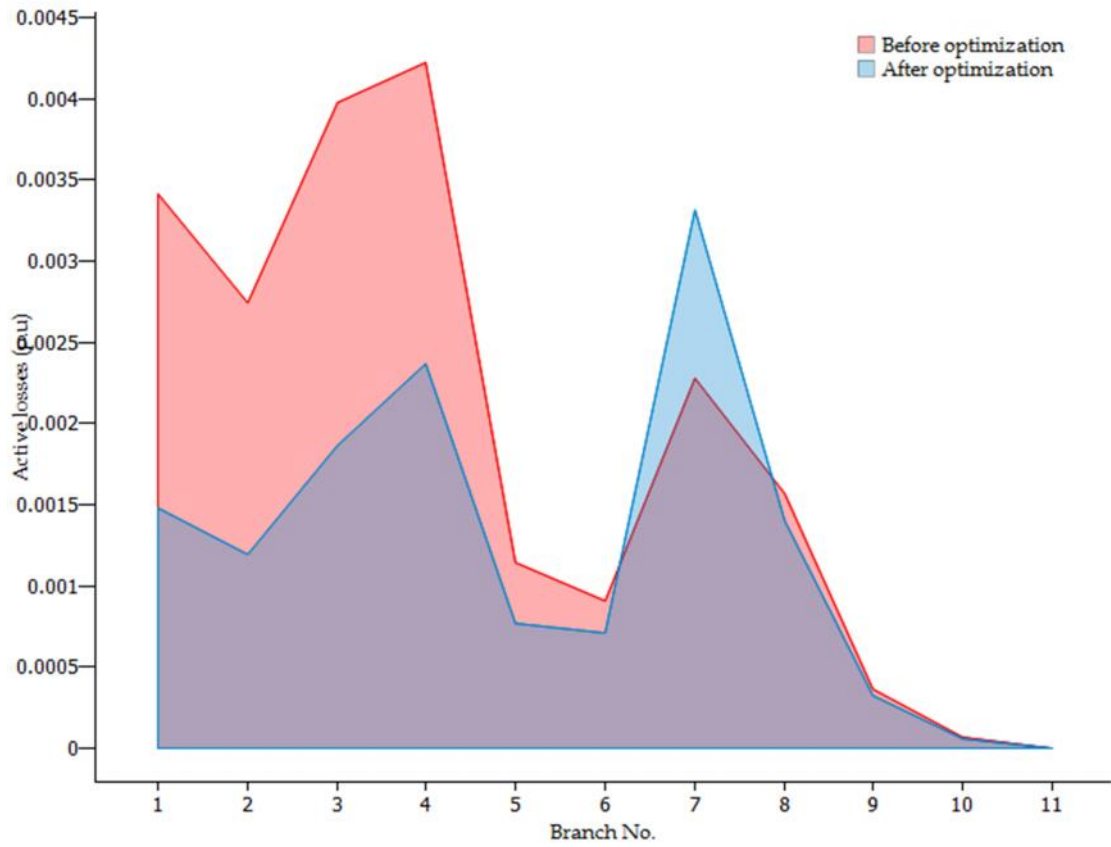


Figure 5.10 Pertes de puissance active au niveau de chaque branche du réseau IEEE 12-nœuds avant et après l'installation du DG

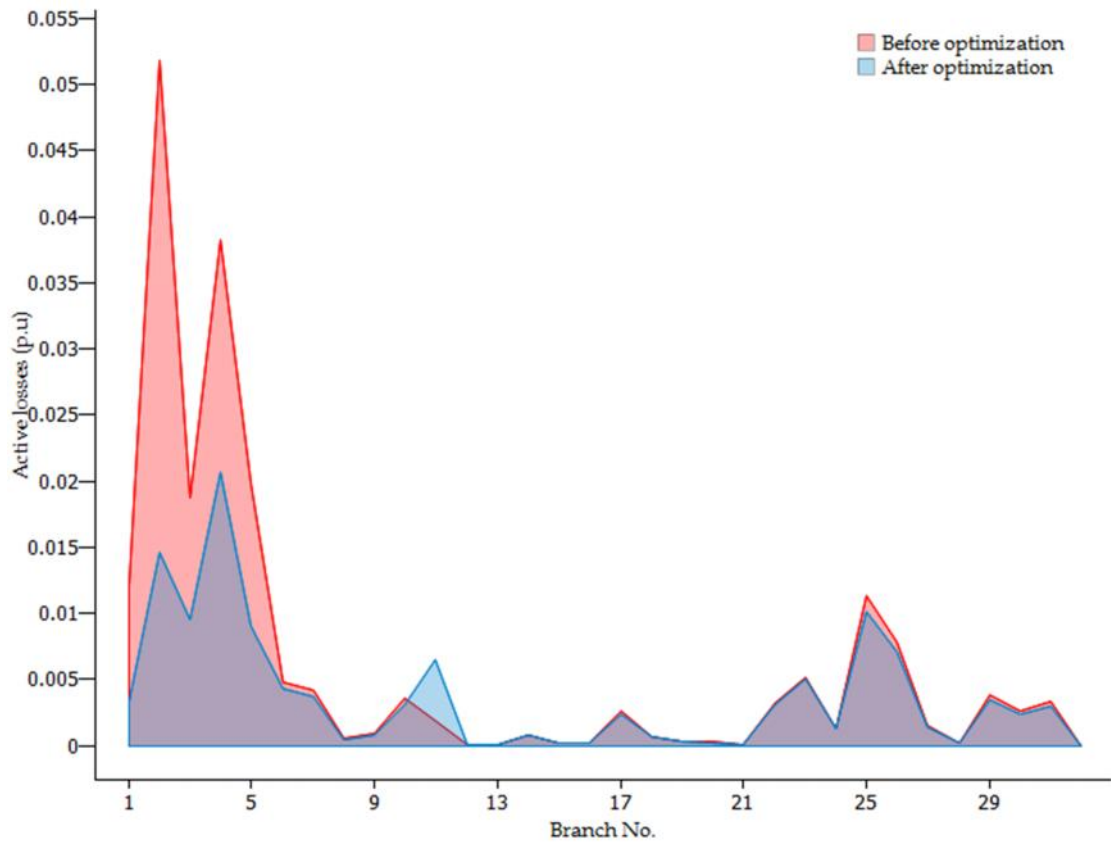


Figure 5.11 Pertes de puissance active au niveau de chaque branche du réseau IEEE 33-nœuds avant et après l'installation du DG

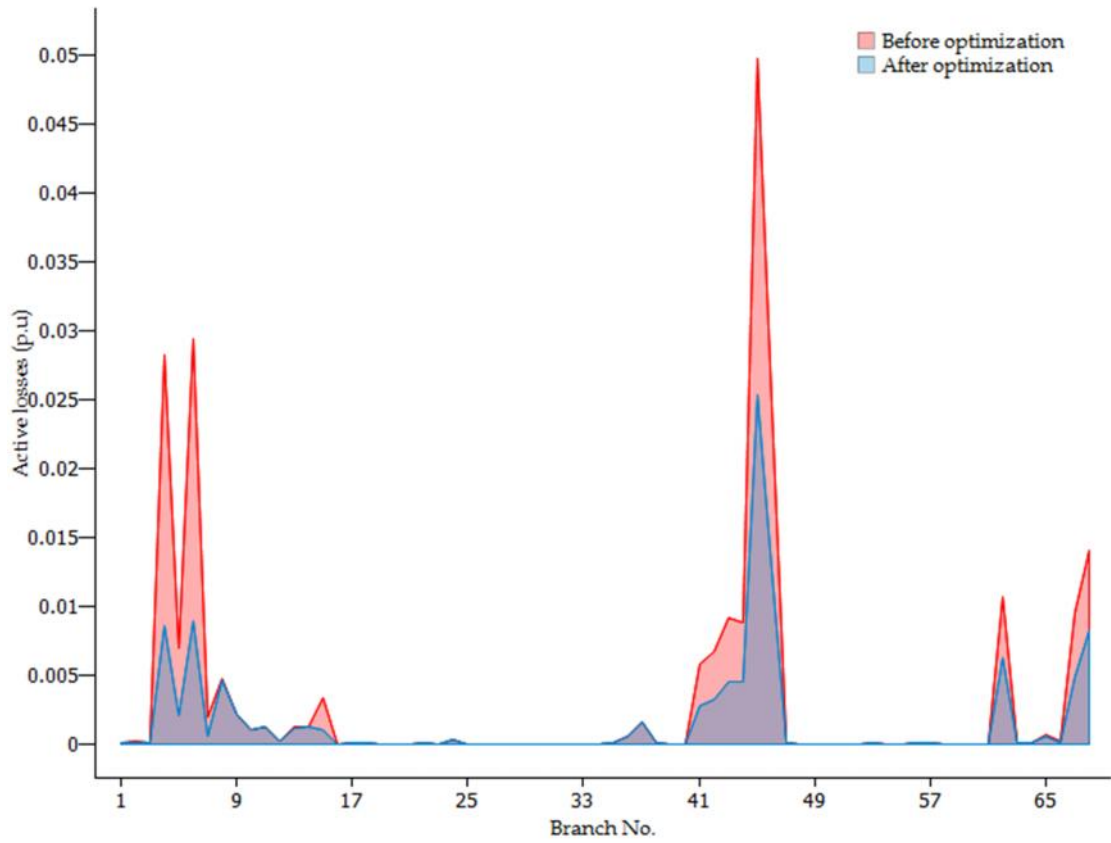


Figure 5.12 Pertes de puissance active au niveau de chaque branche du réseau IEEE 69-nœuds avant et après l'installation du DG

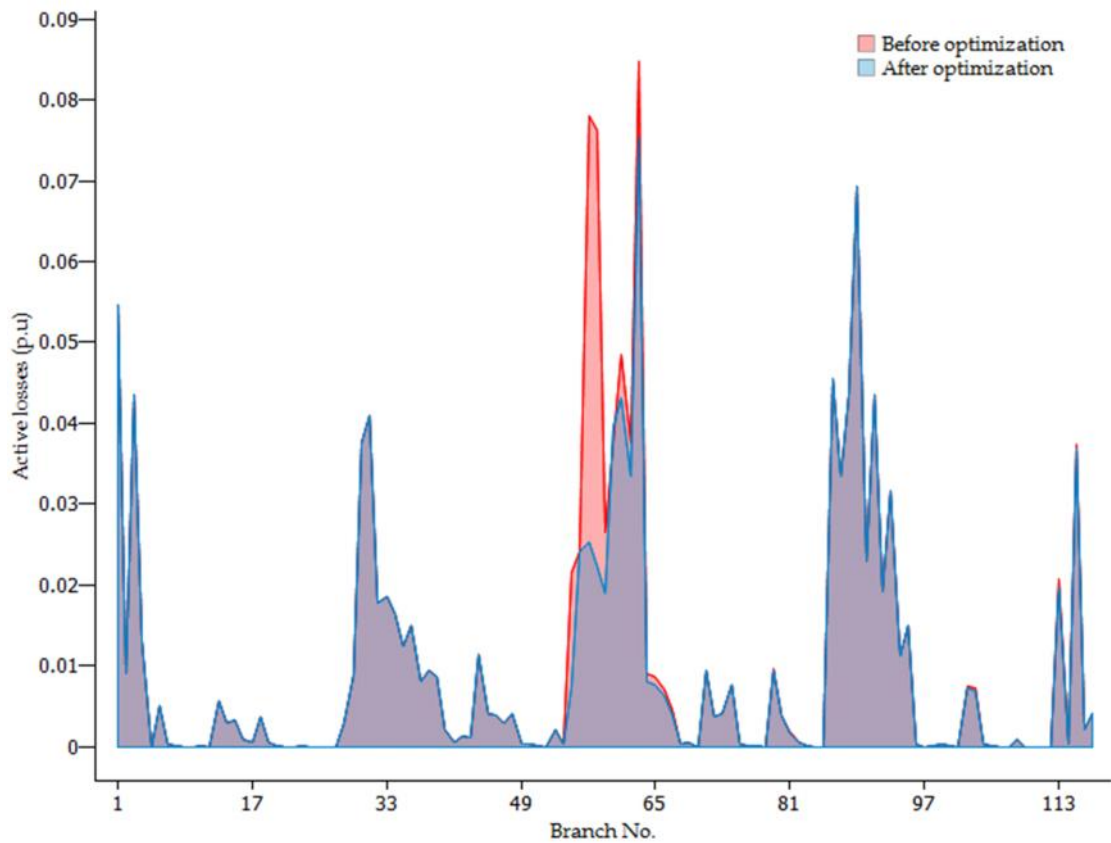


Figure 5.13 Pertes de puissance active au niveau de chaque branche du réseau IEEE 118-nœuds avant et après l'installation du DG

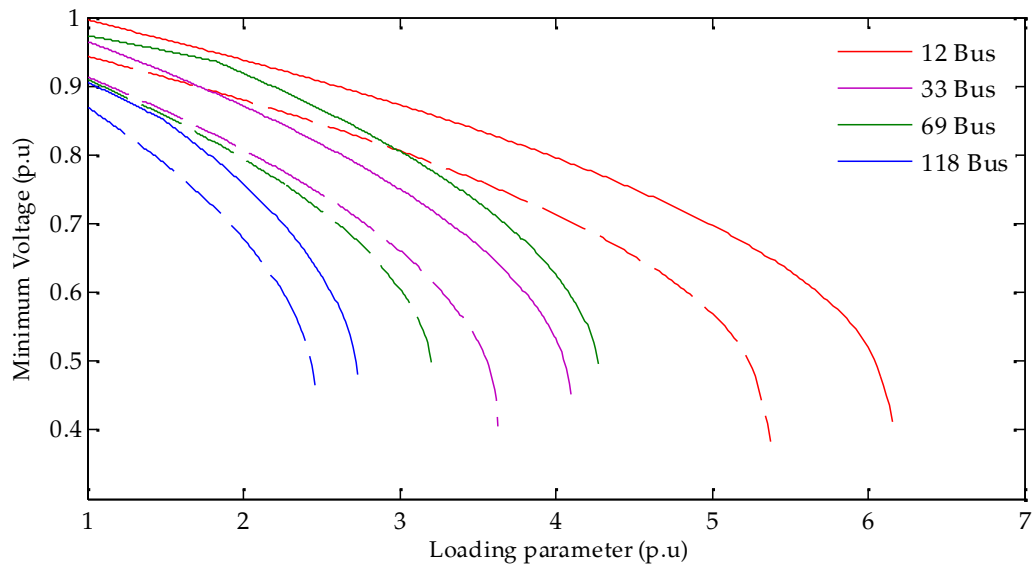


Figure 5.14 Capacité maximale de charge avant et après l'installation du DG

5.4 Reconfiguration du réseau de distribution en présence des DGs

Le problème de la reconfiguration optimale en présence de génération d'énergie dispersée en utilisant l'algorithme NSGA-II a été abordé aussi pour les mêmes objectifs, minimisation des pertes actifs et amélioration de la stabilité de tension. Il s'agit de déterminer une nouvelle topologie optimale du réseau par l'application de cet algorithme en trouvant les interrupteurs normalement ouverts à fermer et les branches munies des interrupteurs normalement fermés à ouvrir correspondant à la meilleure solution possible. L'opération se fait selon l'algorithme présenté dans le chapitre précédent. Les paramètres de simulation adoptés pour l'NSGA-II sont les mêmes utilisées la section précédente.

Les tableaux (5.11) et (5.12) représentent le résumé des résultats obtenus avant et après reconfiguration des réseaux test IEEE 33-nœuds [87], IEEE 69-nœuds [88] et IEEE 118-nœuds [89] par application de l'algorithme NSGA-II pour les deux cas, sans et avec insertion optimale des DGs. Les figures (5.15)-(5.17) représentent l'amélioration du profil de tension pour les cas étudiés : configuration initiale sans DGs et sans reconfiguration, configuration initiale avec insertion optimale des DGs, reconfiguration optimale sans DGs et enfin reconfiguration optimale en présence des DGs.

Aux termes de ces résultats, les remarques suivantes peuvent être annoncées :

1. D'après le tableau (5.11), la reconfiguration de tous les réseaux test a permis une minimisation des pertes actives et améliorer de la stabilité de tension par rapport aux configurations initiales de tous les réseaux test.

2. Avec la reconfiguration du réseau en présence des DGs, les performances du réseau sont davantage améliorées permettant d'avoir le minimum de pertes totales de puissance et la meilleure stabilité de tension.
3. D'après les figures (5.15)-(5.17), les niveaux de tension pour tous les réseaux test sont améliorés et placés dans une marge acceptable.
4. La reconfiguration du réseau en présence des DGs a permis d'améliorer d'avantage la capacité maximale de charge pour tous les réseaux test.

Les figures (5.18)-(5.23) donnent les nouvelles configurations optimales obtenues par l'application de l'algorithme NSGA-II en présence des DGs pour mes réseaux test où les lignes noires indiquent les branches en service ou fermées et les lignes rouges indiquent les branches hors service ou ouvertes. Les réseaux sont toujours radiaux et toutes les charges connectées sont alimentées.

Tableau 5.11 Résultats avant et après reconfiguration optimale en absence des DGs

Réseau test	Avant la reconfiguration					Après la reconfiguration optimale				
	Branches mis hors service	Real Power Loss (kW)	Minimum de tension (p.u)	min(SI)	Capacité Maximale de Charge	Branches mis hors service	Pertes de puissance (kW)	Minimum de tension (p.u)	min(SI)	Capacité Maximale de Charge
33-bus	(21-8) (9-15) (25-29) (12-22) (18-33)	202.66	0.91311	0.6952	3.63	(9-10) (14-15) (28-29) (7-8) (32-33)	139.94	0.94130	0.7848	4.24
69-bus	(27-65) (50-59) (13-20) (11-43) (15-46)	225.07	0.90916	0.6838	3.21	(61-62) (55-56) (13-20) (14-15) (10-11)	105.26	0.94278	0.7900	4.73
118-bus	(8-24) (17-27) (9-40) (46-27) (25-35) (99-77) (75-88) (110-118) (105-86) (108-83) (73-91) (54-43) (37-62) (96-58) (62-49)	1,29809	0.86880	0.5727	2.46	(42-43) (26-27) (39-40) (32-33) (22-23) (76-77) (74-75) (110-109) (107-108) (105-86) (71-72) (51-52) (48-49) (96-91) (58-59)	912.51	0.93111	0.7614	3.16

Tableau 5.12 Résultats avant et après reconfiguration optimale en présence des DGs

Réseau test	Avant la reconfiguration					Après la reconfiguration optimale				
	Branches mis hors service	Real Power Loss (kW)	Minimum de tension (p.u)	min(SI)	Capacité Maximale de Charge	Branches mis hors service	Pertes de puissance (kW)	Minimum de tension (p.u)	min(SI)	Capacité Maximale de Charge
33-bus	(21-8) (9-15) (25-29) (12-22) (18-33)	116.29	0.96450	0.8660	4.10	(9-10) (14-15) (25-29) (21-8) (32-33)	112.56	0.97330	0.8969	4.75
69-bus	(27-65) (50-59) (13-20) (11-43) (15-46)	108.88	0.97332	0.8975	4.27	(27-65) (55-56) (13-20) (14-15) (11-43)	68.79	0.98627	0.9462	4.84
118-bus	(8-24) (17-27) (9-40) (46-27) (25-35) (99-77) (75-88) (110-118) (105-86) (108-83) (73-91) (54-43) (37-62) (96-58) (62-49)	1,147.08	0.90529	0.6756	2.73	(42-43) (26-27) (39-40) (32-33) (21-22) (98-99) (74-75) (109-110) (108-83) (105-86) (71-72) (51-52) (48-49) (96-58) (60-61)	885.83	0.93229	0.75957	4.14

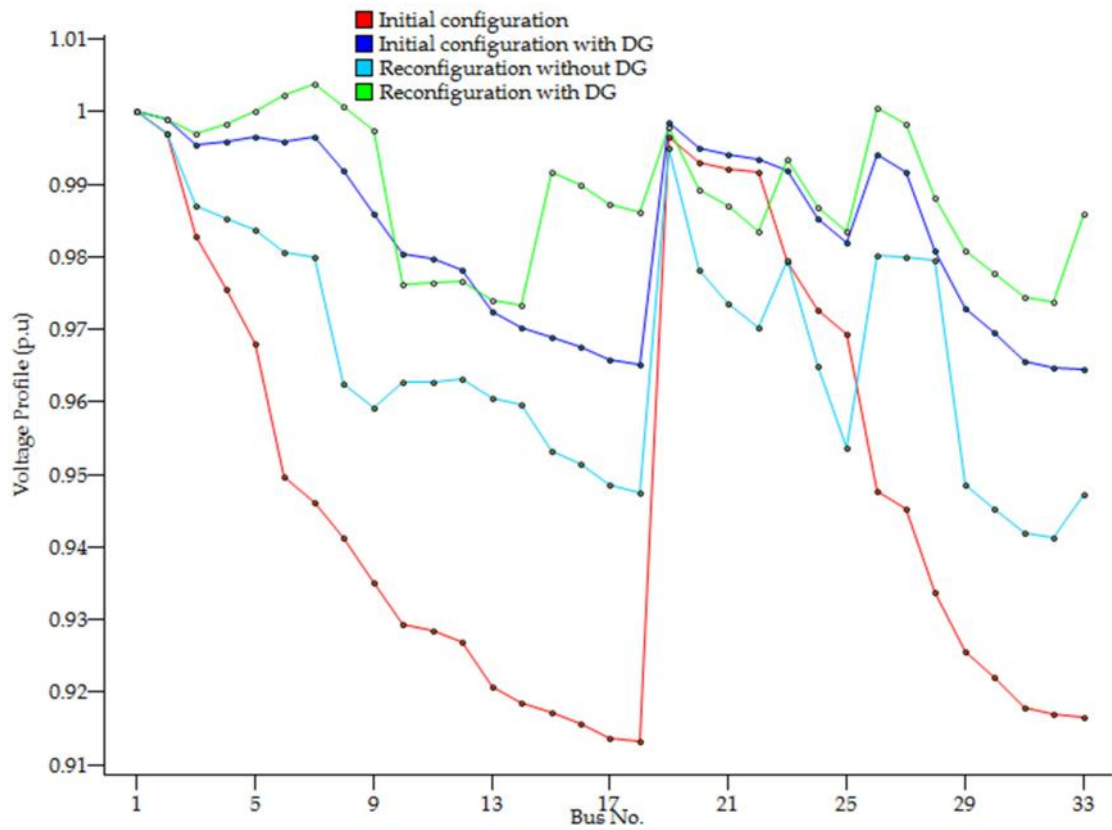


Figure 5.15 Amélioration du profil de tension par rapport au cas sans reconfiguration et sans DGs (Réseau IEEE 33-nœuds)

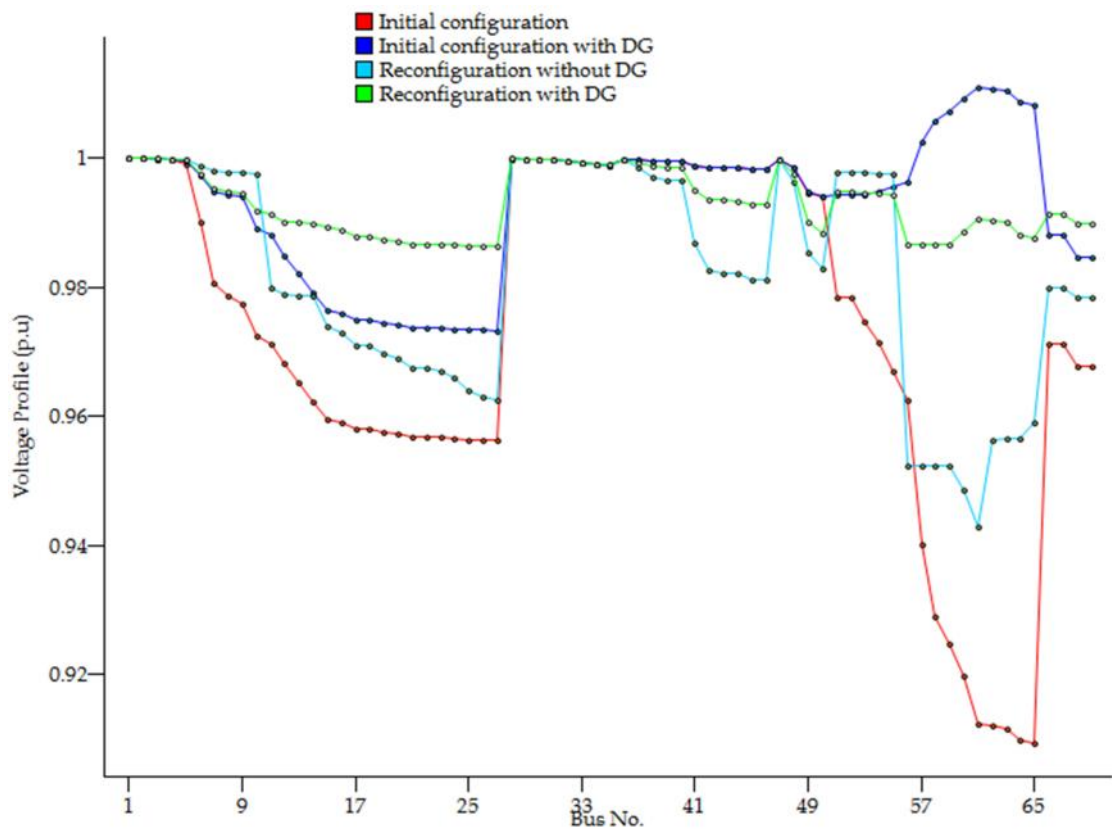


Figure 5.16 Amélioration du profil de tension par rapport au cas sans reconfiguration et sans DGs (Réseau IEEE 69-nœuds)

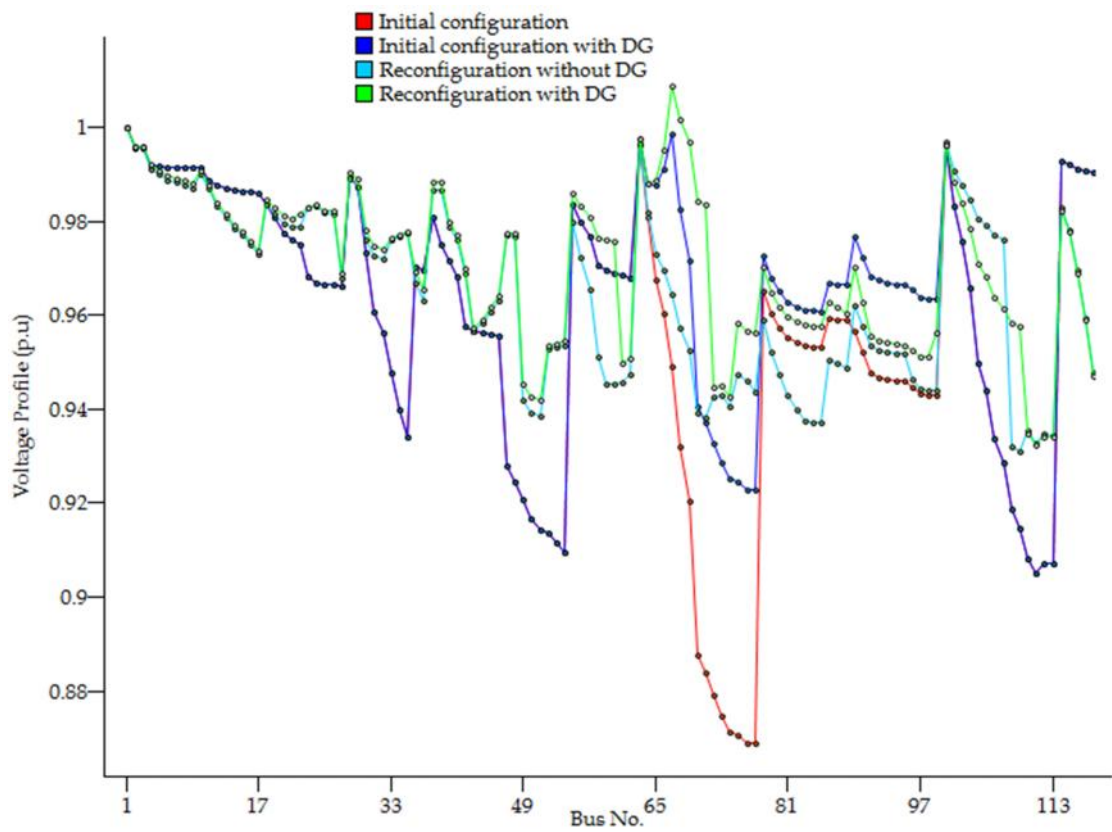


Figure 5.17 Amélioration du profil de tension par rapport au cas sans reconfiguration et sans DGs (Réseau IEEE 118-nœuds)

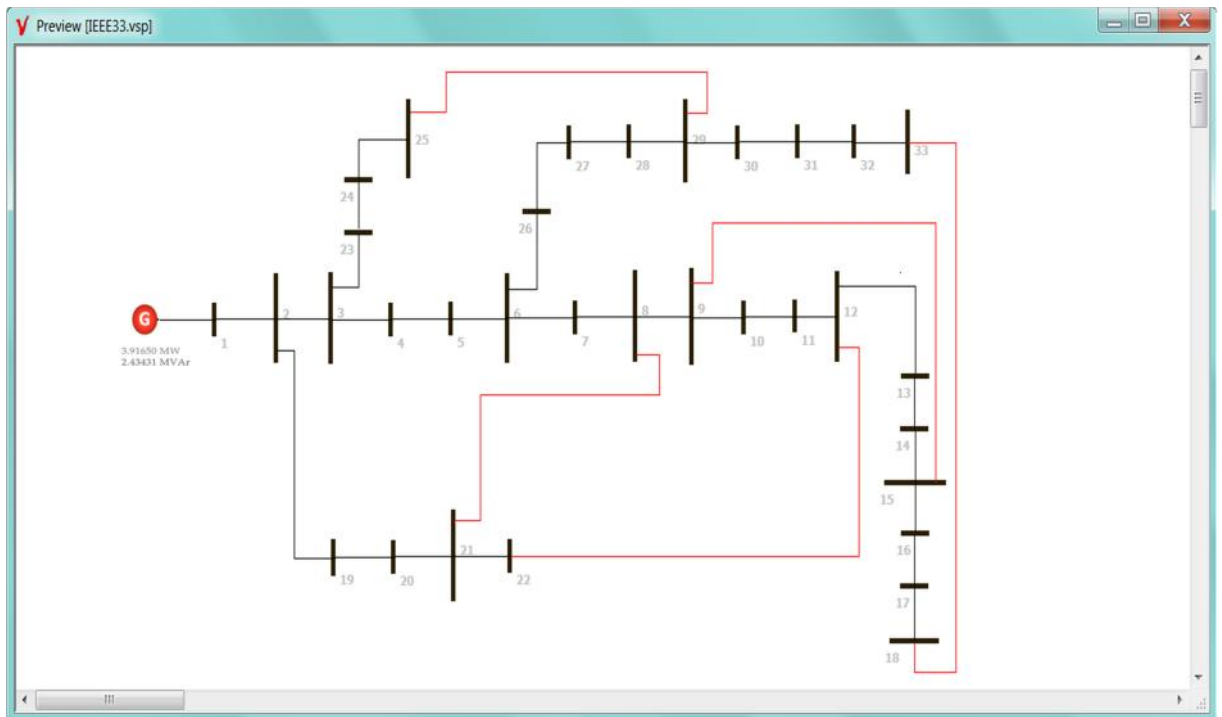


Figure 5.18 Réseau IEEE 33-nœuds avec la configuration initiale en absence du DG

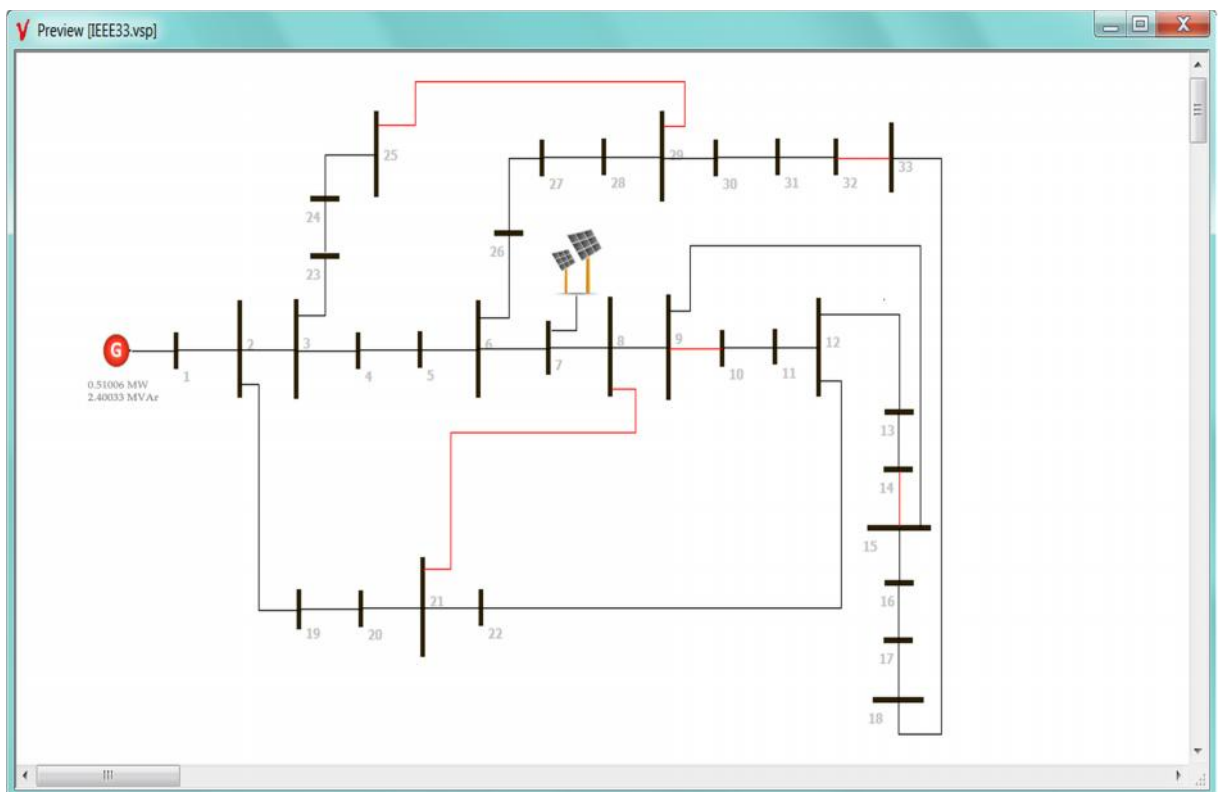


Figure 5.19 Réseau IEEE 33-nœuds avec la nouvelle configuration optimale en présence du DG

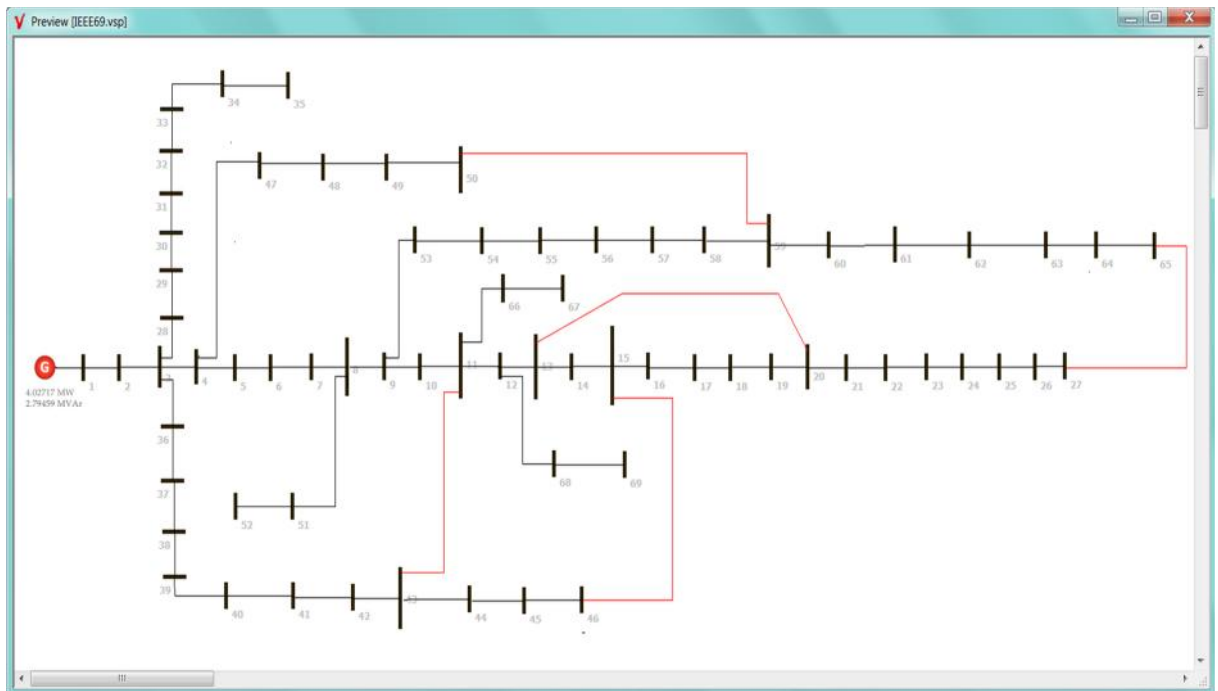


Figure 5.20 Réseau IEEE69-nœuds avec la configuration initiale en absence du DG

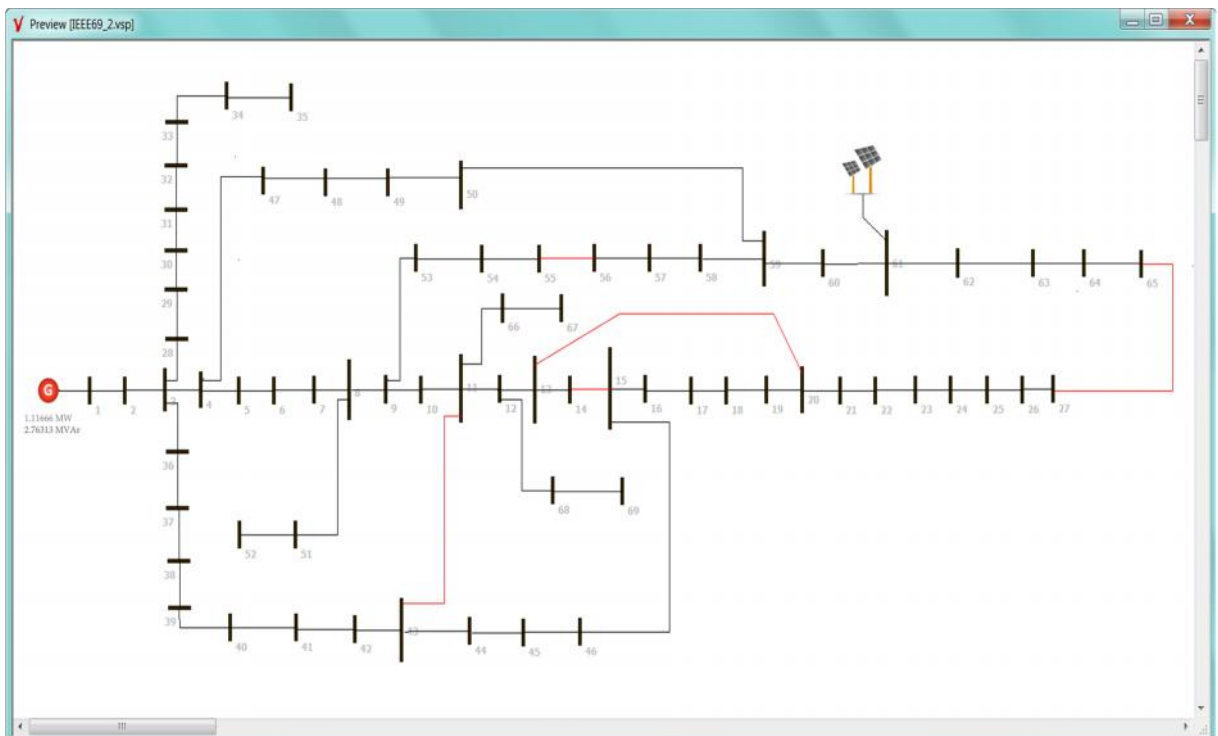


Figure 5.21 Réseau IEEE 69-nœuds avec la nouvelle configuration optimale en présence du

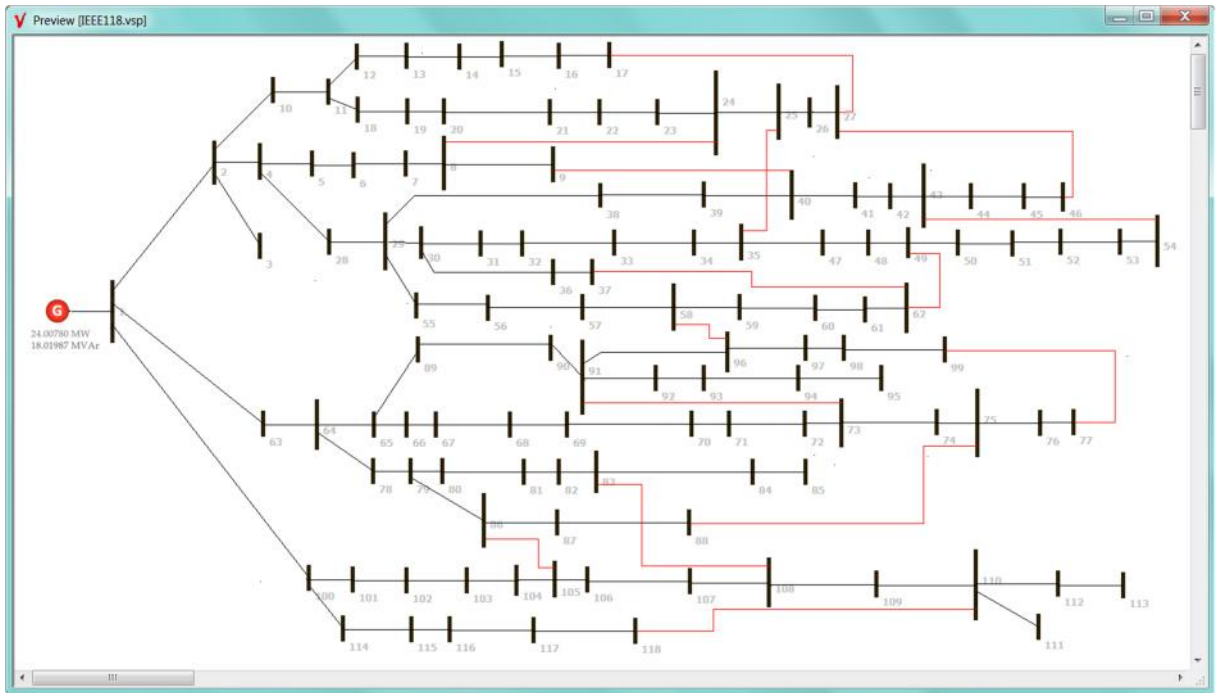


Figure 5.22 Réseau IEEE 118-nœuds avec la configuration initiale en absence du DG

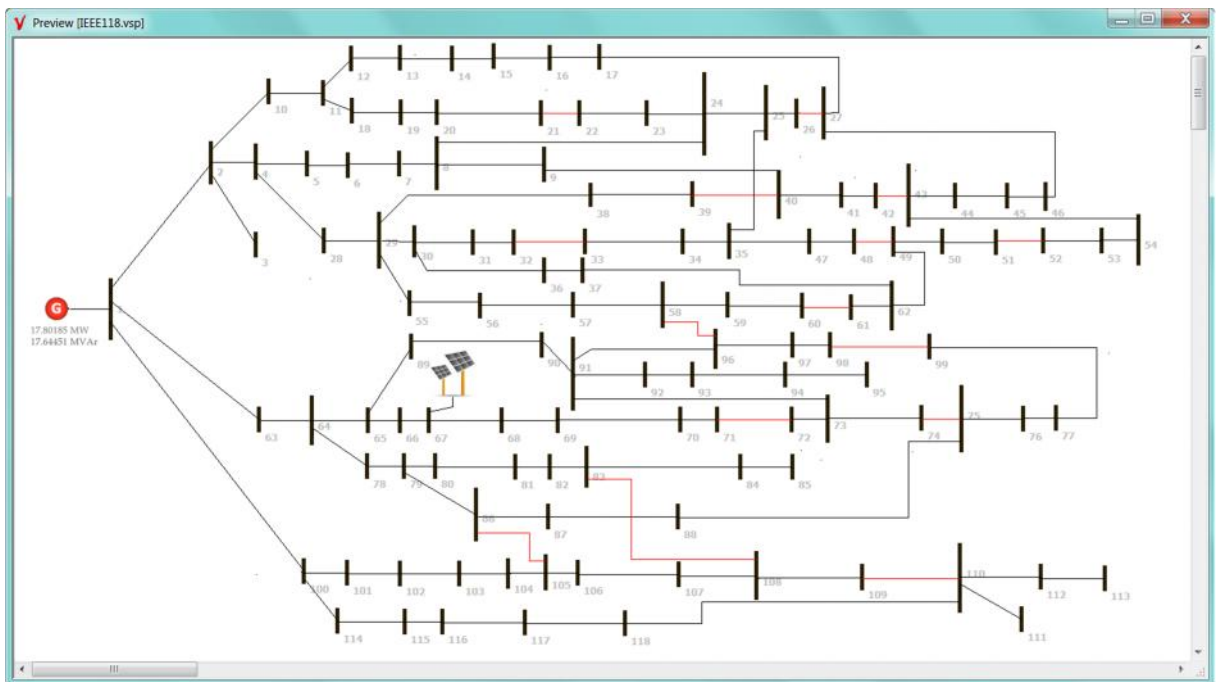


Figure 5.23 Réseau IEEE 118-nœuds avec la nouvelle configuration optimale en présence du DG

5.5 Conclusion

Dans ce chapitre, on a testé les méthodes d'optimisation présentées dans les chapitres 1 et 2 et qui se trouvent dans la bibliothèque d'optimisation orientée objet proposée dans le chapitre 3. Les résultats présentés dans ce chapitre montrent que la technique orientée objet donne une facilité de tester avec efficacité les différentes méthodes d'optimisation sur des réseaux tests de tailles différentes. Dans la première partie de ce chapitre, nous avons proposé une nouvelle technique se basant sur l'hybridation de deux métaheuristiques de voisinage la recherche taboue et le recuit simulé pour résoudre le problème de l'UC sans et avec les contraintes d'EPO. Cette hybridation nous a permis de profiter de l'avantage du RS : l'acceptation des 'mauvaises' solutions ce qui permet alors d'explorer une plus grande partie de l'espace de recherche et l'avantage de la RT qui consiste à interdire de revenir sur les dernières positions explorées. Dans la deuxième partie de ce chapitre, nous avons utilisé la méthode d'optimisation multi-objective NSGA-II pour déterminer la taille et de l'emplacement de la production décentralisée dans les réseaux électriques de distribution, l'objectif est de minimiser les pertes actives de puissance et améliorer la stabilité de tension. Ensuite nous avons appliqué une méthode hybride entre le NSGA-II et la théorie des graphes pour résoudre le problème de la reconfiguration optimal sans et avec l'emplacement optimal des DGs avec les mêmes objectives. Nous avons remarqué qu'avec la reconfiguration du réseau en présence des DGs, les performances du réseau sont davantage améliorées permettant d'avoir le minimum de pertes totales de puissance et la meilleure stabilité de tension et que les profils de tension et les capacités maximales des charges sont considérablement amélioré pour tous les réseaux test étudiés.

Conclusion Générale

Nous avons abordé dans cette thèse, le développement d'une plate-forme logicielle pour la visualisation, la simulation, et l'optimisation des réseaux électriques qui est en particulier flexible, extensible et facile à maintenir en utilisant les avantages de la méthodologie orienté-objet où le réseau électrique est modélisé comme un ensemble d'objets liés et structurés dans une hiérarchie de classes.

Deux grandes structures de classes sont développées : la structure des éléments physiques du réseau électrique (jeux de barres, lignes de transmission, générateurs,..) et la structure des méthodes d'analyse (applications). Les classes créées sont simples et pourront être utilisées et réutilisées ce qui va permettre de se concentrer sur les spécificités de chaque application. Trois applications intégrées dans la version actuelle de notre logiciel sont présentées dans cette thèse, il s'agit de l'Engagement de Turbines (Unit Commitment), l'optimisation de l'emplacement et de la taille des générateurs dispersés (DGs) dans les réseaux de distribution, et la reconfiguration optimale des réseaux en présence des DGs.

La construction de l'interface usager graphique GUI est une partie importante de la plate-forme logicielle développée. Cependant, dans la description de ce travail et ses classes, on a mis l'accent sur tout ce qui est électrique et on a survolé tout ce qui est graphique (plus difficile) ou purement informatique pour mieux présenter la TOO et ses avantages dans le domaine Electrotechnique en général.

Les résultats d'optimisation obtenues sont améliorés considérablement avec la programmation orientée objet pour toutes les fonctions objectif utilisées. Cela est justifié par les grands avantages de la programmation orientée objet surtout la gestion optimale de la mémoire, utilisation optimale des ressources du matériel et vitesse d'exécution, aptitude à protéger ses composantes contre des modifications non autorisées et la protection des données.

Perspectives et Suggestions

Les travaux accomplis jusqu'à présent nous ont permis de construire un ensemble cohérent constitué d'un modèle objet du réseau électrique (et de ses composants) et de son interaction avec les différentes applications intégrées constituant une première version d'une plateforme de simulateur de réseau électrique. Il reste cependant de nombreux prolongements qui se situent sur plusieurs plans. Nous suggérons pour les travaux futurs :

- Amélioration de la convivialité de l'interface graphique, par la présentation des résultats de simulation directement sur le diagramme unifilaire ;
- Extension de la bibliothèque des composants (FACTS...);
- Intégration des nouvelles applications et des nouvelles méthodes d'optimisation;

Bibliographie

Bibliographie

- [1] M. Tamali, *Modélisation et Simulation des réseaux électriques: Développement du logiciel NMSS*: Editions Edilivre, 2013.
- [2] S. A. Soman, S. Khaparde, and S. Pandit, *Computational methods for large sparse power systems analysis: an object oriented approach*: Springer Science & Business Media, 2012.
- [3] A. MESSAOUDI, "Application de la Programmation Orientée Objet et des Méthodes Heuristiques à l'Optimisation de l'Écoulement de Puissance," Doctorat En Sciences, Département d'Electrotechnique, Université Hadj Lakhdar, Batna, 2013.
- [4] T. Bouktir, "Application de la Programmation Orientée Objet à l'Optimisation de l'Écoulement de Puissances," Doctorat En Sciences, Département d'Electrotechnique, Université Hadj Lakhdar, Batna, 2004.
- [5] A. Gherbi, "Développement d'un Programme Orienté Objet de la Stabilité Transitoire des Réseaux Electriques," Département d'Electrotechnique, Université Ferhat Abbas, Setif, Doctorat En Sciences, 2006.
- [6] A. TOLBA, "Coordination Orientee Objet de la Protection des Reseaux Electriques," Doctorat En Sciences, Département d'Electrotechnique, Université Hadj Lakhdar, Batna, 2010.
- [7] L. BENFARHI, "Developpement d'un Outil Graphique pour la Simulation des Reseaux Electriques par la Programmation Orientee Objets," Doctorat En Sciences, Département d'Electrotechnique, Université Hadj Lakhdar, Batna, 2006.
- [8] J. Rumbaugh, I. Jacobson, and G. Booch, *Unified Modeling Language Reference Manual, The*: Pearson Higher Education, 2004.
- [9] K. L. JAMES, *SOFTWARE ENGINEERING*: PHI Learning, 2008.
- [10] M. Saguan, "L'Analyse économique des architectures de marché électrique. L'application au market design du temps réel," Université Paris Sud-Paris XI, 2007.
- [11] B. Wollenberg and A. Wood, "Power generation, operation and control," *John Wiley&Sons, Inc*, pp. 264-327, 1996.
- [12] H. W. Dommel and W. F. Tinney, "Optimal power flow solutions," *power apparatus and systems, IEEE transactions on*, pp. 1866-1876, 1968.
- [13] D. Sun, B. Ashley, B. Brewer, A. Hughes, and W. F. Tinney, "Optimal power flow by Newton approach," *power apparatus and systems, iee transactions on*, pp. 2864-2880, 1984.
- [14] J. F. Bard, "Short-term scheduling of thermal-electric generators using Lagrangian relaxation," *Operations Research*, vol. 36, pp. 756-766, 1988.
- [15] C.-P. Cheng, C.-W. Liu, and C.-C. Liu, "Unit commitment by Lagrangian relaxation and genetic algorithms," *Power Systems, IEEE Transactions on*, vol. 15, pp. 707-714, 2000.
- [16] I. G. Damousis, A. G. Bakirtzis, and P. S. Dokopoulos, "A solution to the unit-commitment problem using integer-coded genetic algorithm," *Power Systems, IEEE Transactions on*, vol. 19, pp. 1165-1172, 2004.
- [17] C. Dang and M. Li, "A floating-point genetic algorithm for solving the unit commitment problem," *European Journal of Operational Research*, vol. 181, pp. 1370-1395, 2007.
- [18] K. Juste, H. Kita, E. Tanaka, and J. Hasegawa, "An evolutionary programming solution to the unit commitment problem," *Power Systems, IEEE Transactions on*, vol. 14, pp. 1452-1459, 1999.

- [19] T. Senjyu, H. Yamashiro, K. Uezato, and T. Funabashi, "A unit commitment problem by using genetic algorithm based on unit characteristic classification," in *Power Engineering Society Winter Meeting, 2002. IEEE, 2002*, pp. 58-63.
- [20] S. P. Simon, N. P. Padhy, and R. Anand, "An ant colony system approach for unit commitment problem," *International Journal of Electrical Power & Energy Systems*, vol. 28, pp. 315-323, 2006.
- [21] L. Sun, Y. Zhang, and C. Jiang, "A matrix real-coded genetic algorithm to the unit commitment problem," *Electric Power Systems Research*, vol. 76, pp. 716-728, 2006.
- [22] J. Valenzuela and A. E. Smith, "A seeded memetic algorithm for large unit commitment problems," *Journal of Heuristics*, vol. 8, pp. 173-195, 2002.
- [23] A. Viana, J. P. de Sousa, and M. Matos, "Using GRASP to solve the unit commitment problem," *Annals of Operations Research*, vol. 120, pp. 117-132, 2003.
- [24] B. Zhao, C. Guo, B. Bai, and Y. Cao, "An improved particle swarm optimization algorithm for unit commitment," *International Journal of Electrical Power & Energy Systems*, vol. 28, pp. 482-490, 2006.
- [25] X. Guan, P. B. Luh, H. Yan, and J. Amalfi, "An optimization-based method for unit commitment," *International Journal of Electrical Power & Energy Systems*, vol. 14, pp. 9-17, 1992.
- [26] S. A. Kazarlis, A. Bakirtzis, and V. Petridis, "A genetic algorithm solution to the unit commitment problem," *Power Systems, IEEE Transactions on*, vol. 11, pp. 83-92, 1996.
- [27] A. Turgeon, "Optimal scheduling of thermal generating units," *Automatic Control, IEEE Transactions on*, vol. 23, pp. 1000-1005, 1978.
- [28] J. Dréo, A. Pétrowski, É. D. Taillard, and P. Siarry, "Métaheuristiques pour l'optimisation difficile," 2003.
- [29] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *science*, vol. 220, pp. 671-680, 1983.
- [30] F. Glover, M. Laguna, E. Taillard, and D. De Werra, *Tabu search*: Baltzer Basel, 1993.
- [31] G. Purushothama and L. Jenkins, "Simulated annealing with local search-a hybrid algorithm for unit commitment," *Power Systems, IEEE Transactions on*, vol. 18, pp. 273-278, 2003.
- [32] W. F. Tinney and C. E. Hart, "Power flow solution by Newton's method," *Power Apparatus and Systems, IEEE Transactions on*, pp. 1449-1460, 1967.
- [33] M. Haque, "Novel decoupled load flow method," in *IEE Proceedings C (Generation, Transmission and Distribution)*, 1993, pp. 199-205.
- [34] D. Raji i and A. Bose, "A modification to the fast decoupled power flow for networks with high R/X ratios," *Power Systems, IEEE Transactions on*, vol. 3, pp. 743-746, 1988.
- [35] B. Stott and O. Alsac, "Fast decoupled load flow," *power apparatus and systems, iee transactions on*, pp. 859-869, 1974.
- [36] R. A. Van Amerongen, "A general-purpose version of the fast decoupled load flow," *Power Systems, IEEE Transactions on*, vol. 4, pp. 760-770, 1989.
- [37] P. R. Rao, S. Sivanagaraju, and P. Prasad, "Backward Forward Propagation Power Flow Method for Radial Distribution Systems," *Journal of Electrical and Control Engineering*, vol. 2, pp. 20-23, 2012.
- [38] V. Jayasankar, N. Kamaraj, and N. Vanaja, "Estimation of voltage stability index for power system employing artificial neural network technique and TCSC placement," *Neurocomputing*, vol. 73, pp. 3005-3011, 2010.
- [39] P. Kessel and H. Glavitsch, "Estimating the voltage stability of a power system," *Power Delivery, IEEE Transactions on*, vol. 1, pp. 346-354, 1986.

- [40] B. D. Thukaram and K. Parthasarathy, "Optimal reactive power dispatch algorithm for voltage stability improvement," *International Journal of Electrical Power & Energy Systems*, vol. 18, pp. 461-468, 1996.
- [41] C.-F. Yang, G. G. Lai, C.-H. Lee, C.-T. Su, and G. W. Chang, "Optimal setting of reactive compensation devices with an improved voltage stability index for voltage stability enhancement," *International Journal of Electrical Power & Energy Systems*, vol. 37, pp. 50-57, 2012.
- [42] D. Devaraj and J. P. Roselyn, "Genetic algorithm based reactive power dispatch for voltage stability improvement," *International Journal of Electrical Power & Energy Systems*, vol. 32, pp. 1151-1156, 2010.
- [43] M. Chakravorty and D. Das, "Voltage stability analysis of radial distribution networks," *International Journal of Electrical Power & Energy Systems*, vol. 23, pp. 129-135, 2001.
- [44] T. Gözel and M. H. Hocaoglu, "An analytical method for the sizing and siting of distributed generators in radial systems," *Electric Power Systems Research*, vol. 79, pp. 912-918, 2009.
- [45] M. Aman, G. Jasmon, H. Mokhlis, and A. Bakar, "Optimal placement and sizing of a DG based on a new power stability index and line losses," *International Journal of Electrical Power & Energy Systems*, vol. 43, pp. 1296-1304, 2012.
- [46] R. Singh and S. Goswami, "Optimum siting and sizing of distributed generations in radial and networked systems," *Electric Power Components and Systems*, vol. 37, pp. 127-145, 2009.
- [47] Z. Moravej and A. Akhlaghi, "A novel approach based on cuckoo search for DG allocation in distribution network," *International Journal of Electrical Power & Energy Systems*, vol. 44, pp. 672-679, 2013.
- [48] W. Tan, M. Y. Hassan, M. Majid, and H. Rahman, "Allocation and sizing of DG using cuckoo search algorithm," in *Power and Energy (PECon), 2012 IEEE International Conference on*, 2012, pp. 133-138.
- [49] M. P. Lalitha, V. V. Reddy, and V. Usha, "Optimal DG placement for minimum real power loss in radial distribution systems using PSO," *Journal of Theoretical and Applied Information Technology*, vol. 13, pp. 107-116, 2010.
- [50] M. Aman, G. Jasmon, A. Bakar, and H. Mokhlis, "A new approach for optimum DG placement and sizing based on voltage stability maximization and minimization of power losses," *Energy Conversion and Management*, vol. 70, pp. 202-210, 2013.
- [51] M. Moradi and M. Abedini, "A combination of genetic algorithm and particle swarm optimization for optimal DG location and sizing in distribution systems," *International Journal of Electrical Power & Energy Systems*, vol. 34, pp. 66-74, 2012.
- [52] J. A. M. García and A. J. G. Mena, "Optimal distributed generation location and size using a modified teaching-learning based optimization algorithm," *International journal of electrical power & energy systems*, vol. 50, pp. 65-75, 2013.
- [53] F. S. Abu-Mouti and M. El-Hawary, "Optimal distributed generation allocation and sizing in distribution systems via artificial bee colony algorithm," *Power Delivery, IEEE Transactions on*, vol. 26, pp. 2090-2101, 2011.
- [54] L. Arya, A. Koshti, and S. Choube, "Distributed generation planning using differential evolution accounting voltage stability consideration," *International Journal of Electrical Power & Energy Systems*, vol. 42, pp. 196-207, 2012.
- [55] H. Hejazi, M. Hejazi, G. B. Gharehpetian, and M. Abedi, "Distributed generation site and size allocation through a techno economical multi-objective Differential Evolution Algorithm," in *Power and Energy (PECon), 2010 IEEE International Conference on*, 2010, pp. 874-879.

- [56] W.-S. Tan, M. Y. Hassan, M. S. Majid, and H. A. Rahman, "Optimal distributed renewable generation planning: A review of different approaches," *Renewable and Sustainable Energy Reviews*, vol. 18, pp. 626-645, 2013.
- [57] N. KETFI, "Contribution à la gestion des réseaux de distribution en présence de génération d'énergie dispersée," Doctorat en sciences, Département d'Électrotechnique, Batna, 2014.
- [58] R. Mallipeddi, S. Jeyadevi, P. N. Suganthan, and S. Baskar, "Efficient constraint handling for optimal reactive power dispatch problems," *Swarm and Evolutionary Computation*, vol. 5, pp. 28-36, 2012.
- [59] M. Kashem, V. Ganapathy, and G. Jasmon, "Network reconfiguration for load balancing in distribution networks," *IEE Proceedings-Generation, Transmission and Distribution*, vol. 146, pp. 563-567, 1999.
- [60] A. Y. Abdelaziz, F. Mohamed, S. Mekhamer, and M. Badr, "Distribution system reconfiguration using a modified Tabu Search algorithm," *Electric Power Systems Research*, vol. 80, pp. 943-953, 2010.
- [61] K. K. Kumar, N. V. Ramana, and S. Kamakshaiah, "Design and Development of Modified PSO Algorithm for Network Reconfiguration," *International Review of Electrical Engineering (IREE)*, vol. 8, pp. 1586-1593, 2013.
- [62] W.-C. Wu and M.-S. Tsai, "Application of enhanced integer coded particle swarm optimization for distribution system feeder reconfiguration," *Power Systems, IEEE Transactions on*, vol. 26, pp. 1591-1599, 2011.
- [63] N. Gupta, A. Swarnkar, and K. Niazi, "Reconfiguration of distribution systems for real power loss minimization using adaptive particle swarm optimization," *Electric Power Components and Systems*, vol. 39, pp. 317-330, 2011.
- [64] S. Sivanagaraju, Y. Srikanth, and E. J. Babu, "An efficient genetic algorithm for loss minimum distribution system reconfiguration," *Electric Power Components and Systems*, vol. 34, pp. 249-258, 2006.
- [65] N. T. Linh and N. Q. Anh, "Application artificial bee colony algorithm (ABC) for reconfiguring distribution network," in *Computer Modeling and Simulation, 2010. ICCMS'10. Second International Conference on*, 2010, pp. 102-106.
- [66] D. Pal, S. Kumar, B. Tudu, K. Mandal, and N. Chakraborty, "Efficient and Automatic Reconfiguration and Service Restoration in Radial Distribution System Using Differential Evolution," in *Proceedings of the International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA)*, 2013, pp. 365-372.
- [67] D. S. Rani, N. Subrahmanyam, and M. Sydulu, "Improved music based harmony search algorithm for optimal network reconfiguration," in *India Conference (INDICON), 2012 Annual IEEE*, 2012, pp. 1030-1035.
- [68] M. Sedighzadeh, M. Esmaili, and M. Esmaili, "Application of the hybrid Big Bang-Big Crunch algorithm to optimal reconfiguration and distributed generation power allocation in distribution systems," *Energy*, vol. 76, pp. 920-930, 2014.
- [69] Y. M. Shuaib, M. S. Kalavathi, and C. C. Asir Rajan, "Optimal reconfiguration in radial distribution system using gravitational search algorithm," *Electric Power Components and Systems*, vol. 42, pp. 703-715, 2014.
- [70] L. W. de Oliveira, E. J. de Oliveira, F. V. Gomes, I. C. Silva, A. L. Marcato, and P. V. Resende, "Artificial immune systems applied to the reconfiguration of electrical power distribution networks for energy loss minimization," *International Journal of Electrical Power & Energy Systems*, vol. 56, pp. 64-74, 2014.
- [71] A. Kouzou, R. Mohammadi, and A. Hellal, "An efficient biogeography-based optimization algorithm for smart radial distribution power system reconfiguration," in *Smart Grid and Renewable Energy (SGRE), 2015 First Workshop on*, 2015, pp. 1-7.
- [72] N. Deo, *Graph theory with applications to engineering and computer science*: PHI Learning Pvt. Ltd., 2004.

- [73] U. Agarwal and U. P. Singh, *Graph Theory*: Laxmi Publications Pvt Limited, 2009.
- [74] K. Deb, *Multi-objective optimization using evolutionary algorithms* vol. 16: John Wiley & Sons, 2001.
- [75] M. Sakawa, H. Yano, and T. Yumine, "An interactive fuzzy satisficing method for multiobjective linear-programming problems and its application," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 17, pp. 654-661, 1987.
- [76] M. Takahashi and H. Kita, "A crossover operator using independent component analysis for real-coded genetic algorithms," in *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, 2001, pp. 643-649.
- [77] G. Booch, *Analyse et conception orientees objets*: Addison-Wesley, 1994.
- [78] E. C. Weber, J. N. Ford, and C. R. Weber, *Developing with Delphi: Object-oriented Techniques*: Prentice Hall PTR, 1996.
- [79] R. D. Mohammadi, A. Hellal, and S. Arif, "An Object-Oriented Programming Approach to Security-Constrained Unit Commitment Problem," *International Review of Electrical Engineering (IREE)*, vol. 9, pp. 1180-1189, 2014.
- [80] T. Bouktir and L. Slimani, "Object-oriented economic power dispatch of electrical power system with minimum pollution using a genetic algorithm," *J. Electrical Systems*, vol. 1, pp. 19-34, 2005.
- [81] T. Bouktir and M. Belkacemi, "Object-oriented optimal power flow," *Electric Power Components and Systems*, vol. 31, pp. 525-534, 2003.
- [82] M. Selvan and K. Swarup, "Modeling and analysis of unbalanced distribution system using object-oriented methodology," *Electric power systems research*, vol. 76, pp. 968-979, 2006.
- [83] A. T. L. B. M. Belkacemi, "A Template Sparse Library for Power Systems," *Journal of ELECTRICAL ENGINEERING*, vol. 57, pp. 161-166, 2006.
- [84] W. Poommalee and P. Damrongkulkamjorn, "Unit commitment considering security-constrained optimal power flow by lagrangian relaxation with genetic algorithm," in *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2008. ECTI-CON 2008. 5th International Conference on*, 2008, pp. 893-896.
- [85] L. Slimani, "Contribution à l'application de l'optimisation par des méthodes métaheuristiques à l'écoulement de puissance optimal dans un environnement de l'électricité dérégulé," Doctorat En Sciences, Département d'Electrotechnique, Université Hadj Lakhdar, Batna, 2009.
- [86] D. Das, H. Nagi, and D. Kothari, "Novel method for solving radial distribution networks," *IEE Proceedings-Generation, Transmission and Distribution*, vol. 141, pp. 291-298, 1994.
- [87] M. Jamil and S. Kirmani, "Optimal allocation of SPV based DG system for loss reduction and voltage improvement in radial distribution systems using approximate reasoning," in *Power Electronics (IICPE), 2012 IEEE 5th India International Conference on*, 2012, pp. 1-5.
- [88] M. E. Baran and F. F. Wu, "Optimal sizing of capacitors placed on a radial distribution system," *Power Delivery, IEEE Transactions on*, vol. 4, pp. 735-743, 1989.
- [89] D. Zhang, Z. Fu, and L. Zhang, "An improved TS algorithm for loss-minimum reconfiguration in large-scale distribution systems," *Electric Power Systems Research*, vol. 77, pp. 685-694, 2007.
- [90] K. Divya and P. N. Rao, "Models for wind turbine generating systems and their application in load flow studies," *Electric Power Systems Research*, vol. 76, pp. 844-856, 2006.
- [91] P. Kayal and C. Chanda, "Placement of wind and solar based DGs in distribution system for power loss minimization and voltage stability improvement," *International Journal of Electrical Power & Energy Systems*, vol. 53, pp. 795-809, 2013.

Annexes

Annexe A

Réseau test Sonelgaz 59-Bus

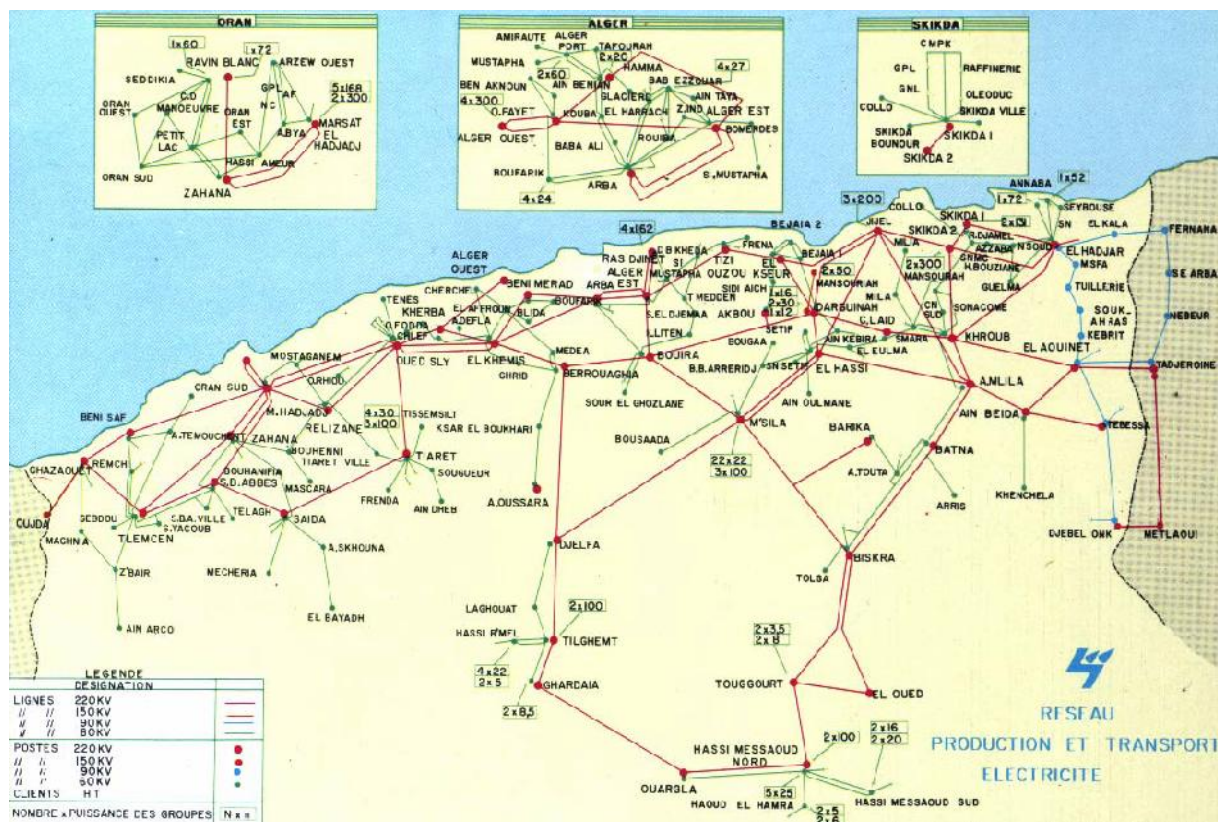


Figure A. 1 Schéma unifilaire du réseau de production et transport algérien (1997)

Tableau A. 1 Paramètres des générateurs du réseau 59 jeux de barres de Sonelgaz

Bus Number	Pmin [MW]	Pmax [MW]	Qmin [Mvar]	Qmax [Mvar]	a [\$ /hr]	B [\$ /MWhr]	c [\$ /MW ² hr]
1	8	72	-10	15	0	1.50	0.0085
2	10	70	-35	45	0	2.50	0.0170
3	30	510	-35	55	0	1.50	0.0085
4	20	400	-60	90	0	1.50	0.0085
13	15	150	-35	48	0	2.50	0.0170
27	10	100	-20	35	0	2.50	0.0170
37	10	100	-20	35	0	2.00	0.0030
41	15	140	-35	45	0	2.00	0.0030
42	18	175	-35	55	0	2.00	0.0030
53	30	450	-100	160	0	1.50	0.0085

Tableau A. 2 Données des jeux de barres du réseau 59 jeux de barres de Sonelgaz

Bus	Type	V	Angle	P _d (MW)	Q _d (Mvar)	P _g (MW)	Q _g (Mvar)	Q _{min} (Mvar)	Q _{max} (Mvar)	Mvar
1	3	1.06	0	0	0	0	0	-10	15	0
2	2	1.04	0	24.2	11	70	0	-35	45	0

3	2	1.05	0	0	0	70	0	-35	55	0
4	2	1.0283	0	68.5	31.2	115	0	-60	90	0
5	1	1	0	22.2	10.2	0	0	0	0	0
6	1	1	0	0	0	0	0	0	0	0
7	1	1	0	6	2.7	0	0	0	0	0
8	1	1	0	3.9	1.8	0	0	0	0	0
9	1	1	0	28.4	12.9	0	0	0	0	0
10	1	1	0	18	8.2	0	0	0	0	0
11	1	1	0	25	11.4	0	0	0	0	0
12	1	1	0	0	0	0	0	0	0	0
13	2	1	0	0	0	0	0	-35	48	0
14	1	1	0	22.5	10.3	0	0	0	0	0
15	1	1	0	19.4	8.8	0	0	0	0	0
16	1	1	0	0	0	0	0	0	0	0
17	1	1	0	6.4	2.9	0	0	0	0	0
18	1	1	0	0	0	0	0	0	0	0
19	1	1	0	0	0	0	0	0	0	0
20	1	1	0	52.9	24.1	0	0	0	0	0
21	1	1	0	0	0	0	0	0	0	0
22	1	1	0	0	0	0	0	0	0	0
23	1	1	0	56.7	25.8	0	0	0	0	0
24	1	1	0	21.4	9.8	0	0	0	0	0
25	1	1	0	0	0	0	0	0	0	0
26	1	1	0	19.6	8.9	0	0	0	0	0
27	2	1.0266	0	23.5	10.8	40	0	-20	35	0
28	1	1	0	7.8	3.5	0	0	0	0	0
29	1	1	0	5.9	2.7	0	0	0	0	0
30	1	1	0	0	0	0	0	0	0	0
31	1	1	0	0	0	0	0	0	0	0
32	1	1	0	0	0	0	0	0	0	0
33	1	1	0	24.7	11.3	0	0	0	0	0
34	1	1	0	0	0	0	0	0	0	0
35	1	1	0	13.9	6.3	0	0	0	0	0
36	1	1	0	13.9	6.3	0	0	0	0	0
37	2	1.0273	0	0	0	30	0	-20	35	0
38	1	1	0	15.6	7.1	0	0	0	0	0
39	1	1	0	1.5	0.7	0	0	0	0	0
40	1	1	0	21.6	9.8	0	0	0	0	0
41	2	1.0966	0	3	1.3	110	0	-35	45	0
42	2	1.034	0	0	0	70	0	-35	55	0
43	1	1	0	7.3	3.3	0	0	0	0	0
44	1	1	0	16.8	7.7	0	0	0	0	0
45	1	1	0	0	0	0	0	0	0	0
46	1	1	0	22.2	10.1	0	0	0	0	0
47	1	1	0	16.3	7.4	0	0	0	0	0
48	1	1	0	19.2	8.8	0	0	0	0	0
49	1	1	0	14.3	6.5	0	0	0	0	0
50	1	1	0	0	0	0	0	0	0	0
51	1	1	0	0	0	0	0	0	0	0
52	1	1	0	16	7.3	0	0	0	0	0
53	2	1	0	0	0	200	0	-100	160	0
54	1	1	0	7.3	3.3	0	0	0	0	0
55	1	1	0	8.7	4	0	0	0	0	0
56	1	1	0	7.2	3.3	0	0	0	0	0

57	1	1	0	0	0	0	0	0	0	0
58	1	1	0	22.3	10.1	0	0	0	0	0
59	1	1	0	0	0	0	0	0	0	0

Tableau A. 3 Données des lignes de transport (en p.u) du réseau 59 jeux de barres de Sonelgaz

From	To	R	X	B	From	To	R	X	B
1	38	0.152	0.483	0.0023	19	22	0.008	0.0285	0.0205
1	40	0.11	0.352	0.0017	19	32	0.016	0.057	0.041
2	20	0.019	0.12	0.0007	20	28	0.281	0.506	0.0023
2	55	0.004	0.023	0.0001	20	55	0.016	0.101	0.0006
3	20	0.018	0.119	0.0007	21	20	0.011	0.439	0
4	27	0.002	0.006	0.002	21	54	0.13	0.349	0.008
4	27	0.003	0.007	0.002	22	20	0.006	0.162	0
5	9	0.087	0.221	0.001	22	21	0.014	0.34	0
5	9	0.088	0.221	0.001	23	26	0.015	0.02	0.004
5	23	0.038	0.138	0.0006	23	27	0.026	0.034	0.007
5	23	0.038	0.14	0.0006	23	46	0.056	0.171	0.0008
5	27	0.045	0.167	0.0007	24	57	0.01378	0.04886	0.035
5	27	0.045	0.168	0.0008	25	29	0.217	0.369	0.0015
5	46	0.071	0.231	0.0011	26	27	0.013	0.017	0.004
6	5	0.002	0.054	0	28	43	0.27	0.477	0.0021
6	13	0.054	0.19	0.137	29	39	0.312	0.789	0.0037
6	13	0.057	0.201	0.144	30	29	0.006	0.216	0
6	30	0.018	0.085	0.064	30	45	0.032	0.15	0.113
6	30	0.025	0.086	0.062	31	34	0.0048	0.0168	0.012
7	40	0.527	0.887	0.0036	31	50	0.0095	0.0335	0.024
7	56	0.364	0.627	0.0026	32	34	0.008	0.0285	0.0205
8	14	0.214	0.491	0.0025	33	35	0.092	0.155	0.0006
8	25	0.157	0.395	0.0019	33	48	0.838	0.413	0.0057
9	14	0.21	0.366	0.0014	34	33	0.006	0.215	0
9	14	0.129	0.324	0.0015	36	43	0.334	0.578	0.0024
10	40	0.014	0.018	0.0014	38	44	0.327	0.561	0.0023
10	40	0.011	0.015	0.003	40	41	0.014	0.019	0.004
11	48	0.222	0.605	0.0026	40	58	0.106	0.301	0.0012
12	11	0.02	0.054	0	40	58	0.107	0.307	0.0012
12	37	0.013	0.045	0.007	42	59	0.00791	0.02806	0.02
13	3	0.014	0.326	0	43	52	0.094	0.16	0.0007
13	34	0.04	0.142	0.101	45	44	0.014	0.327	0
13	34	0.04	0.141	0.101	45	59	0.019	0.089	0.068
14	29	0.357	0.622	0.0023	47	49	0.339	0.857	0.0039
15	54	0.115	0.277	0.006	47	58	0.219	0.547	0.0026
16	15	0.014	0.285	0	49	56	0.016	0.028	0.0001
16	34	0.03	0.104	0.079	50	53	0.0048	0.0168	0.012
17	39	0.12	0.308	0.0014	51	53	0.0055	0.02	0.0143
17	44	0.37	0.949	0.0043	53	52	0.006	0.163	0
18	22	0.0055	0.02	0.0143	57	56	0.01	0.351	0

Annexe B

Réseaux tests de distribution

Tableau A. 4 Données du réseau de distribution IEEE 12-nœuds

ND	NA	P(kW)	Q(kVAR)	R(ohms)	X(ohms)	$I_{max}(A)$
1	2	60	60	1.093	0.455	-
2	3	40	30	1.184	0.494	-
3	4	55	55	2.095	0.873	-
4	5	30	30	3.188	1.329	-
5	6	20	15	1.093	0.455	-
6	7	55	55	1.002	0.417	-
7	8	45	45	4.403	1.215	-
8	9	40	40	5.642	1.597	-
9	10	35	30	2.89	0.818	-
10	11	40	30	1.514	0.428	-
11	12	15	15	1.238	0.351	-

SB = 100 MVA; VB = 11.00 kV.

Tableau A. 5 Données du réseau de distribution IEEE 33-nœuds

ND	NA	P(kW)	Q(kVAR)	R(ohms)	X(ohms)	$I_{max}(A)$
1	2	100	60	0.0922	0.0470	400
2	3	90	40	0.4930	0.2510	400
3	4	120	80	0.3661	0.1864	400
4	5	60	30	0.3811	0.1941	400
5	6	60	20	0.8190	0.7070	400
6	7	200	100	0.1872	0.6188	300
7	8	200	100	1.7117	1.2357	300
8	9	60	20	1.0299	0.7400	200
9	10	60	20	1.0440	0.7400	200
10	11	45	30	0.1967	0.0651	200
11	12	60	35	0.3744	0.1237	200
12	13	60	35	1.4680	1.1549	200
13	14	120	80	0.5416	0.7129	200
14	15	60	10	0.5909	0.5260	200
15	16	60	20	0.7462	0.5449	200
16	17	60	20	1.2889	1.7210	200
17	18	90	40	0.7320	0.5739	200
2	19	90	40	0.1640	0.1564	200
19	20	90	40	1.5042	1.3555	200
20	21	90	40	0.4095	0.4784	200
21	22	90	40	0.7089	0.9373	200
3	23	90	50	0.4512	0.3084	200
23	24	420	200	0.8980	0.7091	200
24	25	420	200	0.8959	0.7010	200
6	26	60	25	0.2031	0.1034	300
26	27	60	25	0.2842	0.1447	300
27	28	60	20	1.0589	0.9338	300
28	29	120	70	0.8043	0.7006	200
29	30	200	600	0.5074	0.2585	200
30	31	150	70	0.9745	0.9629	200

31	32	210	100	0.3105	0.3619	200
32	33	60	40	0.3411	0.5302	200
21	8	-	-	2.0000	2.0000	-
9	15	-	-	2.0000	2.0000	-
12	22	-	-	2.0000	2.0000	-
18	33	-	-	0.5000	0.5000	-
25	29	-	-	0.5000	0.5000	-

I_{\max} est le courant maximal en ampères, importé depuis le Ref. [54]. SB = 100 MVA; VB = 12.66 kV.

Tableau A. 6 Données du réseau de distribution IEEE 69-nœuds

ND	NA	P(kW)	Q(kVAR)	R(ohms)	X(ohms)	I _{max} (A)
1	2	0	0	0.0005	0.0012	400
2	3	0	0	0.0005	0.0012	400
3	4	0	0	0.0015	0.0036	400
4	5	0	0	0.0251	0.0294	400
5	6	2.6	2.2	0.3660	0.1864	400
6	7	40.4	30	0.3811	0.1941	400
7	8	75	54	0.0922	0.0470	400
8	9	30	22	0.0493	0.0257	400
9	10	28	19	0.8190	0.2707	400
10	11	145	104	0.1872	0.0619	200
11	12	145	104	0.7114	0.2351	200
12	13	8	5	1.0300	0.3400	200
13	14	8	5.5	1.0440	0.3450	200
14	15	0	0	1.0580	0.3496	200
15	16	45.5	30	0.1966	0.0650	200
16	17	60	35	0.3744	0.1238	200
17	18	60	35	0.0047	0.0016	200
18	19	0	0	0.3276	0.1083	200
19	20	1	0.6	0.2106	0.0696	200
20	21	114	81	0.3416	0.1129	200
21	22	5	3.5	0.0140	0.0046	200
22	23	0	0	0.1591	0.0526	200
23	24	28	20	0.3463	0.1145	200
24	25	0	0	0.7488	0.2475	200
25	26	14	10	0.3089	0.1021	200
26	27	14	10	0.1732	0.0572	200
3	28	26	18.6	0.0044	0.0108	200
28	29	26	18.6	0.0640	0.1565	200
29	30	0	0	0.3978	0.1315	200
30	31	0	0	0.0702	0.0232	200
31	32	0	0	0.3510	0.1160	200
32	33	14	10	0.8390	0.2816	200
33	34	19.5	14	1.7080	0.5646	200
34	35	6	4	1.4740	0.4873	200
3	36	26	18.55	0.0044	0.0108	200
36	37	26	18.55	0.0640	0.1565	200
37	38	0	0	0.1053	0.1230	200
38	39	24	17	0.0304	0.0355	200

39	40	24	17	0.0018	0.0021	200
40	41	1.2	1	0.7283	0.8509	200
41	42	0	0	0.3100	0.3623	200
42	43	6	4.3	0.0410	0.0478	200
43	44	0	0	0.0092	0.0116	200
44	45	39.22	26.3	0.1089	0.1373	200
45	46	39.22	26.3	0.0009	0.0012	200
4	47	0	0	0.0034	0.0084	300
47	48	79	56.4	0.0851	0.2083	300
48	49	384.7	274	0.2898	0.7091	300
49	50	384.7	274	0.0822	0.2011	300
8	51	40.5	28.3	0.0928	0.0473	300
51	52	3.6	2.7	0.3319	0.1114	200
9	53	4.35	3.5	0.1740	0.0886	300
53	54	26.4	19	0.2030	0.1034	300
54	55	26	17.2	0.2842	0.1447	300
55	56	0	0	0.2813	0.1433	300
56	57	0	0	1.5900	0.5337	300
57	58	0	0	0.7837	0.2630	300
58	59	100	72	0.3042	0.1006	300
59	60	0	0	0.3861	0.1172	300
60	61	1244	888	0.5075	0.2585	300
61	62	32	23	0.0974	0.0496	300
62	63	0	0	0.1450	0.0738	300
63	64	227	162	0.7105	0.3619	300
64	65	59	42	1.0410	0.5302	300
11	66	18	13	0.2012	0.0611	200
66	67	18	13	0.0047	0.0014	200
12	68	28	20	0.7394	0.2444	200
68	69	28	20	0.0047	0.0016	200
11	43	-	-	0.5000	0.5000	-
13	21	-	-	0.5000	0.5000	-
15	46	-	-	1.0000	1.0000	-
50	59	-	-	2.0000	2.0000	-
27	65	-	-	1.0000	1.0000	-

I_{\max} est le courant maximal en ampères, importé depuis le Ref. [54]. SB = 100 MVA; VB = 12.66 kV.

Tableau A. 7 Données du réseau de distribution IEEE 118-nœuds

ND	NA	P(kW)	Q(kVAR)	R(ohms)	X(ohms)	I _{max} (A)
1	2	133.84	101.14	0.000298	0.000107	-
2	3	16.214	11.292	0.000273	0.000098	-
2	4	34.315	21.845	0.000372	0.000134	-
4	5	73.016	63.602	0.000124	0.000446	-
5	6	144.2	68.604	0.000124	0.000446	-
6	7	104.47	61.725	0.000124	0.000103	-
7	8	28.547	11.503	0.000149	0.000116	-
8	9	87.56	51.073	0.000174	0.000521	-
2	10	198.2	106.77	0.001372	0.001111	-
10	11	146.8	75.995	0.000926	0.000652	-

11	12	26.04	18.687	0.001546	0.002587	-
12	13	52.1	23.22	0.001174	0.001250	-
13	14	141.9	117.5	0.001488	0.000975	-
14	15	21.87	28.79	0.001240	0.000372	-
15	16	33.37	26.45	0.001322	0.001488	-
16	17	32.43	25.23	0.001298	0.001413	-
11	18	20.234	11.906	0.001802	0.002355	-
18	19	156.94	78.523	0.000975	0.001529	-
19	20	546.29	351.4	0.001322	0.001620	-
20	21	180.31	164.2	0.000992	0.001562	-
21	22	93.167	54.594	0.000992	0.000652	-
22	23	85.18	39.65	0.011653	0.005975	-
23	24	168.1	95.178	0.002422	0.001114	-
24	25	125.11	150.22	0.001099	0.000860	-
25	26	16.03	24.62	0.001471	0.001107	-
26	27	26.03	24.62	0.001471	0.001107	-
4	28	594.56	522.62	0.000124	0.000245	-
28	29	120.62	59.117	0.000099	0.000228	-
29	30	102.38	99.554	0.000992	0.002286	-
30	31	513.4	318.5	0.001736	0.002008	-
31	32	475.25	456.14	0.000992	0.000446	-
32	33	151.43	136.79	0.001471	0.001934	-
33	34	205.38	83.302	0.001471	0.001934	-
34	35	131.6	93.082	0.001273	0.001339	-
30	36	448.4	369.79	0.001546	0.002157	-
36	37	440.52	321.64	0.001099	0.000818	-
29	38	112.54	55.134	0.002727	0.001603	-
38	39	53.963	38.998	0.002562	0.001603	-
39	40	393.05	342.6	0.001074	0.001603	-
40	41	326.74	278.56	0.002314	0.001240	-
41	42	536.26	240.24	0.009752	0.007025	-
42	43	76.247	66.562	0.003471	0.002013	-
43	44	53.52	39.76	0.002231	0.000803	-
44	45	40.328	31.964	0.002802	0.001009	-
45	46	39.653	20.758	0.002231	0.001470	-
35	47	66.195	42.361	0.001736	0.001143	-
47	48	73.904	51.653	0.000992	0.000652	-
48	49	114.77	57.965	0.001240	0.000816	-
49	50	918.37	1205.1	0.001240	0.000816	-
50	51	210.3	146.66	0.001984	0.001307	-
51	52	66.68	56.608	0.000992	0.000652	-
52	53	42.207	40.184	0.003347	0.001205	-
53	54	433.74	283.41	0.003347	0.001205	-
29	55	62.1	26.86	0.003231	0.001165	-
55	56	92.46	88.38	0.003355	0.001207	-
56	57	85.188	55.436	0.003355	0.001207	-
57	58	345.3	332.4	0.005835	0.004513	-
58	59	22.5	16.83	0.002793	0.001007	-
59	60	80.551	49.156	0.002793	0.001007	-
60	61	95.86	90.758	0.001711	0.000617	-

61	62	62.92	47.7	0.002041	0.007374	-
1	63	478.8	463.74	0.000231	0.000346	-
63	64	120.94	52.006	0.000967	0.001666	-
64	65	139.11	100.34	0.002107	0.000759	-
65	66	391.78	193.5	0.001736	0.000627	-
66	67	27.741	26.713	0.003165	0.001141	-
67	68	52.814	25.257	0.004165	0.002730	-
68	69	66.89	38.713	0.003355	0.001207	-
69	70	467.5	395.14	0.007950	0.006289	-
70	71	594.85	239.74	0.001364	0.000496	-
71	72	132.5	84.363	0.002504	0.000903	-
72	73	52.699	22.482	0.002504	0.000903	-
73	74	869.79	614.775	0.001703	0.001190	-
74	75	31.349	29.817	0.001926	0.000694	-
75	76	192.39	122.43	0.004884	0.001465	-
76	77	65.75	45.37	0.001041	0.000374	-
64	78	238.15	223.22	0.004620	0.003047	-
78	79	294.55	162.47	0.001537	0.001014	-
79	80	485.57	437.92	0.001537	0.001014	-
80	81	243.53	183.03	0.002149	0.001149	-
81	82	243.53	183.03	0.001273	0.001223	-
82	83	134.25	119.29	0.001901	0.001058	-
83	84	22.71	27.96	0.002083	0.000876	-
84	85	49.513	26.515	0.001488	0.001223	-
79	86	383.78	257.16	0.001322	0.001504	-
86	87	49.64	20.6	0.001653	0.001901	-
87	88	22.473	11.806	0.001322	0.003248	-
65	89	62.93	42.96	0.005529	0.001993	-
89	90	30.67	34.93	0.002198	0.001014	-
90	91	62.53	66.79	0.002198	0.001014	-
91	92	114.57	81.748	0.002198	0.001014	-
92	93	81.292	66.526	0.002198	0.001014	-
93	94	31.733	15.96	0.001926	0.000950	-
94	95	33.32	60.48	0.004099	0.001141	-
91	96	531.28	224.85	0.001620	0.001488	-
96	97	507.03	367.42	0.001620	0.001488	-
97	98	26.39	11.7	0.001542	0.001008	-
98	99	45.99	30.392	0.000617	0.002628	-
1	100	100.66	47.572	0.000517	0.000219	-
100	101	456.48	350.3	0.001241	0.001934	-
101	102	522.56	449.29	0.001113	0.000734	-
102	103	408.43	168.46	0.001907	0.000994	-
103	104	141.48	134.25	0.003694	0.001329	-
104	105	104.43	66.024	0.001349	0.000486	-
105	106	96.793	83.647	0.002727	0.000818	-
106	107	493.92	419.34	0.001289	0.000464	-
107	108	225.38	135.88	0.003156	0.001136	-
108	109	509.21	387.21	0.001344	0.000484	-
109	110	188.5	173.46	0.003156	0.001136	-
110	111	918.03	898.55	0.002021	0.000726	-

110	112	305.08	215.37	0.001726	0.000622	-
112	113	54.38	40.97	0.001902	0.000684	-
100	114	211.14	192.9	0.005043	0.001815	-
114	115	67.009	53.336	0.001542	0.001050	-
115	116	162.07	90.321	0.003084	0.002033	-
116	117	48.785	29.156	0.003347	0.003033	-
117	118	33.9	18.98	0.004041	0.003620	-
46	27	-	-	0.004346	0.002417	-
17	27	-	-	0.004346	0.002410	-
8	24	-	-	0.003531	0.001272	-
54	43	-	-	0.003967	0.001428	-
62	49	-	-	0.002975	0.001071	-
37	62	-	-	0.004711	0.004727	-
9	40	-	-	0.004380	0.002767	-
58	96	-	-	0.003270	0.001178	-
73	91	-	-	0.005620	0.005355	-
88	75	-	-	0.003357	0.001210	-
99	77	-	-	0.003823	0.001384	-
108	83	-	-	0.005380	0.001934	-
105	86	-	-	0.006715	0.002417	-
110	118	-	-	0.005859	0.002110	-
25	35	-	-	0.004132	0.004132	-

SB = 1 MVA; VB = 11.00 kV.

Annexe C

C.1 Performance des éoliennes

La puissance d'une éolienne varie comme le cube de la vitesse (ϵ_w) du vent [90, 91]:

$$P_{rated} = 0.5 \dots A \epsilon_w^3 C_p \quad \text{B.35}$$

avec \dots = densité de l'air ; A = surface balayée par l'éolienne ; C_p = rendement de l'éolienne ou efficacité dynamique.

$$P_w = \begin{cases} 0 & \epsilon_w < \epsilon_{cin} \text{ ou } \epsilon_w > \epsilon_{cout} \\ P_{rated} \frac{\epsilon_w - \epsilon_{cin}}{\epsilon_N - \epsilon_{cin}} & \epsilon_{cout} \leq \epsilon_w \leq \epsilon_N \\ P_{rated} & \epsilon_N \leq \epsilon_w \leq \epsilon_{cout} \end{cases} \quad \text{B.36}$$

Avec ϵ_N est la puissance nominale de l'éolienne, ϵ_{cin} (*cin* pour cut-in speed) est la vitesse au-dessous de laquelle le couple fourni par l'éolienne ne peut compenser les différents frottements. La puissance fournie au réseau est nulle pour $\epsilon_w < \epsilon_{cin}$. La vitesse ϵ_{cout} (*cout* pour cut-out speed) est la vitesse au-delà de laquelle on arrête l'éolienne car la puissance est trop importante. En cas de vent fort, les éoliennes modernes contrôlent de manière aérodynamique l'hélice de l'éolienne pour maintenir la puissance nominale entre ϵ_N et ϵ_{cout} .

C.2 Performance des modules photovoltaïques

Le rayonnement solaire et la température ambiante sont les principaux gouverneurs pour le dimensionnement des modules photovoltaïques (PV). Pour un générateur photovoltaïque composé de N_s modules en série et N_p modules en parallèle, la puissance de sortie maximale peut être calculé par [91]:

$$P_{pv} = N_s \times N_p \times P_{md} \quad \text{B.37}$$

avec

$$P_{md} = FF \times V_{oc} \times I_{sc} \quad \text{B.38}$$

FF , V_{oc} et I_{sc} sont respectivement le facteur de forme, la tension du circuit ouvert et le courant de court-circuit du PV données par :

$$V_{oc} = \frac{V_{Noc}}{1 + c_2 \times \ln \frac{G_N}{G_a}} \left(\frac{T_N}{T_a} \right)^{c_1} \quad \text{B.39}$$

$$I_{sc} = I_{Nsc} \left(\frac{G_a}{G_N} \right)^{c_3} \quad \text{B.40}$$

$$FF = \left(1 - \frac{R_s}{V_{oc}/I_{sc}} \right) \frac{\frac{V_{oc}}{nKT_a/q} - \ln \left(\frac{V_{oc}}{nKT_a/q} + 0.72 \right)}{1 + \frac{V_{oc}}{nKT_a/q}} \quad \text{B.41}$$

G_N et G_a sont les irradiances solaire nominale et actuelle, T_N et T_a sont les températures nominale et actuelle, R_s est la résistance série du module. c_1 , c_2 et c_3 trois constantes définies dans la référence [91], n le facteur de densité ($n = 1.5$); K la constante de Boltzmann (1.38×10^{-23} J/K); et q est la charge de l'électron (1.6×10^{-19} C).