

Titre: Analyse paramétrique des réseaux électriques sur MATLAB
Title:

Auteur: Philippe Junior Ossoucah
Author:

Date: 2010

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Ossoucah, P. J. (2010). Analyse paramétrique des réseaux électriques sur MATLAB [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie.
Citation: <https://publications.polymtl.ca/486/>

Document en libre accès dans PolyPublie

Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/486/>
PolyPublie URL:

Directeurs de recherche: Frédéric Sirois
Advisors:

Programme: Génie Électrique
Program:

UNIVERSITÉ DE MONTRÉAL

**ANALYSE PARAMÉTRIQUE DES RÉSEAUX ÉLECTRIQUES SUR
MATLAB**

PHILIPPE JUNIOR OSSOUCAH
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE ÉLECTRIQUE)
DÉCEMBRE 2010

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

ANALYSE PARAMÉTRIQUE DES RÉSEAUX ÉLECTRIQUES SUR MATLAB

Présenté par : OSSOUCAH Philippe Junior

en vue de l'obtention du diplôme de : Maîtrise ès Sciences Appliquées

a été dûment accepté par le jury d'examen constitué de :

M. MAHSEREDJIAN Jean, Ph.D., président

M. SIROIS Frédéric, Ph.D., membre et directeur de recherche

M. HUNEAULT Maurice, Ph.D., membre

DÉDICACE

*J'aimerais dédicacer mon mémoire à mes deux fils,
William et Enzo.*

REMERCIEMENTS

J'aimerais remercier mon directeur de recherche, Frédéric Sirois, qui m'a épaulé, guidé et conseillé tout au long de mes travaux, ma conjointe, Myriam Vaugon, pour son soutien et sa patience, mais également toutes les personnes qui ont su faire grandir ce projet par leur encouragement et leurs nombreuses idées.

RÉSUMÉ

Le domaine de l'énergie est en pleine mutation. Les sociétés d'énergie souhaitent tirer partie des nouvelles technologies et sources d'énergie afin de réduire les coûts d'opération et augmenter l'efficacité et la fiabilité de leur réseau. Afin de prédire l'impact de ces changements, l'institut de recherche d'Hydro-Québec a décidé de développer, à travers un vaste projet, le savoir-faire et les outils dédiés à l'évaluation des nouvelles architectures et règles d'exploitation de ces futurs réseaux électriques.

L'outil développé dans le cadre de cette présente recherche fait partir de ce grand projet et propose une nouvelle approche pour réaliser des analyses paramétriques et étudier l'évolution des réseaux électriques.

Le nouvel outil d'analyse de réseaux électriques est développé en langage Orienté Objet sur Matlab. Des boucles de simulations peuvent être programmées dans un langage de haut niveau afin d'étudier l'évolution des paramètres et la topologie des réseaux électriques et évaluer la fiabilité des réseaux selon la méthode de Monte-Carlo. L'objectif est d'analyser l'évolution des réseaux électriques en tenant compte d'une multitude de contraintes aléatoires, quelles soient techniques, économiques ou environnementales.

L'environnement Matlab offre à l'usager une plateforme de développement dans un langage de haut niveau sur laquelle il peut adapter l'outil d'Analyse Paramétrique des Réseaux Électriques sur Matlab (APREM) à ces propres besoins d'analyse de réseaux électriques.

ABSTRACT

The Power System field is going through a lot of changes. Utilities are willing to take advantage of new energy sources and technologies with the aim of reducing operating costs and increasing efficiency and reliability of their network. In order to predict those changes and quantify their impact, Hydro-Québec Research Institute decided to develop, through a large project, the knowhow and tools dedicated to the evaluation of new architectures and operating processes of those future networks.

The tool developed under this present research forms is part of this large project and proposes a new method to run parameterized studies and addresses the evolution of electrical networks.

The new network analysis tool developed uses the object-oriented programming capabilities of Matlab. Loop simulations can be performed in a high level language in order to study the evolution of network parameters and topology and evaluate the reliability of power systems using the Monte-Carlo method. The aim is to address the evolution of electrical networks taking account of various random constraints arising from environmental, technical or economic challenges.

The Matlab environment provides to the user a development platform in a high level language on which he can adapt the APREM tool to its electrical analysis needs.

TABLE DES MATIÈRES

DÉDICACE	III
REMERCIEMENTS	IV
RÉSUMÉ	V
ABSTRACT	VI
TABLE DES MATIÈRES	VII
LISTE DES TABLEAUX.....	X
LISTE DES FIGURES.....	XI
LISTE DES ANNEXES.....	XIV
INTRODUCTION.....	1
CHAPITRE 1 REVUE DE LITTÉRATURE ET ÉTAT DE L'ART.....	6
1.1 Fiabilité.....	6
1.1.1 Techniques analytiques	6
1.1.2 Simulations (Monte-Carlo)	14
1.2 Choix de l'algorithme de calcul de répartition de puissance	21
1.3 Logiciel commerciaux.....	24
1.4 Matlab orienté objet	27
CHAPITRE 2 APPROCHE MÉTHODOLOGIQUE.....	29
CHAPITRE 3 CORPS DE L'OUVRAGE : OUTIL APREM	32
3.1 Matlab.....	32
3.2 Matrice nodale augmentée modifiée	32
3.2.1 Matrice d'admittances	33
3.2.2 Matrices des sources de tension	35
3.2.3 Matrices des sources de courant.....	37

3.2.4 Matrices des transformateurs à deux enroulements	38
3.2.5 Matrices des interrupteurs	40
3.3 Ajout des contraintes de puissance à la matrice nodale	42
3.3.1 Contraintes des charges PQ.....	42
3.3.2 Contraintes des sources PV	44
3.4 Écoulement de puissance	45
3.4.1 Méthode de Newton-Raphson.....	45
3.4.2 La matrice Jacobienne.....	46
3.4.3 Initialisation et itérations de Newton-Raphson	49
3.5 Les nouvelles classes de l'outil APREM	52
3.5.1 Organisation des classes.....	52
3.5.2 La classe cEquipment.....	58
3.5.3 La classe cImpedance.....	62
3.5.4 La classe cIsource.....	66
3.5.5 La classe cVsource	69
3.5.6 La classe cTxfo.....	73
3.5.7 La classe cSwitch	78
3.5.8 La classe cLoad	82
3.5.9 La classe cPVsource.....	86
3.6 Comment fonctionne l'outil APREM	90
3.6.1 Création du réseau	90
3.6.2 Écoulement de puissance du réseau	92
3.6.3 La fonction fIsland	95
3.6.4 Analyse paramétrique.....	98

3.6.5 Le paramètre <i>function</i> et la fiabilité	100
3.7 Validation et résultats.....	105
3.7.1 Exemple I : projet de ligne en Bolivie.....	105
3.7.2 Exemple II : projet étudiant - cours de fiabilité	123
3.7.3 Exemple III : outil APREM versus PSS/E	129
CHAPITRE 4 CONLUSION ET RECOMMANDATIONS POUR TRAVAUX FUTURS..	135
CHAPITRE 5 RÉFÉRENCES	137

LISTE DES TABLEAUX

Tableau 1.1 : <i>Capacity Outage Probability Table (COPT)</i>	8
Tableau 1.2 : <i>Loss Of Load Probability (LOLP)</i>	10
Tableau 1.3 : <i>Loss Of Load Expectation (LOLE)</i>	11
Tableau 3.1 : Données de base sur l'évolution des charges	106
Tableau 3.2 : Exemple I : données du réseau	109
Tableau 3.3 : Exemple I : niveaux de tension sur les barres obtenus sur APREM et PSAF.....	122
Tableau 3.4 : Exemple I : transits de puissance obtenus sur APREM et PSAF	122
Tableau 3.5 : Exemple II : données du réseau.....	124
Tableau 3.6 : Exemple II : LOLP sans reconfiguration	127
Tableau 3.7 : Exemple II :LOLP avec reconfiguration.....	127
Tableau 3.8 : Exemple III : résultats comparatifs entre l'outil APREM et PSS/E	130
Tableau 3.9 : Exemple III : niveaux de tension obtenus sur APREM et PSS/E	131
Tableau 3.10 : Exemple III : données du réseau	132

LISTE DES FIGURES

Figure 1.1 : Évolution de la probabilité avec le nombre de simulation	16
Figure 3.1 : Schéma de référence d'une impédance.....	34
Figure 3.2 : Schéma de référence d'une impédance connectée aux nœuds k et m	35
Figure 3.3 : Schéma de référence d'une source de tension	35
Figure 3.4 : Schéma de référence d'une source de courant.....	37
Figure 3.5 : Schéma de référence d'un transformateur idéal à deux enroulements	38
Figure 3.6 : Schéma de référence d'un interrupteur (switch)	40
Figure 3.7 : Schéma de référence d'une charge PQ.....	43
Figure 3.8 : Schéma de référence d'une source PV	44
Figure 3.9 : la classe <i>cEquipement</i>	53
Figure 3.10 : la fonction <i>fNode</i>	54
Figure 3.11 : la fonction <i>fCurrent</i>	55
Figure 3.12 : Organisation des classes dans l'outil APREM	57
Figure 3.13 : fonction built-in <i>fieldnames</i> de Matlab.....	58
Figure 3.14 : fonction <i>fieldnames</i> redéfinie dans l'outil APREM.....	59
Figure 3.15 : constructeur de la classe <i>cEquipement</i>	60
Figure 3.16 : fonction <i>fAdjancyList</i> de <i>cEquipement</i>	61
Figure 3.17 : construction de la classe <i>cImpedance</i>	63
Figure 3.18 : fonction <i>fMatrix</i> de la classe <i>cImpedance</i>	64
Figure 3.19 : constructeur de la classe <i>cIsource</i>	67
Figure 3.20 : fonction <i>fMatrix</i> de la classe <i>cIsource</i>	68
Figure 3.21 : constructeur de la classe <i>cVsource</i>	70
Figure 3.22 : fonction <i>fMatrix</i> de la classe <i>cVsource</i>	71

Figure 3.23 : constructeur de la classe <i>cTxfo</i>	75
Figure 3.24 : fonction <i>fMatrix</i> de la classe <i>cTxfo</i>	76
Figure 3.25 : fonction <i>fAdjencyList</i> de la classe <i>cTxfo</i>	77
Figure 3.26 : constructeur de la classe <i>cSwitch</i>	79
Figure 3.27 : fonction <i>fMatrix</i> de la classe <i>cSwitch</i>	80
Figure 3.28 : constructeur de la classe <i>cLoad</i>	83
Figure 3.29 : réseau test pour initialiser l'objet de la classe <i>cLoad</i>	84
Figure 3.30 : fonction <i>fMatrix</i> de la classe <i>cLoad</i>	84
Figure 3.31 : constructeur de la classe <i>cPVsource</i>	87
Figure 3.32 : réseau test pour initialiser l'objet de la classe <i>cPVsource</i>	88
Figure 3.33 : fonction <i>fMatrix</i> de la classe <i>cPVsource</i>	89
Figure 3.34 : création du réseau Network dans Matlab : <i>NetworkStruct.m</i>	90
Figure 3.35 : réseau Network crée à partir du fichier <i>NetworkStruct.m</i>	91
Figure 3.36 : structure Network créé à partir du fichier <i>NetworkStruct.m</i>	92
Figure 3.37 : Impression des résultats de l'écoulement de puissance de Network avec <i>fReport</i>	93
Figure 3.38 : structure de la fonction <i>fLoadFlow</i>	94
Figure 3.39 : liste d'adjacence du réseau Network	96
Figure 3.40 : fonction <i>fTraversal</i>	97
Figure 3.41 : Analyse paramétrique avec l'outil APREM	99
Figure 3.42 : tension à la barre B3 de Network en fonction de la charge PQ3	100
Figure 3.43 : utilisation du paramètre <i>function</i> pour l'étude de fiabilité	102
Figure 3.44 : fonction <i>fProbabilityTest</i> pour simuler les pannes d'équipement	102
Figure 3.45 : Simulation Monte-Carlo avec l'outil APREM	103
Figure 3.46 : Exportation des résultats vers Excel : <i>fReportxls</i>	104

Figure 3.47 : Exemple I : projet de ligne en Bolivie	105
Figure 3.48 : Évolution des charges	106
Figure 3.49 : fichier <i>NetworkDefinition.m</i> (1/2)	107
Figure 3.50 : fichier <i>NetworkDefinition.m</i> (2/2)	108
Figure 3.51 : niveau de tension sur la ligne sans compensation	110
Figure 3.52 : fichier <i>Bolivie_MAIN.m</i> (1/3).....	111
Figure 3.53 : fichier <i>Bolivie_MAIN.m</i> (2/3).....	112
Figure 3.54 : fichier <i>Bolivie_MAIN.m</i> (3/3).....	113
Figure 3.55 : fichier <i>compensation.m</i>	114
Figure 3.56 : fichier <i>losses.m</i>	115
Figure 3.57 : fonction <i>fSave.m</i>	116
Figure 3.58 : Contenu de la structure <i>SavedData</i> retournée par la fonction <i>fSave</i>	117
Figure 3.59 : aperçu du fichier <i>Bolovie_230kV_Line.xls</i>	118
Figure 3.60 : Profile de tension, compensation et charge (ex : ligne 230kV, Bolivie)	119
Figure 3.61 : Puissance réelle et réactive à la centrale (ex : ligne 230kV, Bolivie).....	120
Figure 3.62 : Pertes sur la ligne 230kV (ex : ligne 230kV, Bolivie).....	121
Figure 3.63 : exemple II, étude de fiabilité d'un réseau de distribution [39].....	123
Figure 3.64 : exemple III, outil APREM versus PSS/E	129
Figure 3.65 : résultats comparatifs entre l'outil APREM et PSS/E	134

LISTE DES ANNEXES

ANNEXE 1	EXEMPLE I : DONNÉES DU RÉSEAU ET RÉSULTATS D'ÉCOULEMENT DE PUISSANCE SUR PSAF	140
ANNEXE 2	EXEMPLE I : CODE MATLAB DU RÉSEAU ET RÉSULTATS D'ÉCOULEMENT DE PUISSANCE SUR APREM.....	144
ANNEXE 3	EXEMPLE II : CODE MATLAB DU RÉSEAU ET RÉSULTATS D'ÉCOULEMENT DE PUISSANCE SUR APREM.....	147
ANNEXE 4	EXEMPLE III : DONNÉES DU RÉSEAU ET RÉSULTATS D'ÉCOULEMENT DE PUISSANCE SUR PSS/E	153
ANNEXE 5	EXEMPLE III : DONNÉES DU RÉSEAU ET RÉSULTATS D'ÉCOULEMENT DE PUISSANCE SUR L'OUTIL APREM.....	162

INTRODUCTION

Contexte

Avec l'avancée des nouvelles technologies de production, d'automatisation et de communication dans le domaine de l'énergie, le secteur est en profonde mutation. Les industries et les services d'électricité y voient une occasion pour moderniser à grande échelle leurs installations afin de réduire les coûts d'opération de leur réseau, tout en augmentant leur fiabilité, leur efficacité et leur sécurité. L'ère du numérique change les modèles de gestion des réseaux et les concepts du « réseau intelligent », ou encore de « production distribuée » émergeant tant en Europe qu'en Amérique du Nord.

Plusieurs distributeurs d'électricité dont Hydro-Québec ont commandé plusieurs études sur l'évolution du secteur de la distribution [1]. Cette démarche commune des distributeurs vise à identifier les lacunes techniques, opérationnelles, et même légales auxquelles ils doivent remédier pour faire face aux changements annoncés dans le secteur. Aux États-Unis, les autorités observent même une dégradation de la fiabilité dans plusieurs régions causée par le manque d'investissement et l'âge avancé des équipements [2]. Parmi les besoins les plus criants, trois (3) constats :

- Les réseaux de distribution doivent être plus intelligents.
- Les nouvelles technologies auront un impact sur les réseaux de distribution et leur intégration doit être planifiée.
- Les règles de tarification et d'opération auront le plus grand impact sur la capacité des distributeurs d'opérer leur réseau, car ces dernières seront de plus en plus imposées par les législateurs et les régulateurs sous l'influence du public et des groupes de pression.

Ces constats viennent renforcer l'idée que l'évolution des réseaux électriques s'organisera suivant une pléthore de paramètres techniques, économiques et sociaux qui façonneront les choix conceptuels des réseaux électriques de la prochaine génération. Force est de reconnaître que chacun des distributeurs en opération en Amérique du Nord définira sa propre orientation pour qu'elle réponde le mieux à ses besoins.

Certaines industries, comme Hydro-Québec, ont emboité le pas en automatisant largement leurs centres de production et leurs réseaux de transport afin d'assurer une supervision continue de leurs installations [3]. Les réflexions engagées sur l'architecture de la prochaine génération de réseaux de distribution ont permis à Hydro-Québec de bâtir son propre plan d'évolution du réseau de distribution [4]. D'autres, tel que HydroOttawa [5] partent à la conquête des « compteurs intelligents ». Le point commun dans l'action des distributeurs d'électricité est qu'ils souhaitent optimiser leurs installations et ajouter sur l'intégralité de leurs réseaux des points de mesures, de contrôle et de télécommunication dans le but d'accroître la performance par une gestion dynamique de l'information recueillie à distance.

Ces investissements ciblés veulent répondre à une demande croissante de la clientèle vis-à-vis de la fiabilité et la qualité de livraison de l'électricité, et un besoin de l'industrie de réduire les coûts futurs d'infrastructures de grande puissance, en les reportant grâce à une meilleure exploitation des réseaux. Pour le distributeur, il est alors primordial d'orienter et planifier adéquatement ces investissements afin d'obtenir les résultats escomptés.

Cadre du projet de recherche

Afin d'anticiper l'évolution technologique du secteur de la distribution, Hydro-Québec, comme toutes les entreprises assurant la distribution de l'électricité, doit se positionner sur ses choix à long terme relatifs à l'architecture de son réseau de distribution afin de répondre aux nouvelles contraintes technico-socio-économiques. Dans ce processus de réflexion, l'institut de Recherche d'Hydro-Québec (IREQ) a reconnu la nécessité de développer de nouvelles compétences et outils d'analyses électriques qui permettront d'orienter judicieusement les choix stratégiques et techniques d'Hydro-Québec.

C'est dans le cadre du projet de Laboratoire d'Évaluation Opérationnelle des Processus et des Architectures de Réseau (LEOPAR) que l'IREQ souhaite développer son savoir-faire et les outils d'analyse de l'évolution des réseaux électriques. Au stade actuel, le projet LEOPAR se penche plus spécifiquement sur le développement d'une application de logique descriptive pour la représentation des connaissances des réseaux électriques. Mais à terme, l'outil regroupera différents modules de traitement de données afin de réaliser diverses analyses sur les réseaux électriques décrits dans une base de connaissances commune de l'IREQ.

Le projet de recherche présenté dans ces pages s'insère dans le cadre du projet LEOPAR et vise le développement d'une application d'analyse paramétrique des réseaux électriques pour l'étude et l'optimisation des architectures de réseaux de distribution.

Objectifs de recherche

Notre objectif de recherche s'intéresse principalement à l'évolution de la topologie et des paramètres des réseaux électriques. Bien que de nombreux logiciels d'analyse de réseaux existent, ces derniers partagent souvent une contrainte commune : ils cherchent avant tout à répondre aux besoins de l'utilisateur final en proposant des interfaces et algorithmes de calculs pour faciliter la modélisation et l'analyse de réseaux dont les paramètres et les topologies sont le plus souvent invariants. Pour traiter plusieurs variantes aux scénarios de base modélisés par l'utilisateur, les logiciels proposent généralement l'utilisation de modules d'analyse de contingence qui s'appuient sur des listes de scénarios prédefinies par l'utilisateur. L'usager doit donc intervenir sur son modèle de réseau à chaque fois qu'il souhaite modifier son scénario de base. Néanmoins, certains logiciels offrent l'accès à des scripts qui permettent de s'affranchir des restrictions imposées par les interfaces entre l'utilisateur et le code source des logiciels. Cela permet de personnaliser les études de réseaux, mais l'utilisation de ces scripts requiert bien souvent des aptitudes avancées en programmation.

L'objectif de recherche est donc la création d'un outil de recherche qui rendra plus accessible et facile la programmation d'études paramétriques de réseaux électriques par l'utilisateur final. L'outil souhaite répondre à la nécessité d'avoir accès facilement aux données afin de programmer avec un minimum d'efforts et d'opérations les modifications à appliquer automatiquement sur la topologie et les paramètres des réseaux à chaque simulation sans l'intervention directe de l'utilisateur durant le processus. . L'outil doit offrir un environnement simple d'utilisation dans lequel l'usager est libre de créer ses scénarios et programmer dans un langage haut niveau et en quelques lignes l'évolution et/ou les modifications sur son réseau.

L'analyse paramétrique offre d'importantes possibilités pour l'étude de la fiabilité des réseaux électriques, car, contrairement aux méthodes analytiques, elle donne accès aux études probabilistes telles que les méthodes de Monte-Carlo qui peuvent prendre en compte l'ensemble des données statistiques d'un même réseau.

Contenu de ce mémoire

Ce mémoire présente donc l'approche adoptée pour développer le premier prototype de cet outil de recherche. L'objectif du projet de maîtrise étant d'aboutir à une première version fonctionnelle afin de valider le concept, plusieurs choix techniques ont été retenus, entre autre concernant l'algorithme de calculs d'écoulement de puissance, la structure du code et plusieurs autres fonctionnalités, sans pour autant restreindre le développement futur de l'outil de recherche sur ces choix techniques. L'optimisation du code et de l'outil dans son ensemble, l'intégration d'une sélection de méthodes de calculs de réseaux parmi les plus populaires et l'utilisation d'une interface graphique ne sont pas couverts par ce mémoire et feront donc l'objet de développements et améliorations futures.

Dans la première partie de ce mémoire, une grande place est accordée à la théorie entourant l'étude de la fiabilité des réseaux, car l'outil de recherche vise ultimement à répondre à la demande de plus en plus importante d'étudier l'impact qu'ont les architectures et la composition des réseaux électriques sur la fiabilité globale des systèmes. L'intention est de faire ressortir les limitations et contraintes des techniques analytiques employées jusqu'à maintenant pour faire l'analyse de la fiabilité des réseaux électriques. Le lecteur non familier avec ce domaine sera ainsi en mesure de saisir l'importance de poursuivre le développement d'une solution logicielle pour aborder et rendre plus accessible l'étude de la fiabilité des réseaux électriques, notamment via les simulations de type Monte-Carlo.

Le premier prototype de l'outil de recherche est développé sous Matlab en code Orienté-Objet et permet de faire la promotion des avantages qu'offrent l'utilisation de Matlab dans la programmation d'écoulements de puissance paramétriques. L'utilisateur final, qui est en même temps programmeur, a un libre accès au code source et aux paramètres de tous les équipements de son réseau. Il peut ainsi développer ses propres algorithmes paramétriques, les appliquer à chaque équipement de son réseau et ainsi personnaliser chacune de ses études.

L'utilisation de Matlab est également intéressante pour la recherche visée, car elle permettra une plus facile collaboration de diverses expertises techniques dans d'autres domaines tels que les statistiques, l'économie, l'intelligence artificielle ou même l'hydrologie. Cette contribution peut être mise en œuvre dans l'élaboration d'algorithmes paramétriques qui seront alors fonction de données statistiques, économiques, ou hydrologiques par exemple.

Le prototype illustre également les possibilités qui s'ouvrent vis-à-vis des études de fiabilité des réseaux électriques assistées par ordinateur étant donné que l'utilisateur a toute la liberté d'associer des probabilités à ses algorithmes de changement d'état et de paramètres de son réseau. L'outil permet ainsi de réaliser des simulations de type Monte-Carlo afin d'évaluer numériquement la fiabilité des réseaux électriques.

CHAPITRE 1 REVUE DE LITTÉRATURE ET ÉTAT DE L'ART

1.1 Fiabilité

Depuis quelques années, l'évaluation de la fiabilité des réseaux électriques prend de l'importance chez les ingénieurs et spécialistes. Un nombre considérable d'articles décrivant les différentes techniques disponibles de modélisation des réseaux pour le calcul des indices de fiabilité sont publiés [6]. L'intérêt grandissant autour de la fiabilité des réseaux électriques est également motivé par les résultats de plusieurs analyses statistiques qui démontrent que la composition des systèmes électriques contribue à hauteur de 80% sur l'indisponibilité du service électrique chez les consommateurs [7].

Les indices de performance en termes de fiabilité des réseaux électriques peuvent être calculés suivant une multitude de méthodes formant deux groupes bien distincts [8] :

- Les techniques analytiques;
- Les simulations.

Aujourd'hui, l'approche analytique représente une très grande proportion des techniques utilisées par les spécialistes dans l'évaluation de la fiabilité des réseaux au détriment des simulations qui nécessitent des temps de calcul trop importants. Les modèles et techniques analytiques disponibles suffisent encore aux planificateurs et concepteurs pour appuyer certaines décisions stratégiques. Néanmoins, cette tendance change due aux possibilités qu'offrent les ressources informatiques disponibles de nos jours. En modélisant plus intuitivement les réseaux électriques et les défaillances qui les caractérisent, l'évaluation des performances peut ainsi prendre en compte un plus grand nombre de paramètres et ainsi rendre l'étude de fiabilité plus tangible. L'approche probabiliste de type Monte-Carlo est la plus considérée pour l'évaluation de la fiabilité des réseaux électriques puisqu'elle se prête aux outils informatiques.

1.1.1 Techniques analytiques

L'approche analytique vise la représentation des systèmes électriques sous forme mathématique et évalue les différents indices de fiabilités en solutionnant numériquement les modèles algébriques équivalents. L'application des notions de base de probabilité demeure l'approche la

plus utilisées pour évaluer la fiabilité d'un système. Bien que très rapide, cette approche a recourt à des simplifications pour les systèmes complexes conduisant inévitablement à des résultats qui omettent certains paramètres significatifs des réseaux étudiés, mais surtout qui négligent certaines actions prises dans son fonctionnement normal (automatismes, protections, etc.). Ci-dessous sont présentées les principales techniques communément utilisées pour évaluer la fiabilité d'un système.

Calcul de la probabilité d'indisponibilité (FOR)

Plusieurs indices de fiabilité calculés analytiquement découlent du traitement d'un ou plusieurs indices FOR (Force Outage Rate) qui évalue la probabilité qu'un équipement soit indisponible. Il se définit comme suit :

$$\text{Unavailability (FOR)} = U = \frac{\lambda}{\lambda + \mu} = \frac{\sum[\text{down time}]}{\sum[\text{down time}] + \sum[\text{up time}]} \quad (1.1)$$

où

λ = expected failure rate (taux de panne)

μ = expeced repair rate (taux de réparation)

m = mean time to failure (temps moyen de panne) = $1/\lambda$

r = mean time to repair (temps moyen de réparation) = $1/\mu$

Le calcul de l'indice FOR s'applique initialement à un modèle simple de deux états (disponible/indisponible) qui n'est valable que pour évaluer la fiabilité d'un ou plusieurs équipements sur une très longue période d'utilisation. Néanmoins, plusieurs variantes aux calculs de l'indice FOR permettent d'introduire d'autres états et leur transition, telles que la maintenance ou les périodes pendant lesquelles l'équipement est volontairement mis hors service (ex : réduction de la production).

Étude de la production

Le *Capacity Outage Probability Table* (COPT) est directement dérivé de l'évaluation du FOR des équipements à l'étude, mais appliqué uniquement à la production. Comme son nom l'indique, le COPT est un tableau réunissant uniquement les différents niveaux d'indisponibilité forcée et les probabilités que ces derniers soient atteints. Il n'inclut pas dans son traitement la charge, ni les systèmes de transport et/ou distribution. L'étude la fiabilité suivant cette approche n'adresse que le problème limité de la suffisance de l'approvisionnement. Elle a l'avantage de fournir une évaluation rapide de la fiabilité de la production. L'exemple qui suit présente le COPT d'un système totalisant 45MW de production, composé des unités suivantes :

- (a) 10 MW (FOR = 0.02)
- (b) 15 MW (FOR = 0.03)
- (c) 20 MW (FOR = 0.05)

Tableau 1.1 : *Capacity Outage Probability Table (COPT)*

Capacité hors service (C_i) (MW)	Probabilité $P(C_i)$	
0	$0.98 \times 0.97 \times 0.95$	= 0.90307
10	$0.02 \times 0.97 \times 0.95$	= 0.01843
15	$0.98 \times 0.03 \times 0.95$	= 0.02793
20	$0.98 \times 0.97 \times 0.05$	= 0.04753
25	$0.02 \times 0.03 \times 0.95$	= 0.00057
30	$0.02 \times 0.97 \times 0.05$	= 0.00097
35	$0.98 \times 0.03 \times 0.05$	= 0.00147
45	$0.02 \times 0.03 \times 0.05$	= 0.00003
$\sum P(C_i)$		= 1.00000

La probabilité de subir une perte de capacité de 0 MW est donc le produit des probabilités de chaque unité d'être disponible :

$$P(0\text{MW}) = (1 - 0.02) \times (1 - 0.03) \times (1 - 0.05) = 0.90307; \quad (1.2)$$

La probabilité de subir une perte de capacité de 10 MW est le produit des probabilités des unités de 15 et 20 MW d'être disponibles multiplié par la probabilité que le groupe de 10 MW soit indisponible :

$$P(10\text{MW}) = (0.02) \times (1 - 0.03) \times (1 - 0.05) = 0.01843; \quad (1.3)$$

La probabilité de subir une perte de capacité de 25 MW est le produit de la probabilité que l'unité de 10 MW soit disponible par les probabilités des unités de 10 et 15 MW d'être indisponibles :

$$P(25\text{MW}) = (0.02) \times (0.03) \times (1 - 0.05) = 0.00057 \quad (1.4)$$

Loss Of Load Probability (LOLP)

L'indice LOLP est une variante du COPT qui inclut dans son évaluation la perte de charge résultant de l'indisponibilité d'unité de production. Il n'est pas très répandu dans l'analyse de la fiabilité d'un système, car il évalue simplement la probabilité de subir des pertes de charge pour une charge fixe, qui ne varie pas dans le temps. Mais comme son prédecesseur, il a l'avantage de fournir une estimation rapide de la fiabilité d'un système.

$$LOLP = \sum_{i=1}^n P_i (C_i < L) = \sum_{k=1}^n p_k \quad (1.5)$$

C_i = capacité disponible

L = charge totale

$P_i(C_i < L)$ = probabilité que la charge totale soit supérieure à la production.

Si le système de production précédent alimente une charge $L=30$ MW, le LOLP sera simplement la somme des probabilités de voir la production passer sous les 30MW :

Tableau 1.2 : *Loss Of Load Probability (LOLP)*

Capacité hors service (C_i) (MW)	Probabilité $P(C_i)$	Perte de charge	Probabilité $P_i(C_i < L)$
0	0.90307	Non	
10	0.01843	Non	
15	0.02793	Non	
20	0.04753	Oui	= 0.04753
25	0.00057	Oui	= 0.00057
30	0.00097	Oui	= 0.00097
35	0.00147	Oui	= 0.00147
45	0.00003	Oui	= 0.00003
$\text{LOLP} = \sum_{i=1}^n P_i (C_i < L) =$			0.05057

Le système de production a donc une probabilité de 5% de subir une perte de charge

Loss Of Load Expectation (LOLE)

L'indice LOLE (Loss Of Load Expectation) est l'indice de fiabilité le plus utilisé. Il couple le calcul des probabilités de perte de production à la courbe de charge que le réseau alimente. Le résultat permet d'évaluer le risque, c'est-à-dire la probabilité que le système subisse des pertes de charge, car contrairement au LOLP, l'indice LOLE tient compte de la variation de la demande dans le temps. Son atout réside également dans la simplicité du calcul.

$$LOLE = \sum_{i=1}^n P_i(C_i - L_i) = \sum_{k=1}^n p_k t_k \quad (1.6)$$

C_i = capacité disponible

L_i = prévision de pointe de charge

$P_i(C_i - L_i)$ = probabilité de perte de charge.

t_k = unité de temps durant laquelle la capacité disponible ne répond pas à la charge L_i .

Le tableau 1.3 ci-dessous présente les résultats obtenus pour le LOLE considérant une courbe de charge linéaire ayant son maximum à 30 MW et son minimum à 10 MW appliqué sur le même système de production de 45 MW utilisé dans les exemples précédents :

Tableau 1.3 : *Loss Of Load Expectation (LOLE)*

Capacité hors service (C_i) (MW)	Probabilité $P(C_i)$	Temps t_k (%)	$P(C_i - L_i)$ (%)
0	0.90307	× 0	
10	0.01843	× 0	
15	0.02793	× 0	
20	0.04753	× 25	= 1.18825
25	0.00057	× 50	= 0.02850
30	0.00097	× 75	= 0.07275
35	0.00147	× 100	= 0.14700
45	0.00003	× 100	= 0.00300
LOLE = $\sum_{i=1}^n P_i(C_i - L_i) =$			1.43950

Dans cette configuration, le système de production de 45 MW ne sera pas en mesure de répondre à la demande 1.4395% du temps, soit 5.2542 jours / année.

Étude des réseaux de distribution

Les indices précédents traitent le problème de fiabilité entre la production et la charge, sans tenir compte de l'indisponibilité des réseaux de transport et distribution. L'étude de la fiabilité des réseaux de distribution est souvent omise des calculs et reçoit peu d'attention. Mais, la tendance change, car les distributeurs sont de plus en plus tenus d'assurer un niveau élevé de qualité de livraison de l'électricité.

L'étude de la fiabilité des réseaux de distribution se concentre surtout sur les systèmes radiaux. Suivant cette configuration, une charge est alimentée seulement si tous les composants entre elle et la source sont en opération. Le taux d'indisponibilité du système λ_s est donc simplement la somme des taux d'indisponibilité λ_i de chaque composant en amont de la charge. À partir de ce constat, plusieurs indices de fiabilité sont calculés à posteriori à partir de statistiques de pannes, dont :

$$\lambda_s : \text{average failure rate} = \sum_i \lambda_i \quad (1.7)$$

$$U_s : \text{Average annual outage time} = \sum_i \lambda_i r_i \quad (1.8)$$

$$SAIFI : \text{System average interruption frequency index} = \frac{\sum_i \lambda_i N_i}{\sum_i N_i} \quad (1.9)$$

$$SAIDI : \text{System average interruption duration index} = \frac{\sum_i U_i N_i}{\sum_i N_i} \quad (1.10)$$

$$CAIDI : \text{Customer interruption duration index} = \frac{\sum_i U_i N_i}{\sum_i \lambda_i N_i} \quad (1.11)$$

où N_i et r_i sont respectivement le nombre de clients connectés et temps moyen de réparation.

Contrairement aux indices COPT, LOLP et LOLE, la production n'entre pas dans l'évaluation de la fiabilité des réseaux de distribution et les calculs se limitent uniquement aux réseaux radiaux. L'étude des réseaux maillés n'est donc pas possible avec ces techniques simples.

Les indices de fiabilité des systèmes composites

Pour les systèmes combinant production, transport et/ou distribution, l'approche traditionnelle préconisée est la méthode de probabilité conditionnelle. L'objectif est donc d'évaluer la probabilité de subir une perturbation majeure suite à l'indisponibilité d'un ou plusieurs composants d'un même réseau (production/transport/distribution). De façon plus générale, si l'occurrence d'un événement A est dépendant de plusieurs événements B_i mutuellement exclusif, alors :

$$P(A) = \sum_i^n P(A|B_i)P(B_i) \quad (1.12)$$

Cette approche s'applique facilement à un système simple radial, mais le nombre de conditions croit rapidement pour les systèmes maillés. Évaluer la fiabilité des systèmes maillés, aussi petit soit-il, suivant l'approche de la probabilité conditionnelle est complexe et requiert une maîtrise des concepts de base de probabilité et des notions de fiabilité de réseaux, car le nombre de possibilités à étudier croît exponentiellement avec le nombre d'équipements composant le réseau.

Pourtant, en suivant l'approche de la probabilité conditionnelle, il est possible de dériver une panoplie d'indices de fiabilité [8] très utiles pour interpréter les performances des systèmes combinant production, transport et/ou distribution. Ils permettent de rendre compte de la fiabilité des systèmes en tenant compte de la probabilité d'indisponibilité de toutes les composantes d'un même système, contrairement aux indices tels que LOLP et LOLE.

L'évaluation de la fiabilité des systèmes composites est un problème très complexe, car plus les réseaux à l'étude sont grands, plus le nombre de possibilités à analyser augmente. Et, au regard du nombre d'indices qu'il est possible de déterminer, l'étude de la fiabilité d'un système requiert une connaissance accrue des principes de base de probabilité et des équations qui définissent ces indices afin d'en faire une bonne interprétation. Il est nécessaire de comprendre leur signification, leurs limites, ce qu'ils incluent et excluent.

C'est d'ailleurs ce que l'auteur soulève dans son ouvrage : « *This area of power system reliability evaluation is probably the least developed and also one of the most complicated* ». Il

ajoute néanmoins: « *In view of the environmental, ecological, societal and economic constraints faced by most electric power utilities it is expected that this area will receive considerable attention in the next decade* ».

Conclusion sur les techniques analytiques

L'étude de la fiabilité des systèmes simples, tels que l'analyse de la production ou des réseaux radiaux, est réalisable avec un minimum d'effort en appliquant les techniques analytiques présentées antérieurement. Le constat est malheureusement bien différent pour les systèmes composites complexes et plus particulièrement les réseaux maillés de grande taille. Pour ces derniers, l'évaluation de la fiabilité représente un énorme obstacle tant le nombre de possibilités, de contraintes et de conditions devient grand et pratiquement ingérable. L'analyse de ces systèmes requiert maintes simplifications, qui biaisen un peu plus les résultats.

Mais avec les avancées informatiques réalisées et les contraintes en vigueur aujourd'hui, il y a lieu de reconnaître que l'approche analytique ne suffit plus à l'évaluation de la fiabilité des réseaux. D'autres moyens doivent prendre le relais pour dépasser les limites auxquelles le domaine fait face. Le recours aux outils de simulations figure parmi les solutions qui reçoivent le plus d'intérêt. C'est d'ailleurs la direction que nous préconisons dans l'outil APREM.

1.1.2 Simulations (Monte-Carlo)

L'approche par simulations évalue la fiabilité d'un réseau électrique en simulant son comportement ainsi que les événements aléatoires qui modifient ce réseau. Cette méthode traite le problème de calcul de fiabilité comme une longue série d'essais aléatoires sur un réseau dynamique.

Cette technique permet de prendre en compte une infinité de paramètres inhérents à la gestion, la composition et l'évolution du réseau électrique. Ces paramètres pourraient décrire une multitude d'événements aléatoires ayant une probabilité d'occurrence, telle que les pannes, les activités d'entretiens, la variation de la charge ou de la production, les différentes procédures de transfert sur le réseau, etc. En simulant un très grand nombre de fois le comportement probabiliste du réseau pour une période de temps donnée, il est donc possible de calculer les indices de fiabilité et d'identifier clairement le type d'anomalies récurrent sur le réseau et déduire les points faibles du système.

Simulation Monte-Carlo

L'exemple du lancé de dé illustre bien la méthode Monte-Carlo utilisée pour le calcul de la probabilité. Analytiquement, un dé est composé de 6 faces et si ce dernier est parfaitement équilibré, les chances d'obtenir le chiffre six (6) sont naturellement de 1/6.

La méthode de Monte-Carlo tente de faire la preuve du même raisonnement logique en réalisant plusieurs essais et diviser le nombre de fois que le chiffre six (6) est obtenu par le nombre d'essais réalisé afin de trouver la fréquence de succès. Lorsque le nombre d'essais est suffisamment élevé, le résultat tend vers la probabilité réelle d'obtenir le chiffre six (6). Cette probabilité s'écrit alors comme suit :

$$P(\text{d'obtenir chiffre six (6)}) = \lim_{n \rightarrow \infty} \left(\frac{S}{N} \right) \quad (1.13)$$

où S est le nombre de fois que le chiffre 6 est obtenu et N le nombre total d'essais réalisés.

La simulation de ce test logique peut être réalisée sur une multitude de logiciels de calculs. À titre d'exemple, les valeurs de probabilité d'obtenir le chiffre six (6) sont présentées graphiquement à la Figure 1.1 pour 3000 essais exécutés à l'aide de la fonction *random* de Microsoft Excel. Les trois plus importants constats qui en découlent sont :

- Un faible nombre d'essais procure une mauvaise estimation de la probabilité;
- La probabilité tend vers la valeur réelle plus le nombre d'essais est grand;
- Si le même exercice était repris plusieurs fois, les résultats et les courbes obtenus pour chaque simulation seraient différents les uns des autres. Cette méthode d'analyse ouvre donc la porte au traitement parallèle, car il n'y existe aucune interdépendance entre les résultats.

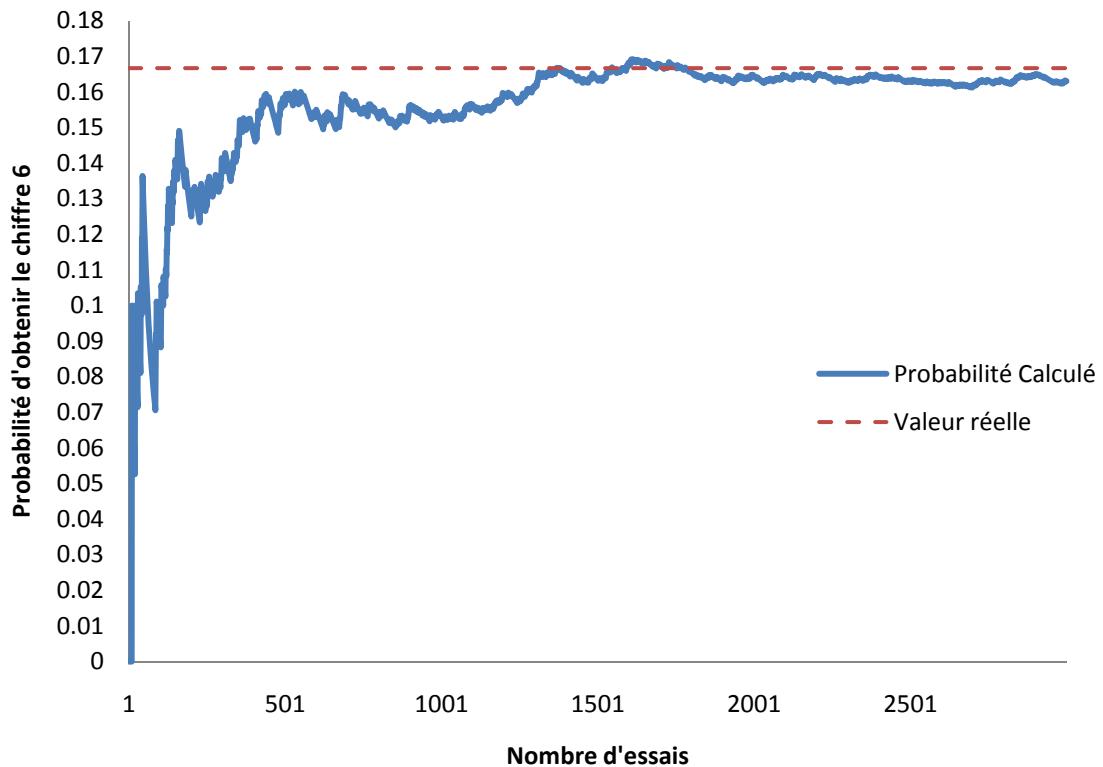


Figure 1.1 : Évolution de la probabilité avec le nombre de simulation

Application à l'étude des réseaux électriques

La méthode de Monte-Carlo est un outil puissant pour l'analyse des réseaux électriques, car elle permet de mettre en commun les caractéristiques variables et probabilistes des éléments des réseaux suivant un raisonnement simple et beaucoup plus accessible aux personnes novices en fiabilité des réseaux. Le même test logique du lancé de dés est alors applicable à un réseau radial ou maillé, de plusieurs dizaines ou centaines de nœuds, car l'essentiel du défi est de modéliser adéquatement les changements probabilistes sur chaque composant du réseau, exécuter une série d'essais et observer le comportement aléatoire du réseau. Tour à tour ou simultanément, et suivant une probabilité prédéterminée, chacun des composants du réseau changera d'état ou de caractéristique au fil des essais et chaque changement aura un impact, minime soit-il, sur le réseau. Comme le préconise l'analyse de type Monte-Carlo, plus le nombre d'essais sera grand, plus il sera possible d'identifier les événements dont la probabilité d'apparition est la plus élevée afin d'évaluer tangiblement les différents indices de fiabilité d'un système.

Bien que le concept soit relativement simple à implémenter, l'ampleur du temps de calcul nécessaire à la réalisation d'étude Monte-Carlo a longtemps freiné l'intérêt suscité pour cette technique et son application aux réseaux électriques. Si 3000 essais sont requis pour trouver la probabilité d'obtenir le chiffre 6 lors d'un lancé de dé, il est alors relativement facile d'imaginer le nombre d'essais nécessaire pour déduire des valeurs justes et tangibles pour un grand réseau. Mais aujourd'hui, des techniques sont disponibles pour réduire le temps de calcul [9] [10] et compte tenu des outils et ressources disponibles, la méthode dite de Monte-Carlo est de plus en plus populaire dans l'analyse des réseaux électriques, car elle ouvre les portes à de nouveaux axes d'études.

Les techniques utilisées

Trois techniques sont principalement utilisées dans l'évaluation de la fiabilité des systèmes composites par la méthode de Monte-Carlo [11] :

- *State sampling technique*
- *State transition sampling technique*
- *Sequential method*

Dans la première, aucun lien de cause à effet entre les états de chaque composant n'est pris en compte. La probabilité de rencontrer les différents états enregistrés pour chaque composant d'un réseau est uniforme. L'analyse porte uniquement sur un système d'états échantillonné hors de toute chronologie.

À l'inverse, la seconde approche tient compte de la transition entre les états du système dans son ensemble (et non la transition des états pour chaque composant). Les liens de cause à effet et les transitions entre les différents états du réseau sont définis globalement pour le système.

Finalement, la dernière approche tient compte de la séquence des états de chaque composant et les liens de cause à effet peuvent être définis à tous les niveaux. Il est donc possible d'introduire une chronologie dans les événements. La technique séquentielle est naturellement plus lourde en termes de calcul, mais elle permet de combiner et modéliser dans les détails toute

sorte de contingences et caractéristiques d'opération d'un réseau (variation de la charge, procédures d'opération, calendrier de maintenance, etc.).

L'analyse de la fiabilité dans le cadre de la présente recherche ne doit donc pas se limiter à l'une des trois méthodes. La pertinence et la justesse de l'information disponible sur les réseaux électriques étudiés détermineront le choix des méthodes et le niveau de détail qui sera possible d'atteindre.

La pertinence de la méthode de Monte-Carlo

L'approche de Monte-Carlo offre la possibilité d'intégrer également de nouveaux paramètres souvent omis dans l'étude de la fiabilité.

À titre d'exemple, d'après une étude réalisée par le NERC (*North American Reliability Council*), les systèmes de protection sont mis en cause dans 75% des perturbations majeures. Il est effectivement intéressant de noter que la grande majorité des études de fiabilité ne tient pas compte de la fiabilité et de l'impact des pannes de ces systèmes de protection [12]. Les composants de second plan des réseaux électriques, tel que les relais de protection, les systèmes auxiliaires, les systèmes de communication sont généralement considérés infaillibles. Malheureusement, l'analyse des *blackouts* vécus ces dernières années les met souvent en cause.

Ces systèmes de second plan sont au cœur de l'évolution technologique des réseaux électriques. De nouvelles fonctionnalités leur sont attribuées afin d'automatiser les réseaux et rendre ces derniers plus intelligents. Cette évolution modifie grandement la composition et les méthodes d'opération des réseaux électriques et ces changements affectent la fiabilité des réseaux car c'est d'ailleurs leur raison d'être. Les recherches s'intensifient à juste titre afin de prendre en compte ces nouveaux facteurs opérationnels dans l'analyse de la fiabilité des réseaux afin d'y inclure les contraintes liées à la planification de la maintenance, la gestion de la production et l'opération du réseau [13].

Force est de reconnaître que tous ces aspects sont difficiles à inclure dans les méthodes analytiques de fiabilité. Pour que l'outil d'analyse puisse s'adapter à n'importe quel champ d'études, il est donc primordial de pouvoir intégrer la méthode de Monte-Carlo dans le cadre de la fiabilité. Cette méthode offre plus de possibilités d'études qui répondent mieux à la composition et aux caractéristiques des réseaux. Grâce à la flexibilité qu'elle apporte, l'approche

de Monte-Carlo se prête beaucoup plus facilement à l'étude de l'évolution des réseaux dans le temps.

Étude de la fiabilité et l'analyse paramétrique des réseaux électriques

L'outil de recherche développé dans le cadre du présent projet répond parfaitement aux prérequis nécessaire à l'élaboration de simulations de types Monte-Carlo. Comme discuté précédemment, l'étude de fiabilité des réseaux électriques selon la méthode de Monte-Carlo consiste essentiellement à simuler en boucle le comportement probabiliste d'un réseau électrique en y intégrer des changements d'état ou de paramètres dictés par une probabilité définie. L'essentiel de l'analyse consiste alors à enregistrer la récurrence des événements pertinents à l'étude de la fiabilité tout au long du processus. On se rappellera alors que, si le nombre d'essais est suffisamment grand, la probabilité de rencontrer un évènement est le ratio du nombre de fois que l'évènement est rencontré sur le nombre total d'essais réalisés (équation 1.13).

Les études de fiabilité des réseaux électriques selon la Monte-Carlo sont donc à toutes fins pratiques des études paramétriques pour lesquelles l'exécution des algorithmes de changement d'état et de paramètres de réseaux respectent les probabilités définies par l'usager afin qu'ils reproduisent le comportement probabiliste de chaque équipement. L'utilisateur peut ainsi user des fonctions disponibles de tirage au sort pour déclencher des changements sur le réseau et simuler des évènements aléatoires comme l'indisponibilité d'un équipement, sa dégradation ou tout autre perturbation sur le réseau.

La méthode de Monte-Carlo en elle-même ne se limite pas au calcul d'un seul et unique indice de fiabilité. Elle permet au contraire de faire l'évaluation de tous les indices cités dans cette section et plus encore. La théorie longuement présentée dans ce chapitre doit servir de support à l'utilisateur qui souhaite évaluer un de ces indices de fiabilité au moyen de la méthode de Monte-Carlo. L'utilisateur doit avant tout comprendre la théorie derrière chaque indice de fiabilité afin de bâtir convenablement l'étude paramétrique permettant de le calculer selon la méthode de Monte-Carlo.

À titre d'exemple, l'utilisateur qui cherche à évaluer le LOLE d'un système complexe suivant l'approche de Monte-Carlo, doit nécessairement paramétrier dans son modèle :

- la variation de la charge sur l'ensemble du système (suivant une courbe de probabilité ou courbe de charge);
- les changements d'état de chaque unité de production du système;
- le délestage qui assure la convergence du système en maintenant la charge du réseau inférieur à la production.

L'outil de recherche proposé dans ce mémoire permet, à juste titre, à l'utilisateur de paramétrier toutes ces fonctions et réaliser, comme requis par la théorie de Monte-Carlo, un très grand nombre de simulations sans aucune intervention de l'usager durant le processus.

Il est même intéressant de préciser que le calcul théorique de l'indice LOLE ne tient pas compte des indisponibilités pouvant survenir sur les équipements de transport et distribution, ni des relations de causes à effet qui pourraient être introduites dans le modèle. Ce sont là des paramètres supplémentaires qui peuvent être pris en compte dans l'évaluation de la fiabilité des systèmes à travers la méthode de Monte-Carlo.

Cependant la méthodologie pour évaluer les différents indices de fiabilité au moyen de l'outil de recherche proposé n'est pas couverte par le présent mémoire et pourra faire l'objet de travaux ultérieurs. Ce chapitre a pour but essentiellement de présenter la possibilité de coupler l'analyse paramétrique à l'évaluation de la fiabilité des systèmes suivant la méthode de Monte-Carlo et familiariser le lecteur à ce domaine en pleine expansion.

1.2 Choix de l'algorithme de calcul de répartition de puissance

Écoulement de puissance versus étude transitoire

Comme décrit dans les pages qui précèdent, la présente recherche s'intéresse à l'évolution des paramètres et de la topologie des réseaux dans le but avoué de faire l'étude de la fiabilité des systèmes. Les changements sur les paramètres des équipements, mais également la connectivité entre les éléments des réseaux électriques, influencent les conditions d'opération de ces derniers et l'objectif de la recherche est justement de déduire la probabilité de dépasser les limites d'opération des réseaux étudiés. Ainsi pour saisir l'impact des modifications qui s'opèrent sur les réseaux, l'outil de recherche doit intégrer au moins un algorithme d'écoulement de puissance afin de réaliser quelques exercices de validation de concept. L'ajout d'une sélection de méthode de calculs de répartition de puissance pour répondre aux mieux aux spécificités propres de chaque réseau pourra faire l'objet de développements futurs de l'outil de recherche.

Bien qu'il existe plusieurs formulations pour solutionner l'écoulement de puissance d'un réseau électrique, ces dernières peuvent être regroupées suivant trois types, soit les méthodes de Gauss-Seidel, de Newton-Raphson et *Backward/forward sweep*. La méthode *Backward/Forward sweep* est la populaire pour sa rapidité dans le traitement des réseaux de distribution faiblement maillés. Ce type d'algorithme est d'ailleurs à la base du solveur itératif d'écoulement de puissance du logiciel CYMDIST [14-15]. Néanmoins, les méthodes de Newton Raphson formulées sur la base des relations d'injections de courant tendent à se démarquer par leurs performances dans le traitement de larges systèmes maillés. Les méthodes de Gauss-Seidel, quant à elles, sont moins robustes et convergent généralement plus difficilement [16].

Le nouvel outil de simulation EMTP-RV [17] utilise justement une formulation basée sur les relations de courant qui diffèrent des algorithmes de Newton-Raphson (NR) usuellement employés dans les logiciels d'analyse de réseaux. Cette formulation permet de décrire les systèmes sous une forme matricielle très simple qu'on appellera matrice nodale-augmenté-modifiée. Elle s'appuie sur les relations de courants (Loi de Kirchoff) au lieu des relations de puissances, ce qui simplifie la modélisation des composants de réseaux, mais aussi la représentation des connexions entre les composants (delta, triangle, phase-phase, etc.). La technique permet ainsi de déduire la matrice Jacobienne sous une forme simplifiée et toutes les

matrices du système linéaire résultant sont creuses. L'expression du système est en coordonnées rectangulaires (réel/imaginaire) et les dérivées partielles sont alors exprimées par rapport aux tensions et courants du système et de ses composants [18-19].

Cette méthode est aussi de modéliser les réseaux à partir de leurs données réelles sans avoir à exprimer les valeurs en « per-unit ». Il est alors possible de modifier les ratios de transformation, mais également les tensions d'opération des systèmes sans avoir à réévaluer la matrice du réseau et ses admittances en fonction des nouvelles tensions de base du système, comme c'est le cas lorsqu'on travaille en per-unit.

Cette formulation permet également d'imbriquer des points de sectionnements et leur état (ouvert/fermé) directement dans la matrice du réseau. La technique fort simple consiste à créer une égalité entre les tensions et les courants des deux nœuds qui se connectent lorsque l'équipement de sectionnement, si ce dernier est en position fermé. Cette égalité devient alors nulle lorsque le circuit s'ouvre. Il est ainsi possible de changer la topologie du réseau en modifiant seulement quelques valeurs à l'intérieur de la matrice du réseau afin de créer ou éliminer cette égalité. Contrairement aux formulations les plus courantes de NR, celle-ci n'a donc pas recours à aucune méthode pour gérer l'ajout et le retrait des nœuds et, conséquemment, la reconstruction de la matrice du réseau pour tenir compte de l'ouverture ou la fermeture d'un circuit. Ainsi, La matrice nodale-augmentée-modifiée n'est donc pas reformulée à chaque changement topologique.

Aussi, la formulation étant basée sur les relations de courant, elle donne un accès direct aux valeurs de courants de tous les composants du réseau ce qui évite d'avoir à les déduire à partir des puissances calculées à chaque nœud selon la méthode standard.

Parmi les nombreux autres avantages, ces trois caractéristiques de la formulation utilisée dans EMTP-RV permettent de réduire significativement les étapes de prétraitement des données et déduire plus rapidement la matrice nodale-augmentée-modifiée du réseau correspondant.

L'algorithme permet ainsi de traiter les systèmes complexes multi-phasés et faire l'analyse de n'importe quelles configurations de réseaux, en autant que le système d'équations soit bien posé. Il est alors possible de modéliser les réseaux dans les moindres détails, en tenant compte des spécificités des connexions.

Aussi, l'intérêt de l'approche EMTP-RV est qu'elle est très simple à implémenter au niveau logiciel. Sa mise en œuvre peut s'avérer relativement souple, surtout à l'intérieur de langage orienté sur le traitement des matrices, tel que Matlab. Chacun des composants du réseau occupe un espace bien spécifique dans la matrice Jacobienne nodale-augmentée-modifiée et cette architecture évite alors d'avoir à réévaluer certaines dérivées partielles à chaque itération, notamment les dérivées partielles des composants linéaires, des sources de tension et des transformateurs. L'expression de ces dérivées partielles n'étant pas fonction de la tension, ni du courant, leurs valeurs sont constantes. Un gain non négligeable peut ainsi être réalisé dans la vitesse d'exécution du logiciel puisque ce sont seulement quelques sous-matrices de la Jacobienne qui doivent être réévaluée à chaque itération.

Mis à part les performances de la méthode EMTP-RV, l'utilisation de cette dernière ouvre entre autres, la porte aux études dynamiques et transitoires des systèmes. Cette extension future du logiciel peut s'avérer intéressant si l'étude des réseaux devait mesurer l'impact des systèmes de protection sur la fiabilité des réseaux. Bien que ce ne soit pas un objectif à court terme, il n'en demeure pas moins que cette possibilité est présente et répond à un intérêt grandissant motivé par l'intégration des nouvelles technologies sur les réseaux électriques.

1.3 Logiciel commerciaux

Logiciels commerciaux

Il existe plusieurs logiciels commerciaux dédiés exclusivement à l'analyse de réseaux électriques. Parmi les plus populaires en Amérique du Nord, les logiciels PSS/E [20], CYMDIST [21], PSAF [22] et EMTP-RV [23] jouissent d'une très grande notoriété dans l'industrie de l'énergie et chacun est maître de son expertise. PSS/E est l'outil de simulation et de planification de prédilection des transporteurs tandis que CYMDIST couvrent le domaine des réseaux de distribution et PSAF, les réseaux de transport et industriels. EMTP-RV est un engin de calcul puissant initialement dédié à l'étude approfondie des phénomènes transitoires électromagnétiques, comme l'indique son acronyme. Tous partagent l'engouement d'offrir des solutions complètes pour l'analyse des réseaux et d'assister l'utilisateur dans la modélisation de son réseau à travers une interface graphique adaptée. Le large déploiement de ces logiciels au sein des grandes utilités électriques prouve leur robustesse et efficacité. Malheureusement, l'accès aux codes sources de ces programmes étant impossible, il est parfois ardu de modifier ou d'ajouter de nouveaux algorithmes aux logiciels afin de prototyper et personnaliser certaines études. Bien que ces logiciels mettent à la disposition des utilisateurs des scripts qui permettent de s'affranchir des restrictions imposées par leurs interfaces et envoyer des commandes directement au code source des logiciels, l'utilisation de ces scripts requiert bien souvent des aptitudes avancées en programmation. Cette contrainte ne concorde avec les besoins de flexibilité inhérents aux activités académiques et de recherche. Pour les besoins spécifiques de la présente recherche, l'effort d'apprentissage et de développement, les habiletés et le niveau de détails requis dans la manipulation et l'adaptation de ces logiciels « commerciaux » n'est pas pleinement justifié au stade actuel du développement de l'outil d'analyse paramétrique des réseaux électriques.

Logiciels « open source »

Une des alternatives aux logiciels commerciaux est le recours aux logiciels « open source » pour lesquels les codes sources sont accessibles et libres de tout droit de propriété. Ces derniers sont généralement dédiés à des domaines précis et se limitent à des calculs plus spécifiques. Néanmoins, les logiciels à code ouvert, tous domaines scientifiques confondus, sont très peu

documentés. Ces contraintes nécessitent le recours à des ressources expérimentées en programmation, ce qui alourdit les étapes de développement de nouveaux modules de calcul et l'adaptation de ces logiciels à la présente recherche.

Logiciels développés sur Matlab

Depuis la dernière décennie, plusieurs logiciels de programmation de haut niveau font leur entrée dans le domaine scientifique pour contrecarrer les difficultés qu'éprouvent les chercheurs dans la définition et l'exécution d'algorithmes mathématiques avancées dans des langages de programmation tel que C, C++, Java et Fortran. Matlab, Mathematica et Modelica, pour ne citer que ces trois plateformes de développement, offrent ainsi des outils de programmation spécialisés dans le calcul et le développement d'algorithmes mathématique particulièrement intéressants pour la recherche scientifique et le milieu académique. Le langage évolué de ces applications, dont la syntaxe se rapproche habilement de l'écriture scientifique, supprime l'effort d'apprentissage et de programmation des utilisateurs.

Contrairement à plusieurs concurrents, Matlab a su s'implanter dans une très grande variété de domaines (énergie, traitement d'images, financier, etc.) grâce aux différents « ToolBox » mis à la disposition des utilisateurs désireux d'utiliser des fonctions prédéfinies propres à chaque expertise. Le large éventail d'utilisateurs prouve que le produit est très répandu dans la communauté scientifique, tous domaines confondus.

Cet atout est intéressant pour la recherche visée, car il permet la contribution de diverses expertises dans l'étude des réseaux électriques, soit statistique, économique, financière, etc. Et, c'est à juste titre que plusieurs développeurs ont déjà choisi Matlab pour monter leur logiciel d'analyse de réseau dont Power System Toolbox (PST) [24], MatPower [25], Voltage Stability Toolbox (VST) [26], MatEMTP [27], SimPowerSystems (SPS) [28], Power Analysis Toolbox (PAT) [29], Educational Simulation Tool (EST) [30] et Power System Analysis Toolbox (PSAT) [25]. Parmi ceux-là, MatPower, VST et PSAT sont des logiciels « open sources » disponibles gratuitement [31].

En outre, en 1997, deux influents chercheurs du domaine des simulations de réseau publiait dans [27] une nouvelle avenue de développement du logiciel EMTP sur la plateforme Matlab en créant le simulateur MatEMTP pour l'analyse transitoire des réseaux électriques, car

selon eux « this new approach to software engineering can afford a dramatic coding simplification for sophisticated algorithmic structures » [32]. Ils accusent entre autres l'utilisation de langage de bas niveau d'être à l'origine du faible taux de renouvellement et d'optimisation du code du logiciel EMTP-RV, programmé en FORTRAN-77. De leur expérience, très peu de lignes de code sont requises pour réaliser le logiciel MatEMTP sur Matlab comparativement à sa version originale EMTP et les résultats obtenus sont identiques à ceux obtenus en utilisation la version originale de EMTP. La simplicité du traitement des matrices, les fonctions graphiques et la gestion de données représentent un atout majeur aux langages conventionnels. « It also eliminates EMTP modelling limitations through a less restrictive formulation of main network equations ».

Néanmoins, EMTP reste en général 2.5 fois plus rapide que sa version Matlab. Mais, encore là, Matlab offre la possibilité de précompiler certaines fonctions en langage C (MEX-files) [33] et supporte également le traitement en parallèle sur multiprocesseurs [34] afin d'améliorer significativement les performances des programmes développés sur sa plateforme. En tenant compte des capacités des ordinateurs d'aujourd'hui, les contraintes de temps et vitesse de calcul peuvent être compressées en fonction des enjeux liés à l'application.

Conclusion

Pour les applications où l'utilisateur est appelé à intervenir directement sur le code pour créer et personnaliser ses propres algorithmes, il est plus simple et cohérent d'utiliser un langage spécialisé de haut niveau facilement interprétable et modifiable par l'ensemble des utilisateurs. Ceci est encore plus vrai pour la recherche visée puisque le développement sera réalisé en grande partie en milieu académique par des ressources ayant des expertises diverses et peu familier aux langages de bas niveau. Et comme discuté plus haut, Matlab s'avère être la plateforme de développement la mieux outillée pour les besoins de la présente recherche. Son utilisation facilitera son renouvellement et ses développements futurs.

1.4 Matlab orienté objet

Retour sur Matlab

Le choix d'utiliser Matlab pour le développement d'application d'analyse paramétrique des réseaux électriques est encouragé, par le grand nombre de logiciels déjà disponibles sur Matlab pour l'analyse des réseaux électriques : Power System Toolbox (PST), MatPower, Voltage Stability Toolbox (VST), MatEMTP, SimPowerSystems (SPS), Power Analysis Toolbox (PAT), Educational Simulation Tool (EST) et Power System Analysis Toolbox (PSAT). Chacun d'eux offre des fonctions d'analyses de réseaux électriques tout aussi performantes et utiles les unes que les autres et certains sont agrémentés dans certains cas d'une interface graphique destinée à faciliter la modélisation des réseaux électriques à l'étude. Alors, pourquoi ne pas construire le nouvel outil d'analyse paramétrique des réseaux électriques à partir d'un de ces logiciels?

L'objectif : étude paramétrique

La plupart des logiciels d'analyse de réseaux cherchent à offrir une solution complète pour solutionner des réseaux dont les paramètres et la topologie sont généralement invariants . Ainsi, s'il faut étudier, à titre d'exemple, l'augmentation d'une charge sur plusieurs années couplée à la variation de la production et des changements topologiques sur le réseau, l'usager doit prédefinir tous ses scénarios ; s'il y a 60 scénarios à étudier, l'usager réalisera 60 simulations. Cette contrainte explique l'engouement pour développer des outils qui faciliteront la programmation d'études paramétriques des réseaux électriques afin d'automatiser l'évolution des réseaux électriques simulés. Simplifier la programmation des études paramétriques rend du même coup plus accessible l'étude du comportement probabiliste des réseaux via la méthode de Monte-Carlo ou d'autres méthodes, alors que les changements sur le réseau seraient dictés par des probabilités.

Orienté Objet dans Matlab

Dans ses plus récentes versions, le logiciel propose une série d'outil et des fonctions intégrées pour faciliter la programmation orientée objet [35] dans son environnement. Bien que de tels outils de développement existe depuis longtemps dans d'autres langages (C, C++, Java,etc.), la syntaxe proposée par Matlab garde à très haut niveau le langage de programmation et permet ainsi à l'utilisateur final d'être en même temps le développeur/programmeur de son application. Il jouit alors des énormes capacités mathématiques de Matlab pour développer ses algorithmes et

fonctions mathématiques les plus complexes, mais également d'un environnement simple et accessible pour créer de nouvelles classes répondant à ses besoins avec toutes les notions d'héritage requises pour appliquer une méthodologie orientée objet.

Pour les objectifs visés par le présent projet de recherche, la plateforme limite les difficultés de développement et de renouvellement du code en rendant plus accessible les activités de programmation et de recherche liée à l'outil. Une migration future de l'outil d'analyse paramétrique ou d'une partie du code vers des langages plus performants peut être envisagée dans des travaux futurs afin d'optimiser et améliorer les performances de ce dernier.

Le développement de l'outil d'analyse paramétrique des réseaux électrique dans Matlab a cependant suivi la méthodologie décrite dans [36] afin d'assurer la portabilité du code et son utilisation dans les versions antérieurs de Matlab.

CHAPITRE 2 APPROCHE MÉTHODOLOGIQUE

Revue de littérature

Le présent projet de recherche regroupe plusieurs champs d'intérêt à savoir la programmation, les algorithmes d'écoulement de puissance et le calcul de fiabilité.

La première étape du projet de recherche s'attarde sur la revue de la littérature afin d'aborder le développement de l'outil d'analyse paramétrique des réseaux électriques :

- Une revue des différentes techniques utilisées pour l'évaluation de la fiabilité des réseaux électriques. Contrairement aux méthodes analytiques difficilement programmables, le calcul de la fiabilité à l'aide de la méthode de Monte-Carlo est simple, efficace et se couple bien avec les calculs d'écoulement de puissance. La méthode de Monte-Carlo nécessite néanmoins un très grand nombre de simulations pour converger vers des résultats tangibles. L'un des objectifs de l'application est d'assurer une grande souplesse dans la programmation des boucles de calcul.
- La seconde réflexion porte sur les algorithmes de calculs d'écoulement de puissance disponible. Puisque notre recherche s'intéresse à la topologie des réseaux, l'algorithme de Newton-Raphson s'impose, car il est souvent plus robuste que les autres algorithmes lors d'étude de grands réseaux complexes et maillés.
- Pour la représentation mathématique des réseaux, notre recherche s'est concentrée sur la méthode des matrices nodales-augmentées-modifiée utilisée dans le logiciel EMTP-RV. Cette technique assure une très grande flexibilité dans la manipulation des matrices et donc des paramètres des réseaux. Cependant, les développements futurs pourront bien entendu aborder l'intégration d'autres algorithmes de calculs sans se restreindre à ce choix technique.

Développement du prototype I (choix du langage)

Suite à cette revue de l'état de l'art, le choix du langage de programmation se confirme et le premier prototype de l'outil d'analyse paramétrique est créé sur Matlab. Ce logiciel propose un environnement de programmation de haut niveau orienté sur le développement d'algorithmes

mathématiques et offre des capacités non négligeables pour le traitement des matrices. Le langage est intuitif et le logiciel est très répandu dans la communauté scientifique.

En s'appuyant sur la méthode de représentation matricielle utilisée dans le logiciel EMTP-RV [17] et décrite dans [18-19], l'algorithme d'écoulement formulé à partir des relations de courant et des matrices nодales-augmentées-modifiées est développé dans un premier prototype afin de solutionner des rseaux ´electriques dcrits dans un fichier texte suivant un gabarit pre tabli (commun ement appell  Netlist dans la plupart des logiciels d'analyse de rseau disponibles).  cette tape, l'outil permet de raliser des tudes param triques, mais requiert une connaissance accrue de Matlab et des op rations internes effectu es par l'outil.

Acquisition des comp tences en programmation orient  objet sur Matlab

Face  la complexit  du code n cessaire  la programmation d'tudes param triques dans le prototype I, la possibilit  de repr senter sous forme d'objets Matlab tous les composants des rseaux ´electriques est pleinement tudi e. La cr ation de nouvelles classes dans Matlab suivant une m thodologie orient e objet permet de d finir  souhait les m thodes d'acc s aux param tres des composants du rseau et simplifier les op rations de param trage. Cette tape de recherche aboutit  l'acquisition des comp tences en programmation orient e objet sur Matlab qui m nera par la suite au d veloppement du second prototype.

D veloppement du Prototype II (conversion vers la programmation orient e objet)

Une fois la programmation orient e objet sur Matlab maitris e, le prototype II est r alis  en b atissant les nouvelles classes d'objets repr s tant les l ments de base des rseaux ´electriques. L'algorithme d'écoulement de puissance et les multiples fonctions d velopp es dans le premier prototype sont adapt s pour permettre l'utilisation des nouveaux objets cr  s.

Tests sur le prototype II

Pour confirmer les solutions utilis es dans le deuxi me prototype de l'outil d'analyse param trique des rseaux ´electriques, ce dernier est soumis  des tests et compar  avec diff rents logiciels commerciaux pour valider les r sultats obtenus.

Les r sultats sont tr s convaincants et concluants pour le prototype II. Les simulations r alis es avec l'outil d'analyse param trique sont compar es aux r sultats d'écoulement de

puissance obtenus sur deux (2) différents logiciels commerciaux reconnus par l'industrie, soit PSS/E et PSAF .

Le prototype est également mis à contribution dans un cours de fiabilité des réseaux électriques donné à des étudiants de 2^{ième} cycle. Il a été utilisé par un groupe d'étudiant afin d'en évaluer la convivialité et valider la possibilité de réaliser des calculs de fiabilités selon la méthode de Monte-Carlo sur des réseaux électriques. Plusieurs calculs de fiabilité ont été réalisés dans le cadre du cours confirmant ainsi que l'outil s'adapte très bien aux études de type Monte-Carlo. L'outil permet d'assigner toute sorte d'évènements probabilistes sur le réseau et mesurer leur impact sur plusieurs points. D'après les commentaires dégagés, l'environnement est simple d'utilisation et les opérations nécessaires pour modifier les paramètres et la topologie des réseaux sont intuitives. Néanmoins, le temps de calcul reste un obstacle important à ce type d'étude.

Optimisation du prototype II

Suite à ces essais, plusieurs fonctions ont été optimisées, et de nouvelles classes créées. Le prototype II, l'outil d'Analyse Paramétrique des Réseaux Électriques sur Matlab (APREM), est fonctionnel et plusieurs fonctionnalités sont ajoutées pour assurer la robustesse de l'outil. Parmi ces fonctionnalités, un algorithme basé sur la théorie des graphes est intégré à l'outil de recherche afin de repérer les îlots électriques et les sources PV isolées connectées à un réseau dépourvu de barre d'équilibre. Bien que la méthode ne soit pas optimale, elle permet d'apporter automatiquement les correctifs essentiels au réseau afin d'assurer sa convergence des écoulements de puissance indépendamment de la validité de sa topologie initiale.

Cette étape est cruciale dans les études paramétriques et de fiabilité, car des milliers de scénarios sont générés et l'usager ne peut pas valider et corriger manuellement la topologie des réseaux à chaque fois que les écoulements de puissance ne convergent pas. La section 3.6.3 aborde la technique qui est mise en œuvre dans le prototype II pour valider la topologie des réseaux simulés.

CHAPITRE 3 CORPS DE L'OUVRAGE : OUTIL APREM

3.1 Matlab

Le prototype de l'outil d'analyse paramétrique est développé entièrement sur Matlab, d'où l'acronyme du logiciel APREM pour Analyse Paramétrique des Réseaux Électrique sur Matlab. Matlab propose un environnement de programmation de haut niveau orienté sur le développement d'algorithmes mathématiques et offre des capacités très puissantes pour le traitement des matrices, notamment avec l'introduction des *cells arrays* [37] abondamment utilisées dans le développement de l'outil APREM. Le langage de programmation est intuitif et bien maîtrisé dans la communauté scientifique.

Matlab supporte le traitement des vecteurs et des matrices sur plusieurs formes (*matrix*, *array* et *cell*), ce qui représente un atout majeur pour le développement rapide d'algorithmes propres aux réseaux électriques souvent représentés sous forme matricielle. Matlab offre également dans ses plus récentes versions des outils pour faciliter la programmation en orienté objet. Ces nouvelles fonctionnalités rendent encore plus accessible le développement de logiciels dans l'environnement Matlab et contribueront à rapprocher l'utilisateur final de son rôle de développeur du prototype APREM.

3.2 Matrice nodale augmentée modifiée

La méthode des matrices-nodales-augmentées-modifiées décrite dans [17-19] permet de représenter les réseaux électriques multiphases déséquilibrés. Cette approche est utilisée notamment dans le logiciel EMTP-RV. Considérant les capacités de traitement des matrices de Matlab, cette solution s'adapte très bien à l'environnement de développement choisi et est reprise dans l'outil APREM, mais limitée pour le moment au traitement des réseaux équilibrés.

Les matrices nodales-augmentées-modifiées incluent les éléments non linéaires dans le modèle numérique matriciel des réseaux électriques. En s'intéressant uniquement à la représentation numérique des réseaux et sans tenir compte des contraintes d'écoulement de puissance, il est possible de modéliser un réseau dans sa totalité et dans les moindres détails avec les 5 éléments suivant :

- Les impédances,
- Les sources de tension idéales qui sont les barres d'équilibres des réseaux,
- Les sources de courant,
- Les transformateurs, et
- Les interrupteurs (*Switch*).

La matrice nodale-augmentée-modifiée [17-19] d'un réseau quelconque, peu importe sa composition, peut prendre la forme suivante :

$$\begin{bmatrix} \mathbf{Y}_n & \mathbf{V}_c & \mathbf{D}_c & \mathbf{S}_c \\ \mathbf{V}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{D}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}_r & \mathbf{0} & \mathbf{0} & \mathbf{S}_d \end{bmatrix} \times \begin{bmatrix} \mathbf{V}_n \\ \mathbf{I}_v \\ \mathbf{I}_D \\ \mathbf{I}_S \end{bmatrix} = \begin{bmatrix} \mathbf{I}_n \\ \mathbf{V}_s \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.1)$$

où

\mathbf{Y}_n = matrice d'admittances

\mathbf{V}_n = vecteurs des tensions des noeuds

\mathbf{I}_n = vecteurs des injections de courant aux noeuds

$\mathbf{V}_r, \mathbf{V}_c, \mathbf{V}_s$ = matrices des sources de tension idéal (ou barres d'équilibre)

$\mathbf{D}_r, \mathbf{D}_c$ = matrices des transformateurs

$\mathbf{S}_r, \mathbf{S}_c, \mathbf{S}_d$ = matrices des interrupteurs

\mathbf{I}_v = vecteurs de courants des sources de tension

\mathbf{I}_D = vecteurs de courants des transformateurs

\mathbf{I}_S = vecteurs de courants des interrupteurs

3.2.1 Matrice d'admittances

Chaque impédance $Z = R + iX$ connectée à un nœud k est représentée par une matrice d'admittances \mathbf{Y}_k de taille 2×2 afin de traiter les nombres réels et imaginaires séparément :

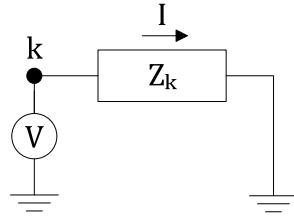


Figure 3.1 : Schéma de référence d'une impédance

$$Y_k = Z_k^{-1} = G_k + jB_k \quad (3.2)$$

$$I = \frac{V}{Z_k} = Y_k \times V \quad (3.3)$$

$$I = (G_k + jB_k)(V_R + jV_I) \quad (3.4)$$

Où V_R et V_I sont respectivement les tensions réelles et imaginaires de la source V . Le courant I est ensuite exprimé en dissociant ses parties réelles (I_R) et imaginaires (I_I):

$$I_R = G_k V_R - B_k V_I \quad (3.5)$$

$$I_I = B_k V_R + G_k V_I \quad (3.6)$$

$$\begin{bmatrix} I_R \\ I_I \end{bmatrix} = \begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \times \begin{bmatrix} V_R \\ V_I \end{bmatrix} \quad (3.7)$$

On utilisera alors la notation uniformisée suivante pour exprimer l'équation 3.7

$$I = Y_k \times V \quad (3.8)$$

où,

$$Y_k = \begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \quad (3.9)$$

Dans un cas général, toute impédance d'un réseau connectée entre les nœuds k et m contribuera aux matrices \mathbf{Y}_k , \mathbf{Y}_m et \mathbf{Y}_{km} de la matrice d'admittances du réseau :

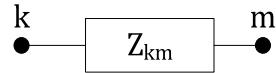


Figure 3.2 : Schéma de référence d'une impédance connectée aux nœuds k et m

$$\begin{bmatrix} \mathbf{Y}_k & -\mathbf{Y}_{km} \\ -\mathbf{Y}_{km} & \mathbf{Y}_m \end{bmatrix} \times \begin{bmatrix} \mathbf{V}_k \\ \mathbf{V}_m \end{bmatrix} = \begin{bmatrix} \mathbf{I}_k \\ \mathbf{I}_m \end{bmatrix} \quad (3.10)$$

$$[\mathbf{Y}_n] \times [\mathbf{V}_n] = [\mathbf{I}_n] \quad (3.11)$$

Où toutes les matrices \mathbf{Y}_k , \mathbf{Y}_m , et \mathbf{Y}_{km} sont des matrices 2×2 et les matrices \mathbf{V}_k , \mathbf{V}_m , \mathbf{I}_k et \mathbf{I}_m sont des vecteurs 2×1 contenant chacun une partie réel (Re) et une partie imaginaire (Im) tel que :

$$\mathbf{Y}_k = \begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \quad \mathbf{Y}_m = \begin{bmatrix} G_m & -B_m \\ B_m & G_m \end{bmatrix} \quad \mathbf{Y}_{km} = \begin{bmatrix} G_{km} & -B_{km} \\ B_{km} & G_{km} \end{bmatrix} \quad (3.12)$$

$$\mathbf{V}_k = \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix}, \mathbf{V}_m = \begin{bmatrix} V_{mR} \\ V_{mI} \end{bmatrix}, \mathbf{I}_k = \begin{bmatrix} I_{kR} \\ I_{kI} \end{bmatrix}, \mathbf{I}_m = \begin{bmatrix} I_{mR} \\ I_{mI} \end{bmatrix} \quad (3.13)$$

3.2.2 Matrices des sources de tension

Chaque source idéale de tension $V_s = V_{sR} + jV_{sI}$ connectée à un réseau entre les nœuds k et m contribue aux matrices \mathbf{Vr} et \mathbf{Vc} . La source idéale sans impédance interne est la barre d'équilibre du réseau. Les valeurs de tensions réelles V_{sR} et imaginaires V_{sI} de la source sont contenues dans la matrice \mathbf{Vs} et l'injection de courant dans le réseau est définie par le vecteur \mathbf{Iv} tel que:

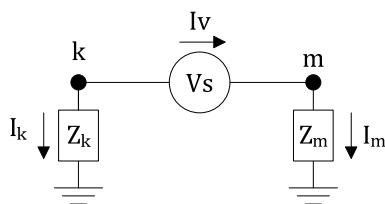


Figure 3.3 : Schéma de référence d'une source de tension

$$V_k - V_m = Vs \quad (3.14)$$

$$V_{kR} - V_{mR} = Vs_R \quad (3.15)$$

$$V_{kI} - V_{mI} = Vs_I \quad (3.16)$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} V_{mR} \\ V_{mI} \end{bmatrix} = \begin{bmatrix} Vs_R \\ Vs_I \end{bmatrix} \quad (3.17)$$

$$uV_k - uV_m = Vs \quad (3.18)$$

Pour simplifier la compréhension et s'en tenir au circuit de la Figure 3.3, la contribution de courants externes sur les nœuds k et m est considérée nulle. Cette hypothèse est également retenue pour tous les développements mathématiques qui suivront. On écrira alors les relations de courant sous la forme suivante :

$$I_k + Iv = 0 \quad (3.19)$$

$$Y_k V_k + Iv = 0 \quad (3.20)$$

$$(G_k + jB_k)(V_{kR} + jV_{kI}) + Iv_R + jIv_I = 0 \quad (3.21)$$

$$G_k V_{kR} - B_k V_{kI} + Iv_R = 0 \quad (3.22)$$

$$B_k V_{kR} + G_k V_{kI} + Iv_I = 0 \quad (3.23)$$

$$\begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} Iv_R \\ Iv_I \end{bmatrix} = \mathbf{0} \quad (3.24)$$

$$Y_k V_k + uIv_I = \mathbf{0} \quad (3.25)$$

$$I_m - Iv = 0 \quad (3.26)$$

$$Y_m V_m - Iv = 0 \quad (3.27)$$

$$(G_m + jB_m)(V_{mR} + jV_{mI}) - Iv_R - jIv_I = 0 \quad (3.28)$$

$$G_m V_{mR} - B_m V_{mI} - Iv_R = 0 \quad (3.29)$$

$$B_m V_{mR} + G_m V_{mI} - Iv_I = 0 \quad (3.30)$$

$$\begin{bmatrix} G_m & -B_m \\ B_m & G_m \end{bmatrix} \times \begin{bmatrix} V_{mR} \\ V_{mI} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} Iv_R \\ Iv_I \end{bmatrix} = \mathbf{0} \quad (3.31)$$

$$Y_m V_m - uIv = \mathbf{0} \quad (3.32)$$

À partir des équations 3.18, 3.25 et 3.32 :

$$\begin{bmatrix} Y_k & \mathbf{0} & \mathbf{u} \\ \mathbf{0} & Y_m & -\mathbf{u} \\ \mathbf{u} & -\mathbf{u} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_V \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ V_s \end{bmatrix} \quad (3.33)$$

L'équation 3.33 sous une notation uniformisée :

$$\begin{bmatrix} Y_k & \mathbf{0} & Vc_k \\ \mathbf{0} & Y_m & Vc_m \\ Vr_k & Vr_m & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_V \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ V_s \end{bmatrix} \quad (3.34)$$

$$\begin{bmatrix} Y_n & Vc \\ Vr & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_n \\ I_V \end{bmatrix} = \begin{bmatrix} I_n \\ V_s \end{bmatrix} \quad \text{où } I_n = 0 \text{ dans le cas présent} \quad (3.35)$$

3.2.3 Matrices des sources de courant

Une source de courant connectée aux noeuds k et m contribue simplement aux vecteurs de courant \mathbf{I}_i du réseau. L'injection de courant aux noeuds k et m est définie par la valeur de courant attribué à la source idéale, soit $I_i = I_R + jI_I$

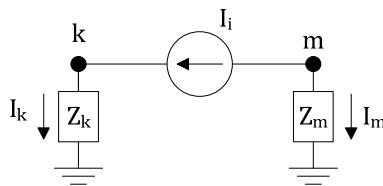


Figure 3.4 : Schéma de référence d'une source de courant

$$I_k = I_i \quad (3.36)$$

$$I_m = -I_i \quad (3.37)$$

$$Y_k V_k = +I_i \quad (3.38)$$

$$Y_m V_m = -I_i \quad (3.39)$$

$$\begin{bmatrix} Y_k & \mathbf{0} \\ \mathbf{0} & Y_m \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \end{bmatrix} = \begin{bmatrix} +I_i \\ -I_i \end{bmatrix} \quad (3.40)$$

$$[Yn] \times [Vn] = [In] \quad \text{Où } I_n = I_i \quad (3.41)$$

3.2.4 Matrices des transformateurs à deux enroulements

Dans un réseau, chaque transformateur idéal de rapport $g = re^{j\theta} = g_R + jg_I$ contribue aux matrices D_r et D_c . Le rapport g complexe permet de créer un déphasage dans le cas de la représentation monophasée d'un réseau équilibré (sources équilibrées, charges équilibrées et lignes équilibrées sans couplage). Le cas des transformateurs est particulier, car les transformateurs sont connectés à 4 nœuds k , m , i et j . L'injection de courant dans le réseau est définie par le vecteur I_D :

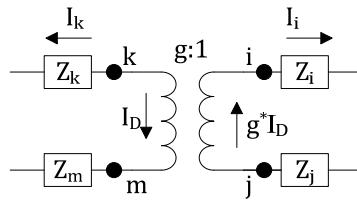


Figure 3.5 : Schéma de référence d'un transformateur idéal à deux enroulements

La relation entre les tensions des nœuds s'écrit :

$$V_k - V_m - gV_i + gV_j = 0 \quad (3.42)$$

$$(V_{kR} + jV_{kI}) - (V_{mR} + jV_{mI}) - (g_R + jg_I)(V_{iR} + jV_{iI}) + (g_R + jg_I)(V_{jR} + jV_{jI}) = 0 \quad (3.43)$$

$$V_{kR} - V_{mR} - g_R V_{iR} + g_I V_{iI} + g_R V_{jR} - g_I V_{jI} = 0 \quad (3.44)$$

$$V_{kI} - V_{mI} - g_I V_{iR} - g_R V_{iI} + g_I V_{jR} + g_R V_{jI} = 0 \quad (3.45)$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} V_{mR} \\ V_{mI} \end{bmatrix} - \begin{bmatrix} g_R & -g_I \\ g_I & g_R \end{bmatrix} \times \begin{bmatrix} V_{iR} \\ V_{iI} \end{bmatrix} + \begin{bmatrix} g_R & -g_I \\ g_I & g_R \end{bmatrix} \times \begin{bmatrix} V_{jR} \\ V_{jI} \end{bmatrix} = 0 \quad (3.46)$$

$$\mathbf{uV}_k - \mathbf{uV}_m - \mathbf{gV}_i + \mathbf{gV}_j = \mathbf{0} \quad (3.47)$$

$$I_k + I_D = 0 \quad (3.48)$$

$$Y_k V_k + I_D = 0 \quad (3.49)$$

$$\begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} I_{DR} \\ I_{DI} \end{bmatrix} = \mathbf{0} \quad (3.50)$$

$$Y_k V_k + u I_D = \mathbf{0} \quad (3.51)$$

$$I_m - I_D = 0 \quad (3.52)$$

$$Y_m V_m - I_D = 0 \quad (3.53)$$

$$\begin{bmatrix} G_m & -B_m \\ B_m & G_m \end{bmatrix} \times \begin{bmatrix} V_{mR} \\ V_{mI} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} I_{DR} \\ I_{DI} \end{bmatrix} = \mathbf{0} \quad (3.54)$$

$$Y_m V_m - u I_D = \mathbf{0} \quad (3.55)$$

$$I_i - g^* I_D = 0 \quad (3.56)$$

$$Y_i V_i - g^* I_D = 0 \quad (3.57)$$

$$(G_i + jB_i)(V_{iR} + jV_{iI}) - (g_R - jg_I)(I_{DR} + jI_{DI}) = 0 \quad (3.58)$$

$$G_i V_{iR} - B_i V_{iI} - g_R I_{DR} - g_I I_{DI} = I_{iR} \quad (3.59)$$

$$B_i V_{iR} + G_i V_{iI} + g_I I_{DR} - g_R I_{DI} = I_{iI} \quad (3.60)$$

$$\begin{bmatrix} G_i & -B_i \\ B_i & G_i \end{bmatrix} \times \begin{bmatrix} V_{iR} \\ V_{iI} \end{bmatrix} - \begin{bmatrix} g_R & g_I \\ -g_I & g_R \end{bmatrix} \times \begin{bmatrix} I_{DR} \\ I_{DI} \end{bmatrix} = \begin{bmatrix} I_{iR} \\ I_{iI} \end{bmatrix} = \mathbf{0} \quad (3.61)$$

$$Y_i V_i - g^* I_D = \mathbf{0} \quad (3.62)$$

$$I_j + g^* I_D = 0 \quad (3.63)$$

$$Y_j V_j + g^* I_D = 0 \quad (3.64)$$

$$(G_j + jB_j)(V_{jR} + jV_{jI}) + (g_R - jg_I)(I_{DR} + jI_{DI}) = 0 \quad (3.65)$$

$$G_j V_{jR} - B_j V_{jI} + g_R I_{DR} + g_I I_{DI} = 0 \quad (3.66)$$

$$B_j V_{jR} + G_j V_{jI} - g_I I_{DR} + g_R I_{DI} = 0 \quad (3.67)$$

$$\begin{bmatrix} G_j & -B_j \\ B_j & G_j \end{bmatrix} \times \begin{bmatrix} V_{jR} \\ V_{jI} \end{bmatrix} + \begin{bmatrix} g_R & g_I \\ -g_I & g_R \end{bmatrix} \times \begin{bmatrix} I_{DR} \\ I_{DI} \end{bmatrix} = \mathbf{0} \quad (3.68)$$

$$Y_j V_j + g^* I_D = \mathbf{0} \quad (3.69)$$

À partir des équations 3.47, 3.51, 3.55 ,3.62 et 3.69, on écrit l'équation :

$$\begin{bmatrix} Y_k & \mathbf{0} & \mathbf{0} & \mathbf{0} & u \\ \mathbf{0} & Y_m & \mathbf{0} & \mathbf{0} & -u \\ \mathbf{0} & \mathbf{0} & Y_i & \mathbf{0} & -g^* \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & Y_j & g^* \\ u & -u & -g & g & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ V_i \\ V_j \\ I_D \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.70)$$

$$\begin{bmatrix} Y_k & 0 & 0 & 0 & Dc_k \\ 0 & Y_m & 0 & 0 & Dc_m \\ 0 & 0 & Y_i & 0 & Dc_i \\ 0 & 0 & 0 & Y_j & Dc_j \\ Dr_k & Dr_m & Dr_i & Dr_j & 0 \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ V_i \\ V_j \\ I_D \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.71)$$

$$\begin{bmatrix} Y_n & Dc \\ Dr & 0 \end{bmatrix} \times \begin{bmatrix} V_n \\ I_D \end{bmatrix} = \begin{bmatrix} I_n \\ 0 \end{bmatrix} \quad \text{où } I_n = 0 \text{ dans le cas présent} \quad (3.72)$$

3.2.5 Matrices des interrupteurs

Chaque interrupteur idéal est modélisé par ses matrices S_r , S_c et S_d . Le contenu de ces trois (3) matrices varie en fonction de l'état de l'interrupteur (ouvert ou fermé). L'injection de courant des interrupteurs dans le réseau est définie dans le vecteur I_s :

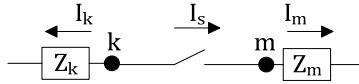


Figure 3.6 : Schéma de référence d'un interrupteur (switch)

Si l'interrupteur est fermé,

$$V_k - V_m = 0 \quad (3.73)$$

$$V_{kR} - V_{mR} = 0 \quad (3.74)$$

$$V_{kl} - V_{ml} = 0 \quad (3.75)$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kl} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} V_{mR} \\ V_{ml} \end{bmatrix} = \mathbf{0} \quad (3.76)$$

$$uV_k - uV_m = \mathbf{0} \quad (3.77)$$

$$I_k + I_S = 0 \quad (3.78)$$

$$Y_k V_k + I_S = 0 \quad (3.79)$$

$$\begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kl} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} I_{SR} \\ I_{SI} \end{bmatrix} = \mathbf{0} \quad (3.80)$$

$$Y_k V_k + uI_S = \mathbf{0} \quad (3.81)$$

$$I_m - I_S = 0 \quad (3.82)$$

$$Y_m V_k - I_S = 0 \quad (3.83)$$

$$\begin{bmatrix} G_m & -B_m \\ B_m & G_m \end{bmatrix} \times \begin{bmatrix} V_{mR} \\ V_{mI} \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} I_{SR} \\ I_{SI} \end{bmatrix} = \mathbf{0} \quad (3.84)$$

$$Y_m V_m - u I_S = \mathbf{0} \quad (3.85)$$

Des équations 3.77, 3.81 et 3.85 :

$$\begin{bmatrix} Y_k & \mathbf{0} & u \\ \mathbf{0} & Y_m & -u \\ u & -u & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_S \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.86)$$

$$\begin{bmatrix} Y_k & \mathbf{0} & S c_k \\ \mathbf{0} & Y_m & S c_m \\ S r_k & S r_m & S d \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_S \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.87)$$

$$\begin{bmatrix} Y_n & S c \\ S r & S d \end{bmatrix} \times \begin{bmatrix} V_n \\ I_S \end{bmatrix} = \begin{bmatrix} I_n \\ \mathbf{0} \end{bmatrix} \quad (3.88)$$

Si l'interrupteur est ouvert,

$$I_S = \begin{bmatrix} I_{SR} \\ I_{SI} \end{bmatrix} = \mathbf{0} \quad (3.89)$$

$$I_k = I_m = \mathbf{0} \quad (3.90)$$

$$\begin{bmatrix} Y_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & Y_m & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & u \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_S \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.91)$$

$$\begin{bmatrix} Y_k & \mathbf{0} & S c_k \\ \mathbf{0} & Y_m & S c_m \\ S r_k & S r_m & S d \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_S \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad (3.92)$$

$$\begin{bmatrix} Y_n & S c \\ S r & S d \end{bmatrix} \times \begin{bmatrix} V_n \\ I_S \end{bmatrix} = \begin{bmatrix} I_n \\ \mathbf{0} \end{bmatrix} \quad \text{où } I_n = 0 \text{ dans le cas présent} \quad (3.93)$$

3.3 Ajout des contraintes de puissance à la matrice nodale

En s'appuyant sur la formulation de [17-19], les contraintes de puissance sont ajoutées à l'expression de la matrice nodale du système à travers les matrices $\mathbf{C}_{LPQ}, \mathbf{A}_{IL}, \mathbf{I}_L, \mathbf{L}_{PQ}$ pour les charges PQ et $\mathbf{C}_{GPV}, \mathbf{A}_{IG}, \mathbf{I}_G, \mathbf{G}_{PV}$ pour les sources PV :

$$\begin{bmatrix} \mathbf{Y_n} & \mathbf{Vc} & \mathbf{Dc} & \mathbf{Sc} & \mathbf{A}_{IL} & \mathbf{A}_{IG} \\ \mathbf{Vr} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{Dr} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{Sr} & \mathbf{0} & \mathbf{0} & \mathbf{Sd} & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_{LPQ} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_{GPV} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} \mathbf{V_n} \\ \mathbf{I_V} \\ \mathbf{I_D} \\ \mathbf{I_S} \\ \mathbf{I_L} \\ \mathbf{I_G} \end{bmatrix} = \begin{bmatrix} \mathbf{0} + \Sigma \mathbf{I_i} \\ \mathbf{V_s} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{L}_{PQ} \\ \mathbf{G}_{PV} \end{bmatrix} \quad (3.94)$$

Où

$\mathbf{Y_n}$ = matrice d'admittances

$\mathbf{V_n}$ = tension des noeuds

$\mathbf{I_n}$ = injection de courant aux noeuds

$\mathbf{Vr}, \mathbf{Vc}, \mathbf{V_s}$ = matrices modélisant les sources de tension idéal (ou barres d'équilibre)

\mathbf{Dr}, \mathbf{Dc} = matrices modélisant les transformateurs

$\mathbf{Sr}, \mathbf{Sc}, \mathbf{Sd}$ = matrices modélisant les switchs

$\mathbf{I_V}$ = vecteurs de courants des sources de tension

$\mathbf{I_D}$ = vecteurs de courants des transformateurs

$\mathbf{I_S}$ = vecteurs de courants des switchs

$\mathbf{C}_{LPQ}, \mathbf{A}_{IL}, \mathbf{I}_L, \mathbf{L}_{PQ}$ = matrices des charges PQ

$\mathbf{C}_{GPV}, \mathbf{A}_{IG}, \mathbf{I}_G, \mathbf{G}_{PV}$ = matrices des charges PV

3.3.1 Contraintes des charges PQ

Les contraintes d'une charge monophasée PQ connectée au nœud k d'un réseau sont définies à travers les matrices \mathbf{C}_{LPQ} et \mathbf{A}_{IL} . L'injection de courant des charges PQ est définie dans le vecteur \mathbf{I}_L . La modélisation d'une charge PQ triphasée peut se faire alors par trois charges monophasées.

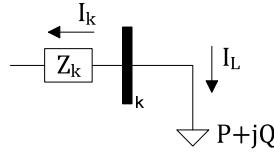


Figure 3.7 : Schéma de référence d'une charge PQ

$$V_k \times I_L^* = P + jQ \quad (3.95)$$

$$I_{LR}V_{kR} + I_{LI}V_{kI} = P \quad (3.96)$$

$$-I_{LI}V_{kR} + I_{LR}V_{kI} = Q \quad (3.97)$$

$$\begin{bmatrix} I_{LR} & I_{LI} \\ -I_{LI} & I_{LR} \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} = \begin{bmatrix} P \\ Q \end{bmatrix} = L_{PQ} \quad (3.98)$$

$$C_{LPQ_k}V_k = L_{PQ} \quad (3.99)$$

Pour simplifier la compréhension, la contribution de courants externes sur le nœud k est considérée nulle. Cette hypothèse est également retenue pour tous les développements mathématiques qui suivront. On écrira la relation de courant au nœud k :

$$I_k + I_L = 0 \quad (3.100)$$

$$Y_k V_k + I_L = 0 \quad (3.101)$$

$$\begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} I_{LR} \\ I_{LI} \end{bmatrix} = 0 \quad (3.102)$$

$$Y_k V_k + u I_L = 0 \quad (3.103)$$

À partir de 3.99 et 3.103 :

$$\begin{bmatrix} Y_k & u \\ C_{LPQ_k} & 0 \end{bmatrix} \times \begin{bmatrix} V_k \\ I_L \end{bmatrix} = \begin{bmatrix} 0 \\ L_{PQ} \end{bmatrix} \quad (3.104)$$

$$\begin{bmatrix} Y_k & A_{IL_k} \\ C_{LPQ_k} & 0 \end{bmatrix} \times \begin{bmatrix} V_k \\ I_L \end{bmatrix} = \begin{bmatrix} 0 \\ L_{PQ} \end{bmatrix} \quad (3.105)$$

$$\begin{bmatrix} Y_n & A_{IL} \\ C_{LPQ} & 0 \end{bmatrix} \times \begin{bmatrix} V_k \\ I_L \end{bmatrix} = \begin{bmatrix} I_n \\ L_{PQ} \end{bmatrix} \quad \text{où } I_n = 0 \text{ dans le cas présent} \quad (3.106)$$

3.3.2 Contraintes des sources PV

Les contraintes d'une source PV monophasée connectée au nœud k d'un réseau sont définies à travers les matrices \mathbf{C}_{GPV} et \mathbf{A}_{IG} . Les 3 phases seront alors combinées pour une source PV triphasée. L'injection de courant des sources PV est définie dans le vecteur \mathbf{I}_G . En supposant que \mathbf{I}_G soit exprimé en fonction de P et V connus, on a alors :

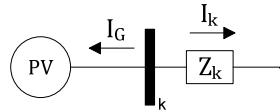


Figure 3.8 : Schéma de référence d'une source PV

$$P = -I_{GR}V_{kR} - I_{GI}V_{kI} \quad (3.107)$$

$$V = V_k = \sqrt{V_{kR}^2 + V_{kI}^2} = \frac{V_{kR}^2 + V_{kI}^2}{|V_k|} \quad (3.108)$$

$$\begin{bmatrix} -I_{GR} & -I_{GI} \\ \frac{V_{kR}}{|V_k|} & \frac{V_{kI}}{|V_k|} \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} = \begin{bmatrix} P \\ V \end{bmatrix} = \mathbf{G}_{PV} \quad (3.109)$$

$$\mathbf{C}_{GPV_k} \mathbf{V}_k = \mathbf{G}_{PV} \quad (3.110)$$

$$I_k + I_G = 0 \quad (3.111)$$

$$Y_k V_k + I_G = 0 \quad (3.112)$$

$$\begin{bmatrix} G_k & -B_k \\ B_k & G_k \end{bmatrix} \times \begin{bmatrix} V_{kR} \\ V_{kI} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} I_{GR} \\ I_{GI} \end{bmatrix} = \mathbf{0} \quad (3.113)$$

$$Y_k V_k + u I_G = \mathbf{0} \quad (3.114)$$

$$\begin{bmatrix} Y_k & u \\ \mathbf{C}_{GPV_k} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ I_G \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_{PV} \end{bmatrix} \quad (3.115)$$

$$\begin{bmatrix} Y_k & A_{IG_k} \\ \mathbf{C}_{GPV_k} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ I_G \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{G}_{PV} \end{bmatrix} \quad (3.116)$$

$$\begin{bmatrix} Y_n & A_{IG} \\ \mathbf{C}_{GPV} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ I_G \end{bmatrix} = \begin{bmatrix} I_n \\ \mathbf{G}_{PV} \end{bmatrix} \quad \text{où } I_n = 0 \text{ dans le cas présent} \quad (3.117)$$

3.4 Écoulement de puissance

3.4.1 Méthode de Newton-Raphson

Pour introduire les contraintes d'écoulement de puissance, le logiciel APREM emprunte l'approche du logiciel EMTP-RV [17-19] et modélise les réseaux sous forme de matrice nodale-augmentée-modifiée, sans avoir à convertir les données en per-unit. On obtient alors le système non linéaire suivant :

$$y = f(x) \quad \text{où } y \text{ et } f \text{ sont connus} \quad (3.118)$$

Les variables inconnues x sont alors évaluées en résolvant le système par itérations suivant la méthode de Newton-Raphson.

L'algorithme de Newton-Raphson est utilisé dans l'outil APREM pour résoudre les analyses de répartition de puissance, car, pour les topologies des grands réseaux complexes, il est plus robuste que les algorithmes de Gauss-Seidel ou Backward/Foward sweep. Chaque itération de Newton-Raphson réalise les opérations suivantes :

$$\Delta y^{(k)} = y - f(x^{(k)}) \quad (3.119)$$

$$\Delta x^{(k)} = -(J^{(k)})^{-1} \Delta y^{(k)} \quad (3.120)$$

$$x^{(k+1)} = x^{(k)} + \Delta x^{(k)} \quad (3.121)$$

où

- k est le numéro de l'itération en cours;
- $x^{(k)}$, le vecteur des inconnus évalués à l'itération k ;
- f , la fonction du système non-linéaire;
- $J^{(k)}$, la Jacobienne du système à l'itération k ;
- $\Delta y^{(k)}$, le vecteur résidu à l'itération k ;
- $\Delta x^{(k)}$, la variation du vecteur x à l'itération k .

Le système non linéaire $f(x) = y$ dans le cas d'un réseau est décrit comme suit :

$$\begin{bmatrix} Y_n & V_c & D_c & S_c & A_{IL} & A_{IG} \\ V_r & 0 & 0 & 0 & 0 & 0 \\ D_r & 0 & 0 & 0 & 0 & 0 \\ S_r & 0 & 0 & S_d & 0 & 0 \\ C_{LPQ} & 0 & 0 & 0 & 0 & 0 \\ C_{GPV} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} V_n \\ I_V \\ I_D \\ I_S \\ I_L \\ I_G \end{bmatrix} = \begin{bmatrix} 0 + \Sigma I_i \\ V_s \\ 0 \\ 0 \\ L_{PQ} \\ G_{PV} \end{bmatrix} \quad (3.122)$$

$$f_n = Y_n V_n + V_c I_V + D_c I_D + S_c I_S + A_{IL} I_D + A_{IG} I_G = 0 + \Sigma I_i \quad (3.123)$$

$$f_V = V_r V_n = V_s \quad (3.124)$$

$$f_D = D_r V_n = 0 \quad (3.125)$$

$$f_S = S_r V_n + S_d I_S = 0 \quad (3.126)$$

$$f_{LPQ} = C_{LPQ} V_n = L_{PQ} \quad (3.127)$$

$$f_{GPV} = C_{GPV} V_n = G_{PV} \quad (3.128)$$

3.4.2 La matrice Jacobienne

La Jacobienne est obtenue en prenant les dérivées partielles du système non linéaire par rapport aux tensions et courants du système non linéaire, soit le vecteur x et ses variables V_n, I_V, I_D, I_S, I_D et I_G :

$$J = \frac{\partial f(x)}{\partial x} \Bigg|_{x=x(k)} = \begin{bmatrix} \frac{\partial f_n}{\partial V_n} & \frac{\partial f_n}{\partial I_V} & \frac{\partial f_n}{\partial I_D} & \frac{\partial f_n}{\partial I_S} & \frac{\partial f_n}{\partial I_L} & \frac{\partial f_n}{\partial I_G} \\ \frac{\partial f_V}{\partial V_n} & \frac{\partial f_V}{\partial I_V} & \frac{\partial f_V}{\partial I_D} & \frac{\partial f_V}{\partial I_S} & \frac{\partial f_V}{\partial I_L} & \frac{\partial f_V}{\partial I_G} \\ \frac{\partial f_D}{\partial V_n} & \frac{\partial f_D}{\partial I_V} & \frac{\partial f_D}{\partial I_D} & \frac{\partial f_D}{\partial I_S} & \frac{\partial f_D}{\partial I_L} & \frac{\partial f_D}{\partial I_G} \\ \frac{\partial f_S}{\partial V_n} & \frac{\partial f_S}{\partial I_V} & \frac{\partial f_S}{\partial I_D} & \frac{\partial f_S}{\partial I_S} & \frac{\partial f_S}{\partial I_L} & \frac{\partial f_S}{\partial I_G} \\ \frac{\partial f_{LPQ}}{\partial V_n} & \frac{\partial f_{LPQ}}{\partial I_V} & \frac{\partial f_{LPQ}}{\partial I_D} & \frac{\partial f_{LPQ}}{\partial I_S} & \frac{\partial f_{LPQ}}{\partial I_L} & \frac{\partial f_{LPQ}}{\partial I_G} \\ \frac{\partial f_{GPV}}{\partial V_n} & \frac{\partial f_{GPV}}{\partial I_V} & \frac{\partial f_{GPV}}{\partial I_D} & \frac{\partial f_{GPV}}{\partial I_S} & \frac{\partial f_{GPV}}{\partial I_L} & \frac{\partial f_{GPV}}{\partial I_G} \end{bmatrix} \quad (3.129)$$

$$J(\mathbf{k}) = \frac{\partial f(x)}{\partial x} \Big|_{x=x(k)} = \begin{bmatrix} \mathbf{Yn} & \mathbf{Vc} & \mathbf{Dc} & \mathbf{Sc} & \mathbf{A}_{IL} & \mathbf{A}_{IG} \\ \mathbf{Vr} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{Dr} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{Sr} & \mathbf{0} & \mathbf{0} & \mathbf{Sd} & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_{LPQ} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\partial f_{LPQ}}{\partial I_L} & \mathbf{0} \\ \mathbf{C}_{GPV} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\partial f_{GPV}}{\partial I_G} \end{bmatrix} \Big|_{x=x(k)} \quad (3.130)$$

On remarque que les fonctions f_{LPQ} et f_{GPV} ont également une dérivée partielle par rapport aux courants \mathbf{I}_L et \mathbf{I}_G respectivement (en plus de leur dérivée par rapport à \mathbf{V}_n) puisque les matrices des charges PQ et sources PV, $\mathbf{C}_{LPQ} = \begin{bmatrix} I_{LR} & I_{LI} \\ -I_{LI} & I_{LR} \end{bmatrix}$ et $\mathbf{C}_{GPV} = \begin{bmatrix} -I_{GR} & -I_{GI} \\ \frac{V_{nR}}{|V_n|} & \frac{V_{nI}}{|V_n|} \end{bmatrix}$, ne sont pas constantes et sont fonction des courants \mathbf{I}_L et \mathbf{I}_G .

Dérivée partielle de la charge PQ

Pour résoudre la dérivée partielle de la charge PQ, la fonction f_{LPQ} est décomposée afin de séparer les dérivées partielles des puissances réelle et réactive :

$$f_{LPQ} = V_n \times I_L^* \quad (3.131)$$

$$f_{LPQ_R} = I_{LR}V_{nR} + I_{LI}V_{nI} = P \quad (3.132)$$

$$f_{LPQ_I} = -I_{LI}V_{nR} + I_{LR}V_{nI} = Q \quad (3.133)$$

$$\partial f_{LPQ_R} = I_{LR}\partial V_{nR} + I_{LI}\partial V_{nI} + V_{nR}\partial I_{LR} + V_{nI}\partial I_{LI} \quad (3.134)$$

$$\partial f_{LPQ_I} = -I_{LI}\partial V_{nR} + I_{LR}\partial V_{nI} + V_{nI}\partial I_{LR} - V_{nR}\partial I_{LI} \quad (3.135)$$

$$\partial f_{LPQ} = \begin{bmatrix} I_{LR} & I_{LI} \\ -I_{LI} & I_{LR} \end{bmatrix} \begin{bmatrix} \partial V_{nR} \\ \partial V_{nI} \end{bmatrix} + \begin{bmatrix} V_{nR} & V_{nI} \\ V_{nI} & -V_{nR} \end{bmatrix} \begin{bmatrix} \partial I_{LR} \\ \partial I_{LI} \end{bmatrix} \quad (3.136)$$

$$\partial f_{LPQ} = \mathbf{C}_{LPQ} \partial \mathbf{V}_n + \mathbf{D}_{LPQ} \partial \mathbf{I}_L \quad (3.137)$$

$$\frac{\partial f_{LPQ}}{\partial V_n} = \mathbf{C}_{LPQ} \quad (3.138)$$

$$\frac{\partial f_{LPQ}}{\partial I_L} = \mathbf{D}_{LPQ} \quad (3.139)$$

Dérivée partielle de la source PV

Pour résoudre la dérivée partielle de la source PV, la fonction f_{GPV} est décomposée afin de séparer les dérivées partielles de la puissance réelle P et la tension absolue V :

$$f_{GPV_P} = -I_{GR}V_{nR} - I_{GI}V_{nI} = P \quad (3.140)$$

$$f_{GPV_V} = \sqrt{V_{nR}^2 + V_{nI}^2} = \frac{V_{nR}^2 + V_{nI}^2}{|V_n|} = V_n \quad (3.141)$$

On trouve alors :

$$\partial f_{GPV_P} = -I_{GR}\partial V_{nR} - I_{GI}\partial V_{nI} + V_{kR}\partial I_{GR} + V_{kI}\partial I_{GI} \quad (3.142)$$

$$\partial f_{GPV_V} = \sqrt{V_{nR}\partial V_{nR} + V_{nI}\partial V_{nI}} = \frac{V_{nR}\partial V_{nR} + V_{nI}\partial V_{nI}}{|V_k|} \quad (3.143)$$

$$\partial f_{GPV} = \begin{bmatrix} -I_{GR} & -I_{GI} \\ \frac{V_{kR}}{|V_k|} & \frac{V_{kI}}{|V_k|} \end{bmatrix} \times \begin{bmatrix} \partial V_{nR} \\ \partial V_{nI} \end{bmatrix} + \begin{bmatrix} V_{kR} & V_{kI} \\ 0 & 0 \end{bmatrix} \times \begin{bmatrix} \partial I_{GR} \\ \partial I_{GI} \end{bmatrix} \quad (3.144)$$

$$\partial f_{GPV} = H_{PV} \partial V_n + D_{GPV} \partial I_L \quad (3.145)$$

$$\frac{\partial f_{GPV}}{\partial V_n} = H_{PV} \quad (3.146)$$

$$\frac{\partial f_{GPV}}{\partial I_G} = D_{GPV} \quad (3.147)$$

La Jacobienne du système non linéaire a donc la forme suivante :

$$J(x) = \frac{\partial f}{\partial x} \Bigg|_{x=x(k)} = \begin{bmatrix} Y_n & V_c & D_c & S_c & A_{IL} & A_{IG} \\ V_r & 0 & 0 & 0 & 0 & 0 \\ D_r & 0 & 0 & 0 & 0 & 0 \\ S_r & 0 & 0 & S_d & 0 & 0 \\ C_{LPQ} & 0 & 0 & 0 & D_{LPQ} & 0 \\ C_{GPV} & 0 & 0 & 0 & 0 & D_{GPV} \end{bmatrix}_{x=x(k)} \quad (3.148)$$

où seul les matrices C_{LPQ} , C_{GPV} , D_{LPQ} et D_{GPV} varient et doivent être réévaluées à chaque itération. Toutes les autres matrices de la Jacobienne sont constantes.

Le système d'équation à résoudre par la méthode de Newton-Raphson est :

$$\begin{bmatrix} Y_n & V_c & D_c & S_c & A_{IL} & A_{IG} \end{bmatrix}^{(k)} \times \begin{bmatrix} \Delta V_n \\ \Delta I_V \\ \Delta I_D \\ \Delta I_S \\ \Delta I_L \\ \Delta I_G \end{bmatrix}^{(k)} = - \begin{bmatrix} \Delta f_n \\ \Delta f_V \\ \Delta f_D \\ \Delta f_S \\ \Delta f_{LPQ} \\ \Delta f_{GPV} \end{bmatrix}^{(k)} \quad (3.149)$$

Le système converge lorsque $\Delta x^{(k)}$ et/ou $\Delta y^{(k)}$ tendent vers 0

3.4.3 Initialisation et itérations de Newton-Raphson

Initialisation

Avant de pouvoir lancer les itérations de Newton-Raphson, le système d'équation $f(x) = y$ doit être initialisé, afin de trouver le vecteur $x^{(0)}$ à partir de l'équation directe du réseau (3.94). Seulement, les sous-matrices C_{LPQ} et H_{PV} ne peuvent être évaluées sans la valeur des courants I_L et I_G , et les tensions V_n .

L'opération d'initialisation utilisée dans l'outil APREM remplace toutes les charges PQ et sources PV, par des impédances (Z_{eq}) et sources de tension (V_{eq}) équivalentes. Il est possible de déduire Z_{eq} en supposant que la tension nominale d'opération est appliquée, résultant en une puissance à la charge $P+jQ$. Pour la source V_{eq} , l'outil fixe l'amplitude imposée de la tension et calcule son angle en fonction de la puissance à débiter. On obtient alors la matrice directe du système, sans les contraintes de puissance :

$$\begin{bmatrix} Y_n & V_c & D_c & S_c \end{bmatrix} \times \begin{bmatrix} V_n \\ I_V \\ I_D \\ I_S \end{bmatrix}^{(0)} = \begin{bmatrix} 0 + \Sigma I_i \\ V_s \\ 0 \\ 0 \end{bmatrix} \quad (3.150)$$

$$\begin{bmatrix} V_n \\ I_V \\ I_D \\ I_S \end{bmatrix}^{(0)} = \begin{bmatrix} Y_n & V_c & D_c & S_c \end{bmatrix}^{-1} \times \begin{bmatrix} 0 + \Sigma I_i \\ V_s \\ 0 \\ 0 \end{bmatrix} \quad (3.151)$$

Les courants équivalents des charges PQ et sources PV sont calculé à partir des résultats obtenus avec les impédances et sources de tension équivalentes et sont ensuite rapportés aux courants \mathbf{I}_L et \mathbf{I}_G . Les sources et impédances équivalentes sont ensuite retirées de la matrice directe du réseau.

L'opération permet ainsi de trouver $\mathbf{C}_{LPQ}^{(0)}$, $\mathbf{H}_{PV}^{(0)}$, $\mathbf{D}_{LPQ}^{(0)}$ et $\mathbf{D}_{GPV}^{(0)}$ en fonction de $\mathbf{I}_L^{(0)}$, $\mathbf{I}_G^{(0)}$ et $\mathbf{V}_n^{(0)}$ et également compléter la matrice du réseau, la Jacobienne et le vecteur des inconnus $\mathbf{x}^{(0)}$ pour l'itération $k = 0$. L'outil exécute alors les 4 étapes suivantes à chaque itération de Newton-Raphson.

Étape 1

$$\Delta \mathbf{y}^{(0)} = \mathbf{y} - \mathbf{f}(\mathbf{x}^{(0)}) \quad (3.152)$$

$$\begin{bmatrix} \Delta f_1 \\ \Delta f_2 \\ \Delta f_3 \\ \Delta f_4 \\ \Delta f_5 \\ \Delta f_6 \end{bmatrix}^{(0)} = \begin{bmatrix} \mathbf{0} + \Sigma \mathbf{I}_i \\ \mathbf{V}_S \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{L}_{PQ} \\ \mathbf{G}_{PV} \end{bmatrix} - \begin{bmatrix} \mathbf{Y}_n & \mathbf{V}_c & \mathbf{D}_c & \mathbf{S}_c & \mathbf{A}_{IL} & \mathbf{A}_{IG} \\ \mathbf{V}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{D}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}_r & \mathbf{0} & \mathbf{0} & \mathbf{S}_d & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_{LPQ}^{(0)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{H}_{PV}^{(0)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} \mathbf{V}_n^{(0)} \\ \mathbf{I}_V \\ \mathbf{I}_D \\ \mathbf{I}_S \\ \mathbf{I}_L \\ \mathbf{I}_G \end{bmatrix} \quad (3.153)$$

Étape 2

$$J(\mathbf{0}) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Bigg|_{\mathbf{x}=\mathbf{x}(0)} \quad (3.154)$$

$$J(\mathbf{0}) = \begin{bmatrix} \mathbf{Y}_n & \mathbf{V}_c & \mathbf{D}_c & \mathbf{S}_c & \mathbf{A}_{IL} & \mathbf{A}_{IG} \\ \mathbf{V}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{D}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}_r & \mathbf{0} & \mathbf{0} & \mathbf{S}_d & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_{LPQ}^{(0)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}_{LPQ}^{(0)} & \mathbf{0} \\ \mathbf{H}_{PV}^{(0)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}_{GPV}^{(0)} \end{bmatrix} \quad (3.155)$$

Étape 3

$$\Delta \boldsymbol{x}^{(0)} = -(\boldsymbol{J}^{(0)})^{-1} \Delta \boldsymbol{y}^{(0)} \quad (3.156)$$

$$\begin{bmatrix} \Delta \mathbf{V}_n \\ \Delta \mathbf{I}_V \\ \Delta \mathbf{I}_D \\ \Delta \mathbf{I}_S \\ \Delta \mathbf{I}_L \\ \Delta \mathbf{I}_G \end{bmatrix}^{(0)} = - \begin{bmatrix} \mathbf{Y}_n & \mathbf{V}_c & \mathbf{D}_c & \mathbf{S}_c & \mathbf{A}_{IL} & \mathbf{A}_{IG} \\ \mathbf{V}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{D}_r & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}_r & \mathbf{0} & \mathbf{0} & \mathbf{S}_d & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_{LPQ}^{(0)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}_{LPQ}^{(0)} & \mathbf{0} \\ \mathbf{H}_{PV}^{(0)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}_{GPV}^{(0)} \end{bmatrix} \times \begin{bmatrix} \Delta f_1 \\ \Delta f_2 \\ \Delta f_3 \\ \Delta f_4 \\ \Delta f_5 \\ \Delta f_6 \end{bmatrix}^{(0)} \quad (3.157)$$

Étape 4

$$\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \Delta \boldsymbol{x}^{(0)} \quad (3.158)$$

$$\begin{bmatrix} \mathbf{V}_n \\ \mathbf{I}_V \\ \mathbf{I}_D \\ \mathbf{I}_S \\ \mathbf{I}_L \\ \mathbf{I}_G \end{bmatrix}^{(1)} = \begin{bmatrix} \mathbf{V}_n \\ \mathbf{I}_V \\ \mathbf{I}_D \\ \mathbf{I}_S \\ \mathbf{I}_L \\ \mathbf{I}_G \end{bmatrix}^{(0)} + \begin{bmatrix} \Delta \mathbf{V}_n \\ \Delta \mathbf{I}_V \\ \Delta \mathbf{I}_D \\ \Delta \mathbf{I}_S \\ \Delta \mathbf{I}_L \\ \Delta \mathbf{I}_G \end{bmatrix}^{(0)} \quad (3.159)$$

L'outil réévalue alors les matrices $\mathbf{C}_{LPQ}^{(1)}$, $\mathbf{H}_{PV}^{(1)}$, $\mathbf{D}_{LPQ}^{(1)}$ et $\mathbf{D}_{GPV}^{(1)}$ en fonction de $\mathbf{I}_L^{(1)}$, $\mathbf{I}_G^{(1)}$ et $\mathbf{V}_n^{(1)}$. L'itération suivante reprend à l'étape 1 avec $\Delta \boldsymbol{y}^{(1)} = \boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}^{(1)})$

L'avantage de cette méthode réside justement dans le fait qu'il n'est pas nécessaire de réévaluer l'intégralité de la matrice à chaque itération. Cet atout est majeur pour les études paramétriques ou de fiabilité, car les changements topologiques ou paramétriques résultant des modifications opérées automatiquement sur le réseau affecteront peu le contenu de la matrice.

3.5 Les nouvelles classes de l'outil APREM

3.5.1 Organisation des classes

L'un des principaux objectifs de l'outil APREM est de rendre disponibles de nouveaux objets à l'usager final afin qu'il puisse définir son réseau électrique pour en faire l'analyse paramétrique dans Matlab. Tous les objets créés pour l'outil APREM suivent une structure orientée-objet. Néanmoins, le présent document ne traite pas en détail des moyens disponibles dans Matlab pour y parvenir, mais présente la structure des objets dans l'outil APREM.

Pour représenter un réseau électrique sous forme mathématique, les équipements électriques sont modélisés en groupant un ou plusieurs des 5 éléments suivants :

- Les impédances,
- Les sources tensions,
- Les sources de courant,
- Les transformateurs, et
- Les interrupteurs (Switch),

auxquels sont ajoutées les contraintes de puissance pour pouvoir effectuer des analyses de répartition de puissance dans le réseau :

- les charges PQ,
- les sources PV.

Ces 7 types d'équipements constituent donc la base de tout réseau électrique et ils ont un certain nombre de paramètres en commun. Tous ont deux terminaux, soit les nœuds k et m . Ils ont aussi une tension V qui est la différence entre la tension aux nœuds k (V_k) et m (V_m), et un courant I qui les traverse. Il est ainsi possible de déterminer la puissance consommée par chaque équipement $S = (V_k - V_m) \times I^*$, mais également la puissance entrante aux nœuds k (S_k) et m (S_m) de la même façon. Et finalement, puisque que tout équipement électrique peut être en service ou non, ils ont tout un état ON/OFF.

Ainsi, tous les objets de l'outil APREM héritent de ces variables. La nouvelle classe *cEquipment* développée est donc à la base de l'outil APREM pour que tout objet créé pour représenter un équipement électrique hérite des variables k , m , V_k , V_m , V , I , S , S_k , S_m et ON de la classe *cEquipment* (voir Figure 3.9).

```
>> a = cEquipment

a =

    ON: 0
    k: 'GND'
    m: 'GND'
    Vk: 0
    Vm: 0
    V: 0
    I: []
    S: []
    Sk: []
    Sm: []
    function: ''
```

Figure 3.9 : la classe *cEquipment*

Le rôle du paramètre *function* qui fait également partie intégrante de la classe *cEquipment* sera traité plus en détail ultérieurement.

Dans l'exemple ci-dessus, les paramètres qui s'affichent sont en réalité les variables accessibles par l'usager, soit les *variables publiques*. Derrière chacune des nouvelles classes développées dans l'outil APREM se cachent également des *variables privées* qui ne sont pas accessibles par l'usager.

Les variables privées de la classe *cEquipment* sont :

- mNode_k* stocke le numéro du nœud k de l'équipement
- mNode_m* stocke le numéro du nœud m de l'équipement
- mON* stocke l'état de l'équipement

- mMatrix* stocke le(s) matrice(s) de l'équipement
mFunction stocke le(s) fonction(s) spéciale(s) développée(s) par l'usager pour l'équipement

Il est important de noter que la classe *cEquipment* ne mémorise pas à l'intérieur de ses objets les valeurs numériques de tension, de courant, ni de puissance des équipements qu'elle modélise, ni même le nom du/des nœuds auxquelles les équipements sont raccordés. Ces données sont externes aux objets de la classe *cEquipment*. Pour les afficher ou les modifier, l'objet appelle via ces méthodes des fonctions de l'outil APREM qui retournent ces valeurs. Ainsi, l'appel des variables publiques suivantes permet de :

- k* afficher/modifier le nom du nœud *k* via la fonction *fNode(mNode_k)*
- m* afficher/modifier le nom du nœud *k* via la fonction *fNode(mNode_m)*
- Vk* afficher la tension calculée au nœud *k* via *fNode(mNode_k, 'voltage')*
- Vm* afficher la tension calculée au nœud *m* via *fNode(mNode_k, 'voltage')*
- V* afficher le résultat de l'opération $V_k - V_m$.

La fonction *fNode* de l'outil APREM gère la numérotation des nœuds des réseaux et retourne les tensions calculées [Vn] après chaque analyse de répartition de puissance (voir Figure 3.10).

```
>> fNode
ans =
GND: 0
B1: 100
B2: 98.5773 - 1.5345i
B3: 49.2887 - 0.7672i
B4: 49.2214 - 2.2439i
B5: 49.2214 - 2.2439i
bus1: 120
bus2: 1.1612e+002 -3.1667e+000i
```

Figure 3.10 : la fonction *fNode*

Puisque que le calcul du courant d'une impédance et d'une source de courant diffère des autres équipements, chaque objet enfant de la classe *cEquipment* redéfinit l'opération exécutée lors de l'appelle de la variable *I*:

- pour une impédance, *I* retourne le résultat de l'opération $(V_k - V_m) / Z$
- pour une source de courant, *I* retourne la valeur assignée par l'usager
- pour les autres équipements, *I* retourne la valeur d'injection de courant de l'équipement dans le réseau en appelant *fCurrent*.

La fonction *fCurrent* de l'outil APREM retourne les injections de courant [\mathbf{I}_V , \mathbf{I}_D , \mathbf{I}_S , \mathbf{I}_L , \mathbf{I}_G] dans le réseau calculées lors de chaque analyse de répartition de puissance (voir Figure 3.11).

```
>> I = fCurrent
I =
V: [2x1 double]
D: 14.7856 + 0.5591i
S: 24.6107 - 1.1219i
LPQ: 5.1044 - 2.2921i
GPV: [0x1 double]

>> I.V
ans =
-14.7856 - 0.5591i
-5.1044 + 2.2921i
```

Figure 3.11 : la fonction *fCurrent*

Dans l'exemple de la Figure 3.11, *I.V* retourne une matrice de 2×1 puisque le réseau est composé de deux (2) sources de tension. Aucune source PV n'est connectée au réseau, c'est pourquoi *I.GPV* retourne une matrice vide.

À partir de la classe *cEquipment*, sept (7) classes enfants sont développées pour représenter les 7 composants de base requis pour construire et analyser les réseaux dans l'outil APREM, soit :

- *cImpedance*,
- *cIsource*,
- *cVsource*,
- *cTxfo*,
- *cSwitch*,
- *cLoad*,
- *cPVsource*.

Ces classes permettent de créer des objets Matlab modélisant respectivement les impédances, les sources de courant, les sources de tension, les transformateurs, les interrupteurs, les charges PQ et les sources PV d'un réseau électrique. La Figure 3.12 présente la structure des classes de base de l'outil APREM.

Les pages qui suivent présentent l'organisation des différentes classes développées pour l'outil APREM. Pour chaque classe, on présente :

- les variables publics accessibles par l'utilisateur
- les fonctions *built-in* redéfinis dans la classe
- le constructeur
- l'organisation des sous-matrices à l'intérieur de l'objet
- les fonctions APREM propres à l'objet

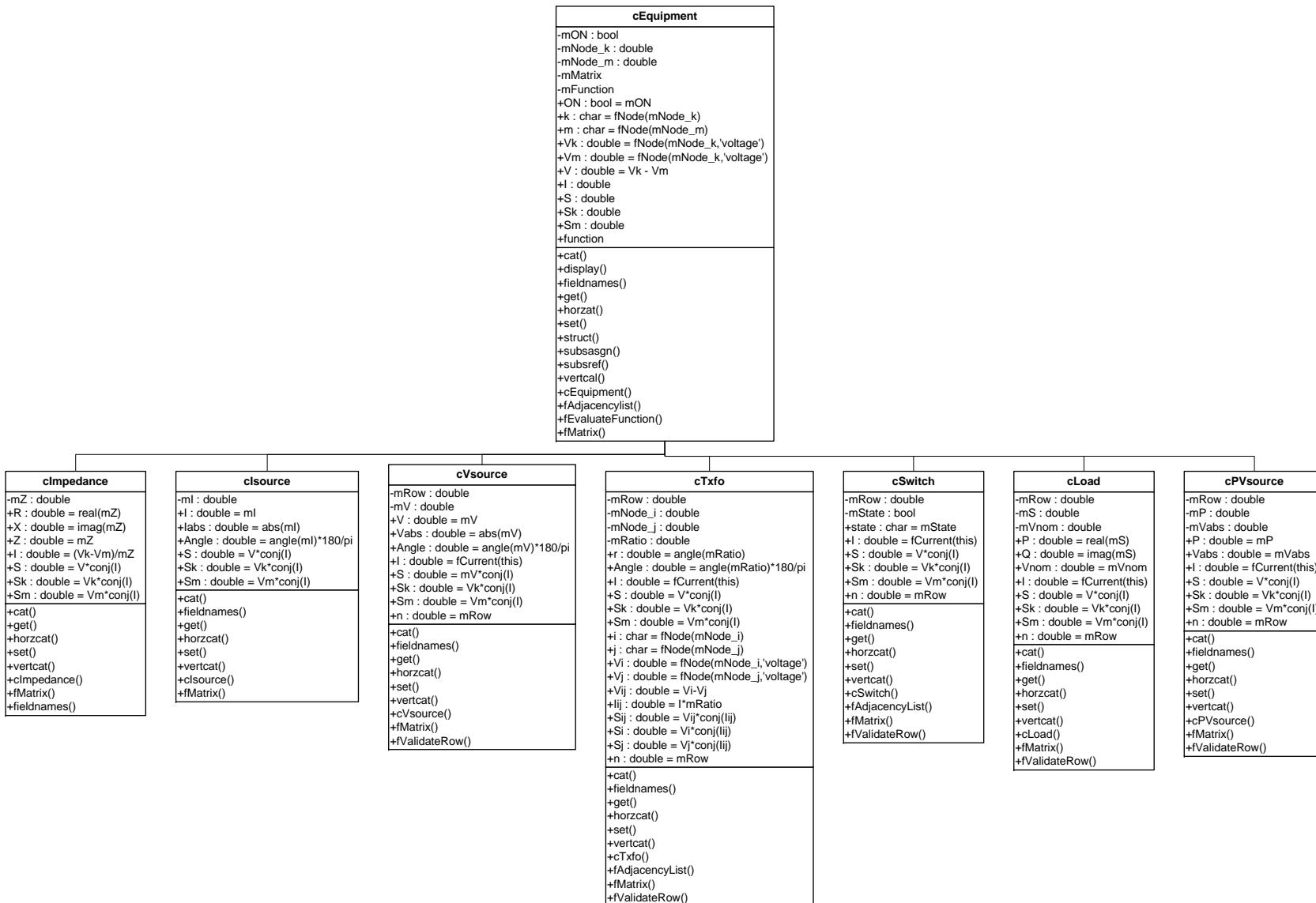


Figure 3.12 : Organisation des classes dans l'outil APREM

3.5.2 La classe cEquipment

La classe *cEquipment* est donc la classe parent de toutes les autres classes de l'outil APREM. Tel qu'expliqué précédemment, les attributs *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction* sont les variables privées de la classe *cEquipment* et ne sont accessibles qu'à travers les variables publiques identifiées à l'aide du signe « + », et les fonctions développées pour la classe telles que *cat*, *display*, *fAdjacencylist*, etc.

Les fonctions publiques disponibles à l'intérieur de la classe *cEquipment* forment trois (3) groupes :

- le constructeur de la classe, *cEquipment*;
- les fonctions créées pour l'outil APREM débutant par le préfixe « *f* »;
- les fonctions built-in de Matlab adaptées aux besoins de la classe tel que *display*, *fieldnames*, *subsref* ou encore *subsasgn*.

Les fonctions *built-in* de Matlab redéfinies

En effet, Matlab offre la possibilité de redéfinir complètement ces fonctions built-in pour qu'elles puissent traiter les objets des nouvelles classes et retourner les résultats désirés. Par exemple, la fonction built-in *fieldnames* retourne les champs contenus dans une structure ou un objet. Si un objet de la classe *cEquipment* est passé en argument dans la fonction built-in *fieldnames* (Figure 3.13), cette dernière retourne le nom des variables privées de la classe, soit les champs de la structure utilisée pour créer la classe *cEquipment*.

```
>> names = builtin('fieldnames',cEquipment)

names =
'mNode_k'
'mNode_m'
'mON'
'mMatrix'
'mFunction'
```

Figure 3.13 : fonction built-in *fieldnames* de Matlab

Dans l'ordre de priorité, en redéfinissant la fonction *fieldnames*, Matlab exécute en priorité la fonction membre de la classe *cEquipment*. Si Matlab ne trouve aucune fonction du même nom dans le répertoire de la classe, il tente de la trouver parmi les fonctions membre des classes parents. La recherche se poursuit ensuite dans le dossier actif (*current directory*) et si Matlab ne trouve pas d'équivalent, il exécute la fonction built-in du même nom.

Telle que redéfinie dans la classe *cEquipment*, mais également pour toutes les autres classes, la fonction *fieldnames* retourne donc le nom des variables publiques (Figure 3.14).

```
>> names = fieldnames(cEquipment)

names =

'ON'
'k'
'm'
'Vk'
'Vm'
'V'
'I'
'S'
'Sk'
'Sm'
'function'
```

Figure 3.14 : fonction *fieldnames* redéfinie dans l'outil APREM

Le même principe est suivi pour adapter les fonctions suivantes à la classe *cEquipment* :

- cat* *cat(DIM, A,B)* exécute, suivant la valeur de DIM, l'équivalent de la concaténation [A,B] ou [A;B] lorsque A et B sont des objets de la classe *cEquipment*. Elle permet de regrouper dans un tableau plusieurs objets de la même classe
- display* affiche le contenu d'un objet de la classe *cEquipment*
- get* *get(A, 'variable')* retourne la valeur de la variable publique passée en argument.
- set* *set(A, 'variable', valeur)* permet d'affecter une valeur à la variable publique passée en argument. Elle se limite aux variables publiques *k*, *m*, *ON*, et *function*, car les autres variables ne sont pas modifiables.

<i>horzcat</i>	<i>horzcat(A,B)</i> exécute l'équivalent de [A B] lorsque A et B sont des objets de la classe <i>cEquipment</i> . Elle permet de regrouper dans un vecteur horizontal plusieurs objets de la même classe
<i>vertcat</i>	<i>vertcat(A,B)</i> exécute l'équivalent de [A;B] lorsque A et B sont des objets de la classe <i>cEquipment</i> . Elle permet de regrouper dans un vecteur vertical plusieurs objets de la même classe
<i>struct</i>	<i>struct(A)</i> convertit l'objet A de la classe <i>cEquipment</i> en une structure de données
<i>subsagn</i>	permet à Matlab d'interpréter l'usage du point(.) dans une opération sur un objet de la classe <i>cEquipment</i> tel que <i>A.k = 'mtl'</i> afin de modifier le nom du nœud k
<i>subref</i>	permet à Matlab d'interpréter l'usage du point(.) dans une opération sur un objet de la classe <i>cEquipment</i> tel que <i>A.k</i> et retourner le nom du nœud k

Le constructeur

Lorsque le constructeur est appelé sans argument, Matlab crée un objet avec les paramètres par défaut de la classe comme montré au début de la section 3.5.1. Il est néanmoins possible d'assigner des noms aux nœuds *k* et *m* en les passant en argument du le constructeur (Figure 3.15).

```
>> EQ = cEquipment('montreal','st_bruno')

EQ =

    ON: 1
    k: 'montreal'
    m: 'st_bruno'
    Vk: 0
    Vm: 0
    V: 0
    I: []
    S: []
    Sk: []
    Sm: []
    function: ''
```

Figure 3.15 : constructeur de la classe *cEquipment*

Les fonctions de l'outil APREM

Trois (3) fonctions propres à l'usage de l'outil APREM sont membres de la classe *cEquipement* :

fAdjacencyList est appelée lors de l'analyse de la topologie du réseau par l'outil APREM afin de déterminer si les nœuds k et m sont adjacents ou non selon la théorie des graphes en fonction du statut de l'équipement (*ON* = 1). La fonction retourne la liste d'adjacence (*adjacency list*) de l'objet dans une structure. Dans l'exemple qui suit, le résultat indique que le nœud *montreal* est adjacent au nœud *st-bruno* et vice versa (Figure 3.16).

```
>> fAdjacencyList(EQ)
ans =
    montreal: {'st_bruno'}
    st_bruno: {'montreal'}
```

Figure 3.16 : fonction *fAdjacencyList* de *cEquipement*

fEvaluateFunction est appelée à chaque itération de algorithme de Newton-Raphson de répartition de puissance afin d'évaluer, au besoin, les fonctions spéciales développées par l'usager pour chacun des équipements sur le réseau.

fMatrix mémorise dans la variable privée *mMatrix* de l'objet les matrices caractérisant l'équipement passé en argument. la fonction retourne également ces matrices.

3.5.3 La classe *cImpedance*

La classe *cImpedance* permet de créer un objet représentant une impédance sur un réseau. Comme sa classe parent, elle contient un attribut privé dénommé *mZ* qui stocke la valeur de l'impédance de l'objet sous forme complexe et, puisqu'elle hérite de la classe *cEquipment*, elle a également accès aux variables privées *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction*. L'accès aux variables publiques permet d'afficher ou modifier les caractéristiques de l'impédance.

<i>R</i>	permet d'afficher ou modifier la valeur résistive de l'impédance
<i>X</i>	permet d'afficher ou modifier la valeur réactive de l'impédance
<i>Z</i>	permet d'afficher ou modifier la valeur complexe de l'impédance
<i>I</i>	retourne le résultat de l'équation $(V_k - V_m)/mZ$
<i>S</i>	retourne le résultat de l'équation V^2/mZ^*
<i>Sk</i>	retourne le résultat de l'équation $V_k \times (V/mZ)^*$
<i>Sm</i>	retourne le résultat de l'équation $V_m \times (V/mZ)^*$

où V est la tension entre les nœuds auxquels est raccordé l'équipement, V_k , la tension du nœud k et V_m la tension au nœud m.

Les fonctions *built-in* de Matlab redéfinies

De la même façon que pour la classe *cEquipment*, certaines fonctions built-in de Matlab sont redéfinies pour pouvoir exécuter les mêmes opérations sur des objets de la classe *cImpedance*. Entre autre, les fonctions *fieldnames*, *get* et *set* sont modifiées pour inclure les nouveaux champs publics *R*, *X*, *Z* de la classe *cImpedance* aux autres paramètres hérités.

Le constructeur

Le constructeur *cImpedance* permet de créer un objet de la classe *cImpedance*. Lorsque le constructeur est appelé sans argument, Matlab crée l'objet avec les paramètres par défaut de la classe. Naturellement, l'objet *cImpedance* hérite également des valeurs initiales de la classe parent. Pour accélérer la définition de l'impédance, il suffit de passer en argument du

constructeur les noms des nœuds k et m , et les valeurs de R et X de l'impédance. Les nouveaux objets de l'outil APREM sont toujours initialisés à $ON=1$ lors de la création. Dans l'exemple de la Figure 3.17, l'impédance $Z = 10+j20$ est créée entre les nœuds *montreal* et *st_bruno*.

```
>> Z = cImpedance('montreal','st_bruno',10,20)

Z =

    ON: 1
    k: 'montreal'
    m: 'st_bruno'
    Vk: 0
    Vm: 0
    V: 0
    I: 0
    S: 0
    Sk: 0
    Sm: 0
    function: ''
    R: 10
    X: 20

>> whos
  Name      Size            Bytes  Class
  Z          1x1              2609  cImpedance object
```

Figure 3.17 : construction de la classe *cImpedance*

Les fonctions de l'outil APREM

La fonction *fMatrix* de la classe *cImpedance* est essentielle au logiciel APREM. Elle construit les matrices Y_n et F_n de l'objet et leur assigne des coordonnées numériques correspondant au numéro des nœuds k et m . Les coordonnées permettent ensuite d'introduire les matrices de l'objet à la bonne place dans la matrice d'admittances du réseau. Lors de la création de l'impédance Z dans le code précédent, l'outil APREM attribue automatiquement les numéros 1 et 2 aux nœuds *montreal* et *st-bruno*.

L'exemple de la Figure 3.18 permet donc de récupérer les matrices de l'impédance Z et les stocke dans la structure *MatZ*. Seules les matrices d'admittances \mathbf{Yn} et d'injection de courant

$\mathbf{f}_n = 0$ sont requises pour représenter l'impédance dans le réseau. La théorie présentée à la section 3.2.1 est utilisée ici pour construire ces matrices :

$$\mathbf{Z} = 10 + j20, \text{ donc } \mathbf{Z}^{-1} = \mathbf{Y} = 0.02 - j0.04 \quad (3.160)$$

$$\mathbf{Y} = \begin{bmatrix} 0.02 & 0.04 \\ -0.04 & 0.02 \end{bmatrix} \quad (3.161)$$

$$\mathbf{Y_n} = \begin{bmatrix} \mathbf{Y}_1 & -\mathbf{Y}_{12} \\ -\mathbf{Y}_{12} & \mathbf{Y}_2 \end{bmatrix} \quad (3.162)$$

$$\mathbf{f}_n = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (3.163)$$

Les numéros des nœuds k et m déterminent alors l'emplacement dans le réseau des matrices et si le paramètre $ON=0$, alors $\mathbf{Yn} = []$

```
>> [Z MatZ]=fMatrix(Z);
>> MatZ

MatZ =

Yn: {4x3 cell}
Fn: {2x3 cell}

>> MatZ.Yn

ans =

[1]      [1]      [2x2 double]
[2]      [2]      [2x2 double]
[1]      [2]      [2x2 double]
[2]      [1]      [2x2 double]

>> MatZ.Yn(1,3)

ans =

    0.0200    0.0400
   -0.0400    0.0200
```

Figure 3.18 : fonction *fMatrix* de la classe *cImpedance*

La nomenclature utilisée pour organiser et gérer les matrices des impédances et également de tous les autres équipements de l'outil APREM, fait l'usage des *cell arrays* de Matlab. Elle se présente comme suit :

$$\mathbf{Y}_k = \{k \quad k \quad [2 \times 2]\} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 0.02 & 0.04 \\ -0.04 & 0.02 \end{bmatrix} \end{Bmatrix} \quad (3.164)$$

$$\mathbf{Y}_{km} = \begin{Bmatrix} m & m & [2 \times 2] \\ k & m & [2 \times 2] \end{Bmatrix} = \begin{Bmatrix} 2 & 2 & \begin{bmatrix} 0.02 & 0.04 \\ -0.04 & 0.02 \end{bmatrix} \\ 1 & 2 & \begin{bmatrix} -0.02 & -0.04 \\ 0.04 & -0.02 \end{bmatrix} \end{Bmatrix} \quad (3.165)$$

$$\mathbf{Y}_m = \{m \quad k \quad [2 \times 2]\} = \begin{Bmatrix} 2 & 1 & \begin{bmatrix} -0.02 & -0.04 \\ 0.04 & -0.02 \end{bmatrix} \end{Bmatrix} \quad (3.166)$$

Ainsi, s'il est nécessaire d'introduire la matrice \mathbf{Y}_{25} de dimension 2×2 d'un objet entre les nœuds 2 et 5 d'un réseau, l'opération dans Matlab se résumera à :

$$\mathbf{Yn}\{2,5\} = \mathbf{Yn}\{2,5\} + [\mathbf{Y}_{25}] \quad (3.167)$$

$$\mathbf{Yn}\{5,2\} = \mathbf{Yn}\{5,2\} + [\mathbf{Y}_{25}] \quad (3.168)$$

L'utilisation des *cell arrays* permet de s'affranchir de la gestion des colonnes et des lignes requise pour imbriquer les matrices 2×2 dans un *array*. Dans un *array*, les nœuds 2 et 5 correspondraient aux lignes et colonnes 3 à 4 et 9 à 10. La même opération s'écrirait alors

$$\mathbf{Yn}(2 * 2 - 1:2 * 2, 5 * 2 - 1:5 * 2) = \mathbf{Yn}(2 * 2 - 1:2 * 2, 5 * 2 - 1:5 * 2) + [\mathbf{Y}_{25}] \quad (3.169)$$

$$\mathbf{Yn}(5 * 2 - 1:5 * 2, 2 * 2 - 1:2 * 2) = \mathbf{Yn}(5 * 2 - 1:5 * 2, 2 * 2 - 1:2 * 2) + [\mathbf{Y}_{25}] \quad (3.170)$$

Il est vrai cependant que l'utilisation des *cell arrays* limite les performances de l'outil APREM. Cependant, l'usage de cette fonctionnalité avait pour but de simplifier le développement de l'outil recherche et afin de le rendre accessible à un utilisateur ayant peu d'expérience en programmation. Cela n'empêche que les développements futurs ne devraient pas se restreindre à ce choix d'architecture logiciel et préconiser l'optimisation du code, lorsque les performances deviendront un enjeu dans le développement.

3.5.4 La classe *cIsource*

La classe *cIsource* permet de créer un objet représentant une source de courant sur un réseau. Sa variable privée *mI* stocke la valeur du courant assignée de la source et elle hérite également des variables privées *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction* de sa classe parent. L'accès aux variables publiques permet d'afficher ou modifier les caractéristiques de la source de courant.

<i>I</i>	permet d'afficher ou modifier la valeur complexe du courant de la source
<i>Iabs</i>	permet d'afficher ou modifier l'amplitude du courant assigné de la source
<i>Angle</i>	permet d'afficher ou modifier l'angle du courant assigné de la source
<i>S</i>	retourne le résultat de l'équation $V \times mI^*$
<i>Sk</i>	retourne le résultat de l'équation $V_k \times mI^*$
<i>Sm</i>	retourne le résultat de l'équation $V_m \times mI^*$

Les fonctions *built-in* de Matlab redéfinies

Bien que la classe *cIsource* hérite des fonctions built in de Matlab redéfinies dans la classe *cEquipment*, certaines sont adaptées pour les besoins de la classe *cIsource*. Entre autres, les fonctions *fieldnames*, *get* et *set* sont modifiées pour inclure les nouveaux champs publics *Iabs*, *Angle* de la classe *cIsource* aux autres paramètres hérités.

Le constructeur

Le constructeur *cIsource* permet de créer un objet de la classe *cIsource*. En appelant le constructeur sans passer d'argument, l'usager crée une source de courant avec les paramètres par défaut de la classe. Il peut également accélérer la définition de la source en passant les noms des nœuds *k* et *m*, et les valeurs de *Iabs* et *Angle* en argument du constructeur. Dans l'exemple de la Figure 3.19, l'injection de courant a lieu au nœud *montreal*

$$I_{\text{src}} = 5e^{-j30 \times \frac{\pi}{180}} \quad (3.171)$$

```

>> Isrc = cIsource('montreal','GND',5,-30)

Isrc =
    ON: 1
    k: 'montreal'
    m: 'GND'
    Vk: 0
    Vm: 0
    V: 0
    I: 4.3301 - 2.5000i
    S: 0
    Sk: 0
    Sm: 0
    function: ''
    Iabs: 5
    Angle: -30.0000

>> whos
  Name      Size            Bytes  Class
  Isrc      1x1             1361   cIsource object

```

Figure 3.19 : constructeur de la classe *cIsource*

Les fonctions de l'outil APREM

La fonction *fMatrix* de la classe *cIsource* construit la matrice **Fi** suivant les coordonnées numériques des nœuds *k* et *m*. L'exemple qui suit permet donc de récupérer la matrice de la source de courant *Isrc* et la stocker dans la structure *MatIsrc*. Seule la matrice d'injection de courant *Fi* est requise pour représenter la source de courant dans le réseau. La théorie présentée à la section 3.2.3 est utilisée ici pour construire la matrice (Figure 3.20)

```

>> [Isrc MatIsrc]=fMatrix(Isrc);
>> MatIsrc

MatIsrc =

Fi: {[1]    [1]    [2x1 double]}

>> MatIsrc.Fi{1,3}

ans =

4.3301
-2.5000

```

Figure 3.20 : fonction *fMatrix* de la classe *cIsource*

La matrice **Fi** est stockée dans une *cell array* sous la forme suivante :

$$\mathbf{Fi} = \{k \quad 1 \quad [+I_{src}]\} = \left\{ 1 \quad 1 \quad \begin{bmatrix} 4.3301 \\ -2.5 \end{bmatrix} \right\} \quad (3.172)$$

3.5.5 La classe *cVsource*

La classe *cVsource* permet de créer un objet représentant une source de tension sur un réseau. Ses variables privées *mV* et *mRow* stockent respectivement la valeur complexe de la tension assignée et le numéro de la source de tension. La classe hérite également des variables privées *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction* de sa classe parent. L'accès aux variables publiques permet d'afficher ou modifier les caractéristiques de la source de courant.

<i>V</i>	permet d'afficher ou modifier la valeur complexe de la tension de la source
<i>Vabs</i>	permet d'afficher ou modifier l'amplitude de la tension assignée de la source
<i>Angle</i>	permet d'afficher ou modifier l'angle de la tension assignée de la source
<i>I</i>	retourne le courant de la source de tension en appelant <i>fCurrent(this)</i>
<i>S</i>	retourne le résultat de l'équation $V \times I^*$
<i>Sk</i>	retourne le résultat de l'équation $V_k \times I^*$
<i>Sm</i>	retourne le résultat de l'équation $V_m \times I^*$
<i>n</i>	retourne le numéro <i>mRow</i> de la source

Les fonctions *built-in* de Matlab redéfinies

cVsource hérite des fonctions *built in* de Matlab redéfinies dans la classe *cEquipment* et les fonctions *fieldnames*, *get* et *set* sont adaptées aux besoins de la classe *cVsource* pour inclure les nouveaux champs publics *Vabs*, *Angle* et *n*.

Le constructeur

Le constructeur *cVsource* permet de créer un objet de la classe *cVsource*. L'usager crée une source de tension avec les paramètres par défaut de la classe ou accélère la définition de la source en passant les noms des nœuds *k* et *m* et les valeurs de *Vabs* et *Angle* en argument du constructeur. Dans l'exemple de la Figure 3.21, la source de tension est connectée entre les nœuds *montreal* et *st_bruno*

$$V_s = 120e^{j12 \times \frac{\pi}{180}} = 117.38 + j24.949 = \begin{bmatrix} 117.38 \\ 24.949 \end{bmatrix} \quad (3.173)$$

```
>> Vs = cVsource('montreal','st_bruno',120,12)

Vs =
    ON: 1
    k: 'montreal'
    m: 'st_bruno'
    Vk: 0
    Vm: 0
    V: 0
    I: 0
    S: 0
    Sk: 0
    Sm: 0
    function: ''
    Vabs: 120.0000
    Angle: 12
    n: 1

>> whos
  Name      Size            Bytes  Class
  Vs       1x1             2653  cVsource object
```

Figure 3.21 : constructeur de la classe *cVsource*

Les fonctions de l'outil APREM

La fonction *fMatrix* de la classe *cVsource* construit les matrices *Vr*, *Vc* et *Fv* suivant les coordonnées numériques des nœuds *k* et *m*. L'exemple qui suit permet donc de récupérer les matrices de la source de tension **Vs** et les stocker dans la structure *MatVs* (Figure 3.22). Les trois (3) matrices **Vr**, **Vc** et **Fv** sont requises pour représenter la source de tension dans le réseau. La théorie présentée à la section 3.2.2 est utilisée ici pour construire les matrices :

$$\begin{bmatrix} \mathbf{Y}_n & \mathbf{V}_c \\ \mathbf{V}_r & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} \mathbf{V}_n \\ \mathbf{I}_V \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{V}_s \end{bmatrix} \quad (3.174)$$

$$\begin{bmatrix} Y_k & \mathbf{0} & \mathbf{u} \\ \mathbf{0} & Y_m & -\mathbf{u} \\ \mathbf{u} & -\mathbf{u} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_V \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ f_V \end{bmatrix} \quad (3.175)$$

```

>> [Vs MatVs]=fMatrix(Vs);
>> MatVs

MatVs =
    Vr: {2x3 cell}
    Vc: {2x3 cell}
    Fv: {[1] [1] [2x1 double]}

>> MatVs.Vr

ans =
    [1]      [1]      [2x2 double]
    [1]      [2]      [2x2 double]

>> MatVs.Vr(1,3)

ans =
    1      0
    0      1

```

Figure 3.22 : fonction *fMatrix* de la classe *cVsource*

Les matrices **Vr**, **Vc** et **Fv** sont stockées dans des *cell arrays* sous la forme suivante :

$$V_r = \left\{ \begin{array}{ccc} mRow & k & [+u] \\ mRow & m & [-u] \end{array} \right\} = \left\{ \begin{array}{ccc} 1 & 1 & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 1 & 2 & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \end{array} \right\} \quad (3.176)$$

$$V_c = \left\{ \begin{array}{ccc} mRow & k & [+u] \\ mRow & m & [-u] \end{array} \right\} = \left\{ \begin{array}{ccc} 1 & 1 & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 2 & 1 & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \end{array} \right\} \quad (3.177)$$

$$\mathbf{Fv} = \begin{Bmatrix} k & 1 & [+Vs] \\ m & 1 & [-Vs] \end{Bmatrix} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 117.38 \\ 24.949 \end{bmatrix} \\ 2 & 1 & \begin{bmatrix} -117.38 \\ -24.949 \end{bmatrix} \end{Bmatrix} \quad (3.178)$$

Finalement, la fonction *fValideRow* est appelée avant de bâtir la matrice du réseau afin de vérifier s'il n'existe pas des sources de tension ayant le même numéro *mRow*. Si tel est le cas, la fonction assigne un nouveau numéro automatiquement et reconstruit les matrices.

3.5.6 La classe cTxfo

La classe *cTxfo* permet de créer un objet représentant un transformateur idéal sur un réseau. Ses variables privées *mRow*, *mNode_i*, *mNode_j* et *mRatio* stockent respectivement le numéro du transformateur, les numéros des nœuds *i* et *j* et le *ratio* (en nombre complexe) du transformateur. La classe hérite également des variables privées *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction* de sa classe parent. L'accès aux variables publiques permet d'afficher ou modifier le transformateur

<i>r</i>	permet d'afficher ou modifier le ratio du transformateur
<i>Angle</i>	permet d'afficher ou modifier l'angle de déphasage du transformateur
<i>I</i>	retourne le courant au primaire du transformateur
<i>i</i>	permet d'afficher ou modifier le nœud <i>i</i> au secondaire du transformateur
<i>j</i>	permet d'afficher ou modifier le nœud <i>j</i> au secondaire du transformateur
<i>Vi</i>	retourne la tension au nœud <i>i</i>
<i>Vj</i>	retourne la tension au nœud <i>j</i>
<i>Vij</i>	retourne $V_i - V_j$
<i>Iij</i>	retourne le courant au secondaire du transformateur
<i>S</i>	retourne $V \times I^*$
<i>Sk</i>	retourne $V_k \times I^*$
<i>Sm</i>	retourne $V_m \times I^*$
<i>Sij</i>	retourne $V_{ij} \times I^* \times mRatio$
<i>Si</i>	retourne $V_i \times I^* \times mRatio$
<i>Sj</i>	retourne $V_j \times I^* \times mRatio$
<i>n</i>	retourne le numéro <i>mRow</i> du transformateur

Les fonctions *built-in* de Matlab redéfinies

cTxfo hérite des fonctions built-in redéfinies dans la classe *cEquipment*, certaines et les fonctions *fieldnames*, *get* et *set* sont modifiées pour inclure les nouveaux champs publics de la classe *cTxfo* aux autres paramètres hérités, soit *r*, *Angle*, *i*, *j*, *Vi*, *Vj*, *Vij*, *Iij*, *Sij*, *Si*, *Sj*, *n*.

Le constructeur

Le constructeur *cTxfo* permet de créer un objet de la classe *cTxfo*. En appelant le constructeur sans passer aucun argument, l'usager crée un transformateur idéal avec les paramètres par défaut de la classe. Il peut également accélérer la définition du transformateur en passant les noms des nœuds *k*, *m*, *i*, *j* et les valeurs de *r* et *Angle* en argument du constructeur. Dans l'exemple de la Figure 3.23, le transformateur est connecté entre les nœuds *montreal*, *st_bruno*, *quebec* et *ottawa*, avec *r* = 25/12.4 et *Angle* = 30.

$$g = 25/12.4 e^{j30 \times \frac{\pi}{180}} = 1.7460 + j1.0081 \quad (3.179)$$

$$\mathbf{g} = \begin{bmatrix} 1.7460 & -1.0081 \\ 1.0081 & 1.7460 \end{bmatrix} \quad (3.180)$$

Les fonctions de l'outil APREM

La fonction *fMatrix* de la classe *cTxfo* construit les matrices **Dr**, **Dc** et **F_D** suivant les coordonnées numériques des nœuds *k*, *m*, *i* et *j*. L'exemple qui suit permet donc de récupérer les matrices du transformateur **D** et les stocker dans la structure *MatD* (Figure 3.24). Les trois (3) matrices **Dr**, **Dc** et **F_D** sont requises pour représenter le transformateur dans le réseau. La théorie présentée à la section 3.2.4 est utilisée ici pour construire les matrices :

$$\begin{bmatrix} \mathbf{Y}_n & \mathbf{D}_c \\ \mathbf{D}_r & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} \mathbf{V}_n \\ \mathbf{I}_D \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{f}_D \end{bmatrix} \quad (3.181)$$

$$\begin{bmatrix} Y_k & 0 & 0 & 0 & u \\ 0 & Y_m & 0 & 0 & -u \\ 0 & 0 & Y_i & 0 & -g^* \\ 0 & 0 & 0 & Y_j & g^* \\ u & -u & -g & g & 0 \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ V_i \\ V_j \\ I_D \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ f_D \end{bmatrix} \quad (3.182)$$

```
>> D = cTxfo('montreal','st_bruno','quebec','ottawa',25/12.4,30)

D =

    ON: 1
    k: 'montreal'
    m: 'st_bruno'
    Vk: 0
    Vm: 0
    V: 0
    I: 0
    S: 0
    Sk: 0
    Sm: 0
    function: ''
        i: 'quebec'
        j: 'ottawa'
    Vi: 0
    Vj: 0
    Vij: 0
    Iij: 0
    Sij: 0
    Si: 0
    Sj: 0
    r: 2.0161
    Angle: 30.0000
    n: 1

>> whos
  Name      Size            Bytes  Class
  D          1x1             3829  cTxfo object
```

Figure 3.23 : constructeur de la classe *cTxfo*

```

>> [D MatD]=fMatrix(D);
>> MatD

MatD =
Dr: {4x3 cell}
Dc: {4x3 cell}
Fd: {[1] [1] [2x1 double]}

>> MatD.Dr

ans =
[1] [1] [2x2 double]
[1] [2] [2x2 double]
[1] [3] [2x2 double]
[1] [4] [2x2 double]

>> MatD.Dr(3,3)

ans =
-1.7460 1.0081
-1.0081 -1.7460

```

Figure 3.24 : fonction *fMatrix* de la classe *cTxfo*

Les matrices **Dr**, **Dc** et **Fd** sont stockées dans des *cell arrays* sous la forme suivante:

$$\mathbf{Dr} = \left\{ \begin{array}{ccc} \text{mRow} & k & [+u] \\ \text{mRow} & m & [-u] \\ \text{mRow} & i & [-g] \\ \text{mRow} & j & [+g] \end{array} \right\} = \left\{ \begin{array}{cc} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{array} \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \\ -1.7460 & 1.0081 \\ -1.0081 & -1.7460 \\ 1.7460 & -1.0081 \\ 1.0081 & 1.7460 \end{bmatrix} \right\} \quad (3.183)$$

$$\mathbf{Vc} = \begin{Bmatrix} k & mRow & [+u] \\ m & mRow & [-u] \\ i & mRow & [-g]^* \\ j & mRow & [+g]^* \end{Bmatrix} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 2 & 1 & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \\ 3 & 1 & \begin{bmatrix} -1.7460 & -1.0081 \\ 1.0081 & -1.7460 \end{bmatrix} \\ 4 & 1 & \begin{bmatrix} 1.7460 & 1.0081 \\ -1.0081 & 1.7460 \end{bmatrix} \end{Bmatrix} \quad (3.184)$$

$$\mathbf{F_D} = \begin{Bmatrix} k & 1 & [0] \\ m & 1 & [0] \\ i & 1 & [0] \\ j & 1 & [0] \end{Bmatrix} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ 2 & 1 & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ 3 & 1 & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ 4 & 1 & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{Bmatrix} \quad (3.185)$$

fAdjacencyList est appelée lors de l'analyse de la topologie du réseau par l'outil APREM afin de déterminer si les noeuds k , m , i et j sont adjacents ou non selon la théorie des graphes dépendant le statut du transformateur ($ON = 1$ ou 0). La fonction retourne la liste d'adjacence (*adjacency list*) du transformateur dans une structure.

```
>> fAdjacencyList(D)

ans =

    montreal: ('st_bruno'  'quebec'  'ottawa')
    st_bruno: ('montreal'  'ottawa'  'quebec')
    quebec: ('ottawa'  'montreal'  'st_bruno')
    ottawa: ('quebec'  'st_bruno'  'montreal')
```

Figure 3.25 : fonction *fAdjencyList* de la classe *cTxfo*

fValideRow est appelée avant de bâtir la matrice du réseau afin de vérifier s'il n'existe pas de transformateurs ayant le même numéro. Si tel est le cas, la fonction assigne un nouveau numéro automatiquement et reconstruit les matrices.

3.5.7 La classe cSwitch

La classe *cSwitch* permet de créer un objet représentant un interrupteur sur un réseau. Ses variables privées *mState* et *mRow* stockent respectivement la position de l'interrupteur (ouvert/fermé) et son numéro. La classe hérite également des variables privées *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction* de sa classe parent. L'accès aux variables publiques permet d'afficher ou modifier les caractéristiques de l'interrupteur

<i>state</i>	permet d'afficher ou modifier la position de l'interrupteur (ouvert/fermé)
<i>I</i>	retourne le courant qui circule dans l'interrupteur appelant <i>fCurrent(this)</i>
<i>S</i>	retourne le résultat de l'équation $V \times I^*$
<i>Sk</i>	retourne le résultat de l'équation $V_k \times I^*$
<i>Sm</i>	retourne le résultat de l'équation $V_m \times I^*$
<i>n</i>	retourne le numéro <i>mRow</i> de l'interrupteur

Les fonctions *built-in* de Matlab redéfinies

Bien que la classe *cSwitch* hérite des fonctions built-in redéfinies dans la classe *cEquipment*, certaines sont adaptées pour les besoins de la classe *cSwitch*, dont les fonctions *fieldnames*, *get* et *set* pour inclure les nouveaux champs publics *state* et *n* aux autres paramètres hérités.

Le constructeur

Le constructeur *cSwitch* permet de créer un objet de la classe *cSwitch*. En appelant le constructeur sans passer aucun argument, l'usager crée un interrupteur avec les paramètres par défaut de la classe. Il peut également accélérer la définition de l'interrupteur en passant les noms des nœuds *k* et *m*, et la position de l'interrupteur *state* en argument du constructeur. Dans l'exemple de Figure 3.26, l'interrupteur est connecté entre les nœuds *montreal* et *st_bruno*. L'interrupteur est fermé par défaut.

```

>> S = cSwitch('montreal',...
'st_bruno','CLOSE')

S =

    ON: 1
    k: 'montreal'
    m: 'st_bruno'
    Vk: 0
    Vm: 0
    V: 0
    I: 0
    S: 0
    Sk: 0
    Sm: 0
    function: ''
    state: 'CLOSE'
    n: 2

```

Figure 3.26 : constructeur de la classe *cSwitch*

Les fonctions de l'outil APREM

La fonction *fMatrix* de la classe *cSwitch* construit les matrices **S_r**, **S_c**, **S_d** et **F_s** suivant les coordonnées numériques des nœuds *k* et *m*. L'exemple de la Figure 3.27 permet donc de récupérer les matrices de l'interrupteur **S** et les stocker dans la structure *MatS*. Les quatre (4) matrices **S_r**, **S_c**, **S_d** et **F_s** sont requises pour représenter l'interrupteur dans le réseau. La théorie présentée à la section 3.2.5 est utilisée ici pour construire les matrices :

$$\begin{bmatrix} \mathbf{Y}_n & \mathbf{S}_c \\ \mathbf{S}_r & \mathbf{S}_d \end{bmatrix} \times \begin{bmatrix} \mathbf{V}_n \\ \mathbf{I}_s \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{f}_s \end{bmatrix} \quad (3.186)$$

Si l'interrupteur est fermé,

$$\begin{bmatrix} \mathbf{Y}_k & \mathbf{0} & \mathbf{u} \\ \mathbf{0} & \mathbf{Y}_m & -\mathbf{u} \\ \mathbf{u} & -\mathbf{u} & \mathbf{0} \end{bmatrix} \times \begin{bmatrix} \mathbf{V}_k \\ \mathbf{V}_m \\ \mathbf{I}_s \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{f}_s \end{bmatrix} \quad (3.187)$$

Si l'interrupteur est ouvert,

$$\begin{bmatrix} Y_k & -Y_{km} & \mathbf{0} \\ -Y_{km} & Y_m & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & u \end{bmatrix} \times \begin{bmatrix} V_k \\ V_m \\ I_S \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ f_S \end{bmatrix} \quad (3.188)$$

```
>> [S MatS]=fMatrix(S);
>> MatS
```

```
MatS =
Sr: {2x3 cell}
Sc: {2x3 cell}
Sd: {[1] [1] [2x2 double]}
Fs: {[1] [1] [2x1 double]}
```

```
>> MatS.Sr
```

```
ans =
[1] [1] [2x2 double]
[1] [2] [2x2 double]
```

```
>> MatS.Sr{2,3}
```

```
ans =
-1 0
0 -1
```

Figure 3.27 : fonction *fMatrix* de la classe *cSwitch*

Les matrices **Sr**, **Sc**, **Sd** et **Fs** sont stockées dans des *cells* sous la forme suivante :

$$\mathbf{Sr} = \left\{ \begin{array}{ccc} \text{mRow} & k & [+u] \\ \text{mRow} & m & [-u] \end{array} \right\} = \left\{ \begin{array}{ccc} 1 & 1 & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 1 & 2 & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \end{array} \right\} \quad (3.189)$$

$$\mathbf{Sc} = \begin{Bmatrix} k & mRow & [+u] \\ m & mRow & [-u] \end{Bmatrix} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ 2 & 1 & \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \end{Bmatrix} \quad (3.190)$$

$$\mathbf{Sd} = \{mRow \quad mRow \quad [0]\} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{Bmatrix} \quad (3.191)$$

$$\mathbf{Fs} = \begin{Bmatrix} k & 1 & [0] \\ m & 1 & [0] \end{Bmatrix} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ 2 & 1 & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{Bmatrix} \quad (3.192)$$

Si l'interrupteur est ouvert, les matrices \mathbf{Sr} , \mathbf{Sc} , \mathbf{Sd} deviennent :

$$\mathbf{Sr} = \begin{Bmatrix} mRow & k & [0] \\ mRow & m & [0] \end{Bmatrix} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ 1 & 2 & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{Bmatrix} \quad (3.193)$$

$$\mathbf{Sc} = \begin{Bmatrix} k & mRow & [0] \\ m & mRow & [0] \end{Bmatrix} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \\ 2 & 1 & \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \end{Bmatrix} \quad (3.194)$$

$$\mathbf{Sd} = \{mRow \quad mRow \quad [+u]\} = \begin{Bmatrix} 1 & 1 & \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{Bmatrix} \quad (3.195)$$

fAdjacencyList est appelée lors de l'analyse de la topologie du réseau par l'outil APREM afin de déterminer si les nœuds k et m sont adjacents ou non selon la théorie des graphes en fonction du état et de la position de l'interrupteur. La fonction retourne la liste d'adjacence (*adjacency list*) de l'interrupteur dans une structure.

fValideRow est appelée avant de bâtir la matrice du réseau afin de vérifier s'il n'existe pas d'interrupteurs ayant le même numéro. Si tel est le cas, la fonction assigne un nouveau numéro automatiquement et reconstruit les matrices.

3.5.8 La classe cLoad

La classe *cLoad* permet de créer un objet représentant une charge PQ sur un réseau. Ses variables privées *mS* et *mRow* stockent respectivement la puissance (en nombre complexe) de la charge et son numéro. La classe hérite également des variables privées *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction* de sa classe parent. L'accès aux variables publiques permet d'afficher ou modifier les caractéristiques de la charge

<i>P</i>	permet d'afficher ou modifier la puissance active de la charge
<i>Q</i>	permet d'afficher ou modifier la puissance réactive de la charge
<i>S</i>	permet d'afficher ou modifier la puissance complexe de la charge
<i>Vnom</i>	permet d'afficher ou modifier la tension nominale de la charge
<i>I</i>	retourne le courant de la charge en appelant <i>fCurrent(this)</i>
<i>Sk</i>	retourne le résultat de l'équation $V_k \times I^*$
<i>Sm</i>	retourne le résultat de l'équation $V_m \times I^*$
<i>n</i>	retourne le numéro <i>mRow</i> de la charge

Les fonctions *built-in* de Matlab redéfinies

La classe *cLoad* hérite des fonctions built-in redéfinies dans la classe *cEquipment* et certaines dont *fieldnames*, *get* et *set* sont modifiées pour inclure les nouveaux champs publics *P*, *Q* et *n* de la classe *cLoad* aux autres paramètres hérités.

Le constructeur

Le constructeur *cLoad* permet de créer un objet de la classe *cLoad*. En appelant le constructeur sans passer aucun argument, l'usager crée une charge avec les paramètres par défaut de la classe. Il peut également définir la charge en passant les noms des nœuds *k* et *m*, et les paramètres *P*, *Q* et *Vnom* en argument du constructeur. Dans l'exemple Figure 3.28, la charge est connectée au nœud *st_bruno*,

$$P = 10 \times 10^6, Q = 3 \times 10^6 \quad \text{et} \quad Vnom = 25 \times 10^3 \quad (3.196)$$

$$S = 10 \times 10^6 + j3 \times 10^6 \quad (3.197)$$

```

>> L = cLoad('st_bruno','GND',10e6,3e6,25e3)

L =

    ON: 1
    k: 'st_bruno'
    m: 'GND'
    Vk: 0
    Vm: 0
    V: 0
    I: 0
    S: 0
    Sk: 0
    Sm: 0
    function: ''
    P: 10000000
    Q: 3000000
    Vnom: 25000
    n: 1

>> whos
  Name      Size            Bytes  Class
  L         1x1              1289   cLoad object

```

Figure 3.28 : constructeur de la classe *cLoad*

Les fonctions de l'outil APREM

La fonction *fMatrix* de la classe *cLoad* construit les matrices \mathbf{C}_{LPQ} , \mathbf{D}_{LPQ} , \mathbf{A}_{IL} et \mathbf{f}_{LPQ} suivant les coordonnées numériques des nœuds *k* et *m*. Dans l'exemple Figure 3.29, un petit réseau est monté afin de calculer le courant de la charge *L* par écoulement de puissance, sans quoi, les matrices \mathbf{C}_{LPQ} et \mathbf{D}_{LPQ} ne peuvent être évaluées. Les matrices sont ensuite stockées dans la structure *MatL*. Les quatre (4) matrices \mathbf{C}_{LPQ} , \mathbf{D}_{LPQ} , \mathbf{A}_{IL} et \mathbf{f}_{LPQ} sont requises pour représenter la charge dans le réseau. La théorie présentée à la section 3.3.1 est utilisée ici pour construire les matrices :

$$\begin{bmatrix} \mathbf{Y}_k & \mathbf{A}_{IL} \\ \mathbf{C}_{LPQ} & \mathbf{D}_{LPQ} \end{bmatrix} \times \begin{bmatrix} \Delta \mathbf{V}_k \\ \Delta \mathbf{I}_L \end{bmatrix} = \begin{bmatrix} \Delta \mathbf{f}_n \\ \Delta \mathbf{f}_{LPQ} \end{bmatrix} \quad (3.198)$$

```
>> Vs = cVsource('montreal','GND',25e3,30);
>> Z = cImpedance('montreal','st_bruno',0.1,0.2);
>> L = cLoad('st_bruno','GND',10e6,3e6,25e3);
>> reseau.L = L; reseau.Z = Z; reseau.Vs = Vs;
>> reseau = fLoadFlow(reseau);
```

Figure 3.29 : réseau test pour initialiser l'objet de la classe *cLoad*

Le code Figure 3.29 crée la structure *reseau* composée d'une source *Vs*, d'une impédance *Z* et de la charge *L*. La fonction *fLoadFlow* réalise ensuite le calcul d'écoulement de puissance du réseau et trouve les tensions des nœuds ainsi que les courants circulant dans la charge. La fonction *fMatrix* retourne ainsi les matrices de la charge *L* calculées à la dernière itération.

```
>> [L MatL] = fMatrix(L,'update');
>> MatL

MatL =

    AIL: {[2]  [1]  [2x2 double]}
    CLPQ: {[1]  [2]  [2x2 double]}
    DLPQ: {[1]  [1]  [2x2 double]}
    FLPQ: {[1]  [1]  [2x1 double]}

>> MatL.CLPQ(1,3)

ans =

    407.7186    95.2130
   -95.2130   407.7186

>> L.I

ans =

    4.0772e+002 +9.5213e+001i

>> L.V

ans =

    2.1629e+004 +1.2409e+004i
```

Figure 3.30 : fonction *fMatrix* de la classe *cLoad*

Toujours selon la même nomenclature, les matrices \mathbf{C}_{LPQ} , \mathbf{D}_{LPQ} , \mathbf{AIL} et \mathbf{f}_{LPQ} sont stockées sous forme de *cells* dans l'objet :

$$\mathbf{C}_{LPQ} = \left\{ mRow \quad k \quad \begin{bmatrix} I_{LR} & I_{LI} \\ -I_{LI} & I_{LR} \end{bmatrix} \right\} = \left\{ 1 \quad 2 \quad \begin{bmatrix} 407.7186 & 95.2130 \\ 95.2130 & 407.7186 \end{bmatrix} \right\} \quad (3.199)$$

$$\mathbf{D}_{LPQ} = \left\{ mRow \quad mRow \quad \begin{bmatrix} V_{kR} & V_{kI} \\ V_{kI} & -V_{kR} \end{bmatrix} \right\} = \left\{ 1 \quad 1 \quad \begin{bmatrix} 21.6e3 & 12.4e3 \\ 12.4e3 & -21.6e3 \end{bmatrix} \right\} \quad (3.200)$$

$$\mathbf{AIL} = \{ mRow \quad k \quad [+u] \} = \left\{ 1 \quad 2 \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right\} \quad (3.201)$$

$$\mathbf{F}_{LPQ} = \left\{ k \quad 1 \quad \begin{bmatrix} P \\ Q \end{bmatrix} \right\} = \left\{ 2 \quad 1 \quad \begin{bmatrix} 10e6 \\ 3e6 \end{bmatrix} \right\} \quad (3.202)$$

fValidateRow est appelée avant de bâtir la matrice du réseau afin de vérifier s'il n'existe pas de charge ayant le même numéro *mRow*. Si tel est le cas, la fonction assigne un nouveau numéro automatiquement et reconstruit les matrices.

3.5.9 La classe *cPVsource*

La classe *cPVsource* permet de créer un objet représentant une source PV sur un réseau. Ses variables privées *mP*, *mVabs* et *mRow* stockent respectivement la puissance active (en nombre complexe) et la tension contrôlée de la source PV et son numéro. La classe hérite également des variables privées *mON*, *mNode_k*, *mNode_m*, *mMatrix* et *mFunction* de sa classe parent. L'accès aux variables publiques permet d'afficher ou modifier les caractéristiques de la source

<i>P</i>	permet d'afficher ou modifier la puissance active de la source
<i>Vabs</i>	permet d'afficher ou modifier la tension contrôlée de la source
<i>I</i>	retourne le courant de la source en appelant <i>fCurrent(this)</i>
<i>S</i>	retourne le résultat de l'équation $V \times I^*$
<i>Sk</i>	retourne le résultat de l'équation $V_k \times I^*$
<i>Sm</i>	retourne le résultat de l'équation $V_m \times I^*$
<i>n</i>	retourne le numéro <i>mRow</i> de la source PV

Les fonctions *built-in* de Matlab redéfinies

Bien que la classe *cPVsource* hérite des fonctions built-in de Matlab redéfinies dans la classe *cEquipment*, certaines sont adaptées pour les besoins de la classe *cPVsource*. Entre autre, les fonctions *fieldnames*, *get* et *set* sont modifiées pour inclure les nouveaux champs publics *P*, *Vabs* et *n* de la classe *cPVsource* aux autres paramètres hérités.

Le constructeur

Le constructeur *cPVsource* permet de créer un objet de la classe *cPVsource*. En appelant le constructeur sans passer d'argument, l'usager crée une source PV avec les paramètres par défaut de la classe. Il peut également définir la source en passant les noms des nœuds *k* et *m*, et les paramètres *P*, *Vabs* en argument au constructeur. Dans l'exemple de la Figure 3.31, la source PV est connectée au nœud *montreal*, $P = 5 \times 10^6$ et $Vabs = 25 \times 10^3$

```

>> PV = cPVsource('montreal','GND',5e6,25e3)

PV =
    ON: 1
    k: 'montreal'
    m: 'GND'
    Vk: 0
    Vm: 0
    V: 0
    I: 0
    S: 0
    Sk: 0
    Sm: 0
    function: ''
    P: 5000000
    Vabs: 25000
    n: 1

```

Figure 3.31 : constructeur de la classe *cPVsource*

Les fonctions de l'outil APREM

La fonction *fMatrix* de la classe *cPVsource* construit les matrices **C_{GPV}**, **D_{GPV}**, **AIG** et **f_G** suivant les coordonnées numériques des nœuds *k* et *m*. Dans l'exemple de la Figure 3.32, un petit réseau est monté afin de calculer le courant de la source PV par écoulement de puissance, sans quoi, les matrices **C_{GPV}** et **D_{GPV}** ne peuvent être évaluées. Les matrices sont ensuite stockées dans la structure *MatPV*. Les quatre (4) matrices **C_{GPV}**, **D_{GPV}**, **AIG** et **f_G** sont requises pour représenter la source PV dans le réseau. La théorie présentée à la section 3.2.2 est utilisée ici pour construire les matrices :

$$\begin{bmatrix} Y_k & A_{IL} \\ C_{LPQ} & D_{GPV} \end{bmatrix} \times \begin{bmatrix} \Delta V_k \\ \Delta I_G \end{bmatrix} = \begin{bmatrix} \Delta f_n \\ \Delta f_{PV} \end{bmatrix} \quad (3.203)$$

```

>> PV = cPVsource('montreal','GND',5e6,25e3);
>> Z = cImpedance('montreal','st_bruno',0.1,0.2);
>> chargeZ = cImpedance('st_bruno','GND',10,8);
>> reseau.PV = PV; reseau.Z = Z; reseau.chargeZ = chargeZ;
>> reseau = fLoadFlow(reseau);

```

Figure 3.32 : réseau test pour initialiser l'objet de la classe *cPVsource*

Le code de la Figure 3.32 crée la structure *reseau* composée d'une source *PV*, d'une impédance *Z* et de la charge *chargeZ* représentée par une impédance. La fonction *fLoadFlow* réalise ensuite le calcul d'écoulement de puissance du réseau et trouve les tensions des noeuds ainsi que le courant circulant dans la source *PV*. La fonction *fMatrix* retourne ainsi les matrices de la source *PV* calculées à la dernière itération (Figure 3.33).

Toujours selon la même nomenclature, les matrices **C_{GPV}**, **D_{GPV}**, **AIG** et **f_{GPV}** sont stockées sous forme de *cells* dans l'objet :

$$\mathbf{C}_{\mathbf{GPV}} = \left\{ \text{mRow} \quad k \quad \begin{bmatrix} -I_{GR} & -I_{GI} \\ \frac{V_{kR}}{|V_k|} & \frac{V_{kI}}{|V_k|} \end{bmatrix} \right\} = \left\{ 1 \quad 2 \quad \begin{bmatrix} 1491.9 & 95.2130 \\ 1 & 0 \end{bmatrix} \right\} \quad (3.204)$$

$$\mathbf{D}_{\mathbf{GPV}} = \left\{ \text{mRow} \quad \text{mRow} \quad \begin{bmatrix} -V_{kR} & -V_{kI} \\ 0 & 0 \end{bmatrix} \right\} = \left\{ 1 \quad 1 \quad \begin{bmatrix} -25e3 & 0 \\ 0 & 0 \end{bmatrix} \right\} \quad (3.205)$$

```

>> [PV MatPV] = fMatrix(PV, 'update');
>> MatPV

MatPV =

    AIG: {[1]  [1]  [2x2 double]}
    CGPV: {[1]  [1]  [2x2 double]}
    DGPV: {[1]  [1]  [2x2 double]}
    FGPV: {[1]  [1]  [2x1 double]}

>> MatPV.CGPV{1,3}

ans =

1.0e+003 *

    1.4919   -1.2112
    0.0010       0

>> PV.I

ans =

-1.4919e+003 +1.2112e+003i

>> PV.V

ans =

    25000

```

Figure 3.33 : fonction *fMatrix* de la classe *cPVsource*

$$\mathbf{AIG} = \{mRow \ k \ [+u]\} = \left\{ 1 \ 1 \ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right\} \quad (3.206)$$

$$\mathbf{F}_{\text{GPV}} = \left\{ k \ 1 \ \begin{bmatrix} P \\ V_{abs} \end{bmatrix} \right\} = \left\{ 1 \ 1 \ \begin{bmatrix} 5e6 \\ 25e3 \end{bmatrix} \right\} \quad (3.207)$$

fValidateRow est appelée avant de bâtir la matrice du réseau afin de vérifier s'il n'existe pas de source PV ayant le même numéro mRow. Si tel est le cas, la fonction assigne un nouveau numéro et reconstruit les matrices.

3.6 Comment fonctionne l'outil APREM

3.6.1 Cration du rseau

Le logiciel APREM est entierement dvelopp dans le langage de programmation Matlab. Tous les fichiers .m qui compose le logiciel sont donc libres d'accès et modifiables par l'utilisateur qui souhaite personnaliser l'application. Le dveloppement des cas d'udes de rseau  travers l'outil APREM suit donc les pratiques et methodologies de dveloppement de Matlab, ce qui limite l'effort d'apprentissage puisque la syntaxe reste la mme. Ce que l'outil APREM fournit de plus  l'utilisateur sont toutes les classes et fonctions ncessaires pour modliser, construire et analyser les rseaux lectriques.

L'utilisation de l'outil APREM se limite donc essentiellement  deux choses, soit dcrire dans un fichier .m un rseau et paramtrer les analyses  excuter.

```
%% Build the Network Structure
addpath lib
initialisation;

Zbase = 100e6/500e3;
Network.L1 = cImpedance('B1','B2', 0.1*Zbase,0.5*Zbase);
Network.L2 = cImpedance('B2','B3', 0.1*Zbase,0.4*Zbase);
Network.L3 = cImpedance('B3','B1', 0.1*Zbase,0.3*Zbase);

Network.PQ1 = cLoad('B1','O',500e6,0,500e3);
Network.PQ2 = cLoad('B2','O',300e6,0,500e3);
Network.PQ3 = cLoad('B3','O',1000e6,300e6,500e3);

Network.V1 = cPVsource('B1','O',700e6,500e3);
Network.V2 = cVsource('B2','O',500e3,0);
```

Figure 3.34 : cration du rseau Network dans Matlab : *NetworkStruct.m*

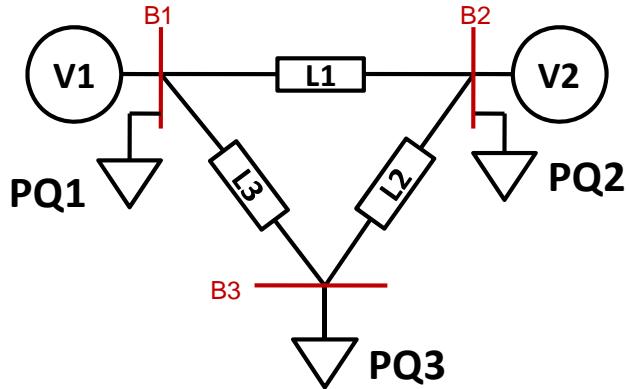


Figure 3.35 : réseau Network crée à partir du fichier *NetworkStruct.m*

Tout débute donc par la création d'un réseau à l'aide des objets disponibles dans l'APREM. Le code Matlab à la Figure 3.34, rédigé dans le fichier *NetworkStruct.m*, génère le réseau de trois nœuds de la Figure 3.35 dans la structure *Network*. Les nœuds B1, B2 et B3 sont reliées entre eux par les lignes L1, L2 et L3. La source PV (V1) est connectée à la barre B1 et la source de tension (V2) à la barre B2. Les charges PQ1, PQ2 et PQ3 sont raccordées aux barres B1, B2 et B3 respectivement. Les commandes *addpath* et *initialisation* initialisent l'outil APREM pour une première utilisation. L'exécution du fichier permet d'appeler la commande *Network* afin d'afficher le contenu de la structure.

Il est important de noter que l'outil APREM ne se limite pas à l'étude d'un seul réseau. Il est possible de créer plusieurs réseaux indépendants en associant les équipements qui les composent à différentes structures telles que *montreal*, *reseau_de_Sherbrooke* ou autre. Le nom des structures n'a aucune importance et le traitement des réseaux peut alors se faire simultanément ou séquentiellement au besoin.

```
>> Network

Network =

L1: [1x1 cImpedance]
L2: [1x1 cImpedance]
L3: [1x1 cImpedance]
PQ1: [1x1 cLoad]
PQ2: [1x1 cLoad]
PQ3: [1x1 cLoad]
V1: [1x1 cPVsource]
V2: [1x1 cVsource]
```

Figure 3.36 : structure Network créé à partir du fichier *NetworkStruct.m*

3.6.2 Écoulement de puissance du réseau

L’analyse de l’écoulement de puissance est le calcul au cœur de l’outil APREM autour duquel l’usager personnalise ses études. Lancer le calcul d’écoulement de puissance dans l’environnement APREM est une étape relativement simple dans processus qui entoure l’utilisation du logiciel.

Lorsque le réseau à l’étude est généré, l’usager appelle la fonction *fLoadFlow* en passant la structure *Network* en argument. La fonction *fLoadFlow* réalise alors le calcul d’écoulement de puissance et associe toutes les tensions et courants aux objets et nœuds qui composent le réseau.

L’usager peut ensuite, s’il le désire, appeler la fonction *fSave* afin de sauvegarder la solution complète du réseau. Cette dernière convertit chaque objet APREM en structure Matlab pour plus de portabilité. Les objets APREM pointent uniquement sur les vecteurs résultats et n’enregistre donc pas les valeurs de tension, courant et puissance à l’intérieur des objets afin de limiter les opérations *get/set* lors des boucles de calculs.

Ainsi, une fois l’opération d’analyse terminée, il est possible de lancer la fonction *fReport* pour imprimer les résultats de l’écoulement de puissance dans un format prédéterminé dans la fenêtre de commandes de Matlab (voir Figure 3.37). Bien entendu, l’outil APREM offre d’autres possibilités d’affichage, dont l’exportation des données dans un fichier Excel, en plus des fonctionnalités déjà disponibles dans Matlab.

```

>> Network = fLoadFlow(Network);
>> fReport(fSave(Network));

14-Aug-2009 05:26:16      Case_1

=====
Bus Data
=====
      Voltage :
Bus ID    Mag(V)      Ang(deg)
-----
      GND      0.000      0.000
      B1     500000.000   -5.973
      B2     500000.000      0.000
      B3    449917.907   -11.369

=====
Branch Data
=====
      Branch      From      To      Voltage:      Current:      Power:
      ID  ON/OFF  Bus k  Bus m    Mag(V)    Ang(deg)    Mag(A)    Ang(deg)    P(W)      Q(VAr)
-----
      L1        1      B1      B2    5.210e+004   -92.986   5.109e+002   -171.676   5.22e+006   2.61e+007
      L2        1      B2      B3    1.065e+005    56.407   1.291e+003   -19.557   3.33e+007   1.33e+008
      L3        1      B3      B1    6.710e+004   -146.880   1.061e+003   141.555   2.25e+007   6.75e+007
      PQ1       1      B1      GND   5.000e+005   -5.973   1.000e+003   -5.973   5.00e+008   0.00e+000
      PQ2       1      B2      GND   5.000e+005      0.000   6.000e+002      0.000   3.00e+008   0.00e+000
      PQ3       1      B3      GND   4.499e+005   -11.369   2.320e+003   -28.068   1.00e+009   3.00e+008
      V1        1      B1      GND   5.000e+005   -5.973   1.563e+003   147.602   -7.00e+008  -3.48e+008
      V2        1      B2      GND   5.000e+005      0.000   2.350e+003   171.230   -1.16e+009 -1.79e+008

```

Figure 3.37 : Impression des résultats de l'écoulement de puissance de Network avec *fReport*

Bien que le processus soit entièrement transparent pour l'usager, la fonction *fLoadFlow* réalise plusieurs opérations préalables aux calculs de répartition de puissance. Ces fonctions développées pour l'outil APREM valident la composition du réseau, vérifient la connectivité des nœuds, identifient les îlots électriques et les sources isolées et corrigent au besoin le réseau afin d'assurer la convergence des résultats, peu importe la justesse du réseau. La fonction *fLoadFlow* initialise également le système pour la première itération de Newton-Raphson en remplaçant les charges et les sources PV par leur impédance et source de tension équivalentes. Séquentiellement, la fonction *fLoadFlow* s'organise pour l'essentiel comme suit (Figure 3.38) :

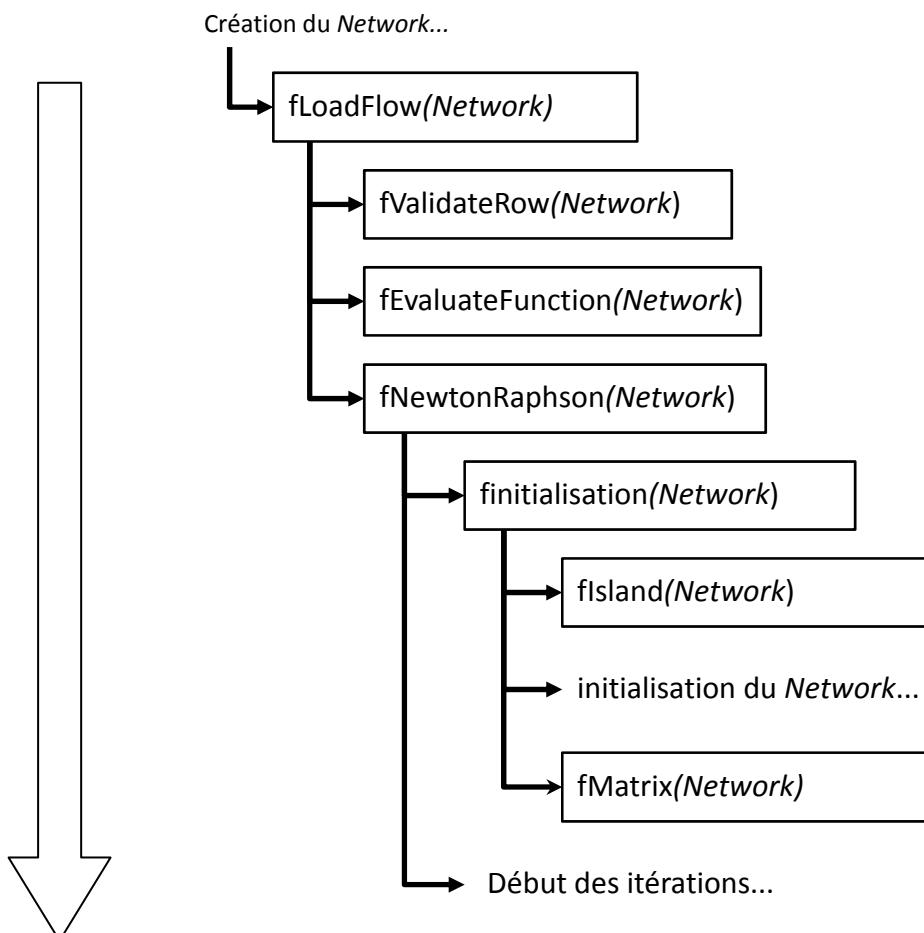


Figure 3.38 : structure de la fonction *fLoadFlow*

fValidateRow s'assure que les objets d'une même classe portent tous un numéro d'identification (*mRow*) différent

fEvaluateFunction exécute les fonctions spéciales définies par l'usager vers lesquelles pointe le paramètre *function* de chaque objet dans le réseau.

- fInitialisation*
- retire du système les îlots électriques qui ne sont alimentés par aucune source et transforme les sources PV isolées en sources de tension idéale.
 - initialise le système en remplaçant les charges PQ et les sources PV par leurs impédances et sources de tension équivalentes,
 - construit la matrice nodale augmentée du réseau,
 - solutionne la matrice directe du réseau et
 - assigne aux charges PQ et sources PV, les injections de courant calculées pour leurs équivalents

fIsland construit la liste d'adjacence [38] des noeuds du réseau et applique la théorie des graphes pour détecter les îlots électriques et les sources isolées.

fMatrix construit la matrice nodale augmentée du réseau

- fNewtonRaphson*
- exécute les itérations de Newton Raphson pour solutionner l'écoulement de puissance en ajoutant les contraintes de puissance des charges PQ et des sources PV à la matrice du réseau pour former la matrice jacobienne et
 - assigne les tensions et courants aux nœuds et équipements du réseau.

3.6.3 La fonction *fIsland*

La fonction *fIsland* est une pièce importante dans la réalisation des études paramétriques des réseaux électriques. L'outil APREM a la vocation de réaliser plusieurs analyses de réseaux suivant des scénarios prédéfinis dans certains cas, et complètement aléatoire dans d'autres. Tous ces scénarios simulés en boucle exposent le logiciel à toutes sortes de configurations de réseaux

qui sont pratiquement impossibles à prédire, surtout dans un contexte probabiliste. Le défi est immense puisque rien ne garantit la validité des réseaux tels qu'ils se présenteront et donc la convergence des écoulements de puissance peut être compromise si la topologie du réseau est mathématiquement dégénérée.

La fonction *fIsland* est donc au cœur du mécanisme de validation des topologies des réseaux à l'étude. Elle est appelée lors de l'initialisation des réseaux avant chaque écoulement de puissance. Elle s'appuie sur la théorie des Graphes utilisé largement dans plusieurs autres domaines [38]. La fonction construit la liste d'adjacence du réseau qui établit la connectivité entre les nœuds des réseaux. Pour le réseau traité plus haut, la liste d'adjacence a la forme suivante (Figure 3.39) :

```
K>> graph
graph =
B1: ('B2'  'B3')
B2: ('B1'  'B3')
B3: ('B1'  'B2')
```

Figure 3.39 : liste d'adjacence du réseau Network

Cette liste permet d'utiliser une méthode appelée *Depth-First Traversal* pour parcourir tous les nœuds du réseau. Cette technique utilise un algorithme récursif (*fTraversal* – Figure 3.40) qui permet d'identifier le ou les nœuds isolés. L'information est par la suite traitée afin de déterminer si le ou les nœuds isolés sont connectés à une source, et si cette source est une source PV. Lors de l'initialisation, l'outil APREM retirera alors tous les nœuds flottants du réseau et convertira les sources PV isolées en source de tension équivalente pour assurer la convergence des itérations.

```

function nodeVisited = fTraversal(graph,k,varargin)
persistent mark
if isempty(mark), mark = (); end
if nargin>2
    mark = varargin(1);
end
mark = [mark, {k}];
if isfield(graph,k)
    m = graph.(k);
    for i = 1:length(m)
        if iscell(m)
            [tmp,n] = intersect(mark,m(i));
            if isempty(n)
                fTraversal(graph,m(i));
            end
        end
    end
end
nodeVisited = mark;

```

Figure 3.40 : fonction *fTraversal*

Ce processus est essentiel au fonctionnement de l'outil APREM puisqu'il permet à l'utilisateur de s'affranchir de l'analyse préliminaire de la topologie du réseau au passage de chaque scénario. La technique permet de résoudre les problèmes liés à la topologie des réseaux, mais ne traite pas les non-convergences dues aux caractéristiques électriques des équipements telles que la capacité de transfert des lignes électriques. L'outil se contentera de limiter le nombre d'itérations lorsque les résultats ne convergent pas. Il est donc primordial que l'usager valide préalablement les paramètres des équipements qu'il intègre dans son réseau afin de refléter des conditions d'exploitations normales du réseau.

Bien que la fonction ne soit pas optimale, elle couvre les besoins essentiels de l'outil de recherche et apporte automatiquement les corrections nécessaires sur les réseaux pour assurer la convergence des systèmes.

3.6.4 Analyse paramétrique

L'outil APREM est développé essentiellement pour permettre à l'usager de modifier facilement les paramètres et la topologie de son réseau à travers un minimum d'opération et de lignes de code. Comme expliqué longuement dans les chapitres précédents, le choix d'introduire des objets Matlab dans l'étude des réseaux électriques est encouragé par la flexibilité qu'offre Matlab de définir les méthodes d'accès aux variables qui caractérisent les objets. Ainsi, le logiciel APREM jouit d'une syntaxe intuitive et comparable à la structure du langage de programmation déjà disponible dans Matlab.

La structure du code utilisée pour paramétriser une étude suivant l'approche APREM s'organise essentiellement autour de 4 blocs :

- la définition du réseau;
- l'initialisation des paramètres des simulations;
- les boucles de simulations;
- l'impression des résultats.

Dans le premier bloc, l'usager crée son réseau tel que suggéré dans la section précédente. Dans le second bloc, l'usager initialise les paramètres d'étude du réseau, le nombre de simulations, les caractéristiques initiales des équipements, les plages de valeurs (puissance, tension, état, etc.) qui seront appliquées aux équipements tout au long des simulations ou toute autre information susceptible de personnaliser l'étude. Les modifications sur les paramètres du réseau et/ou de l'étude sont ensuite appliquées l'une après l'autre à l'intérieur des boucles de simulations afin de simuler tous les scénarios. Chaque modification est suivie des calculs d'écoulement de puissance pour en apprécier l'impact et les résultats peuvent être mémorisés à cette étape. L'opération suivante est naturellement le traitement et l'affichage des résultats pour lesquels Matlab offre plusieurs outils. Bien entendu, rien n'empêche le chevauchement de tous ces blocs dans la réalisation des études.

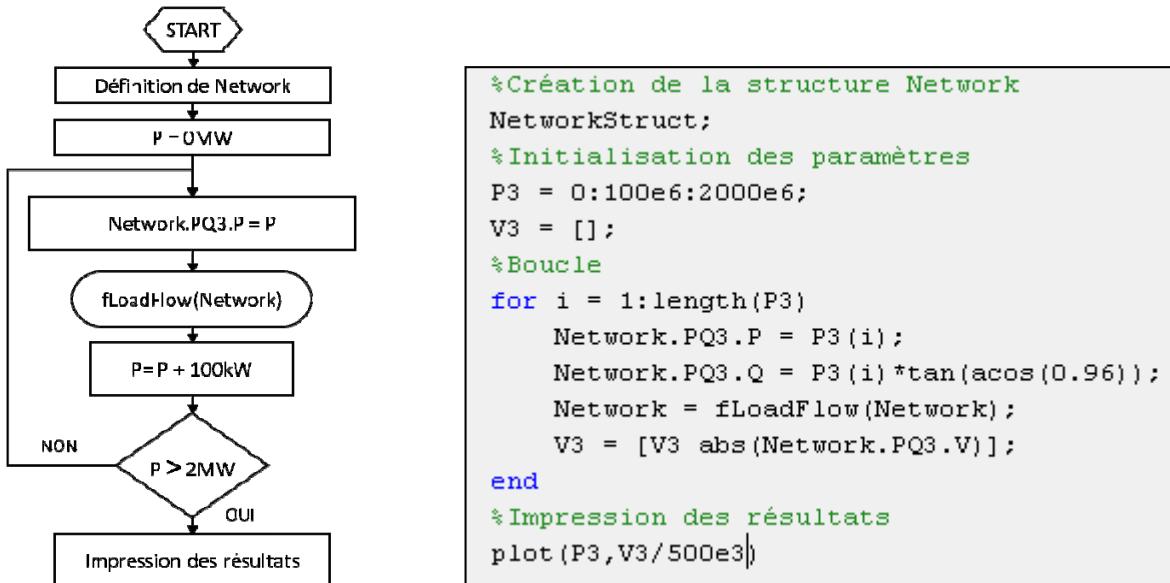


Figure 3.41 : Analyse paramétrique avec l’outil APREM

L’exemple précédent de la Figure 3.35 est repris ci-dessus à la Figure 3.41 afin de présenter cette approche. **NetworkStruct** appelle le fichier *NetworkStruct.m* dans lequel est construit le réseau à trois barres. Pour l’étude, la puissance de la charge PQ3 varie de 0 à 2MW et la tension aux bornes de la charge PQ3 est enregistrée après chaque simulation. L’objectif est d’observer la variation de la tension à la barre B3 en fonction de la charge PQ3. L’opération est simple, mais, à titre indicatif, ces quelques lignes de code permettent ainsi de réaliser 200 écoulements de puissance et de renseigner l’usager sur la charge admissible à la barre B3 (Figure 3.42).

L’intérêt d’une telle approche est qu’encore aujourd’hui, la grande proportion des logiciels d’analyse de réseaux ne permettent pas facilement de paramétriser l’étude des réseaux. Si l’exemple précédent est repris sur un logiciel tel que PSAF de CYME International, l’usager doit manuellement modifier et simuler 200 fois son réseau afin de tirer les mêmes résultats, sans compter l’effort requis pour extraire et compiler les données afin de tracer la même courbe sur un autre logiciel (Excel).

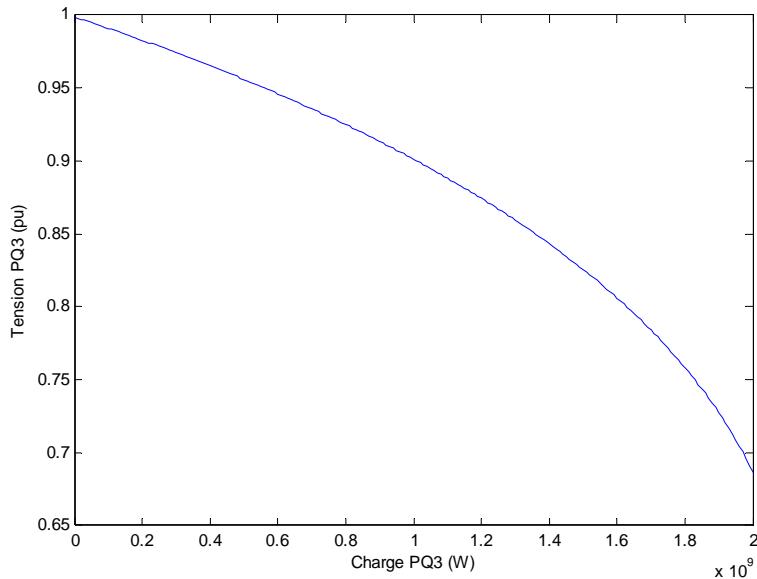


Figure 3.42 : tension à la barre B3 de Network en fonction de la charge PQ3

Les logiciels d’analyse de réseaux optent majoritairement pour une démarche séquentielle dans la définition et le traitement des études ou encore utilisent des listes de contingence appliquées sur un scénario de base. Pour chaque scénario simulé, l’usager doit donc prédefinir les changements à son réseau et lancer une simulation. L’outil APREM préconise une approche programmée dans laquelle les changements opérés sur le réseau pour chaque scénario sont exécutés automatiquement simulation après simulation, sans intervention de l’usager. Toutes les opérations sont préalablement définies et font partie intégrante du projet d’étude. Ces changements peuvent également être définis en fonction de règles conditionnelles afin d’introduire des relations de cause à effet sur réseau.

3.6.5 Le paramètre *function* et la fiabilité

L’objectif de l’outil APREM est sans contredit de pouvoir paramétrier les études des réseaux électriques. Mais, l’intérêt principal de cette approche est l’analyse de la fiabilité des réseaux électriques.

La programmation des études sur Matlab permet d’introduire des règles d’exploitations conditionnelles à l’état et/ou la configuration du réseau, mais aussi des notions de probabilités afin de reproduire des évènements probabilistes simulation après simulation, tel que le bris d’un

équipement sur le réseau, la perte de capacité suite à une panne, un entretien, etc., sans l'intervention de l'utilisateur. Le but de l'approche demeure de pouvoir réaliser des études de type Monte-Carlo et l'outil APREM reste extrêmement flexible sur la structure à adopter pour incorporer ces évènements probabilistes. Il est possible de programmer, à même le code principal et les boucles de simulations, ces règles d'opération conditionnelles ou aléatoires indépendamment des équipements et de définir les relations de cause à effet, comme cela serait requis pour le changeur de prise d'un transformateur par exemple.

Néanmoins, dans le but d'alléger le contenu des boucles de simulations, l'outil met à disposition de tous les équipements le paramètre **function** auquel l'usager peut assigner à un ou plusieurs équipements, une ou des fonctions spéciales propres qu'il aura définies dans un fichier .m portant le même nom (ex : *changeurDePrise.m*). À travers ce paramètre **function**, l'usager est en mesure d'introduire des fonctions de probabilité ou des fonctions personnalisées qui agissent sur les caractéristiques des équipements en fonction d'une contrainte, d'une condition ou encore d'une probabilité. Ces fonctions spéciales peuvent être appelées à chaque simulation ou chaque itération de la méthode de Newton-Raphson.

Les possibilités sont donc illimitées et peuvent s'avérer d'une très grande utilité pour les études de type Monte-Carlo, puisqu'il est possible d'associer, par exemple, une probabilité de panne à chacun des équipements d'un réseau, simuler le réseau en boucle et laisser ces fonctions probabilistes dicter l'état des équipements. En suivant la théorie applicable aux études Monte-Carlo, la compilation des résultats d'un grand nombre de simulations renseigneraient alors sur les zones où l'inclusion des probabilités a le plus d'impact.

Le code à la Figure 3.43 propose une étude de fiabilité appliquée toujours au réseau de la Figure 3.35. Les impédances des lignes L1, L2 et L3 ont volontairement été diminuées pour respecter la capacité de transfert des lignes et assurer la convergence des résultats.

```

%% Build the Network Structure (nnetwork4.m)
- Network = [];
- Zbase = 100e6/500e3;
- Network.L1 = cImpedance('B1','B2', 0.01*Zbase,0.05*Zbase);
- Network.L2 = cImpedance('B2','B3', 0.01*Zbase,0.04*Zbase);
- Network.L3 = cImpedance('B3','B1', 0.01*Zbase,0.03*Zbase);

- Network.PQ1 = cLoad('B1','O',500e6,0,500e3);
- Network.PQ2 = cLoad('B2','O',300e6,0,500e3);
- Network.PQ3 = cLoad('B3','O',1000e6,300e6,500e3);

- Network.V1 = cPVsource('B1','O',700e6,500e3);
- Network.V2 = cVsource('B2','O',500e3,0);

- index = fieldnames(Network);
- for n = 1:length(index)
    Network.(index(n)).function = 'fProbabilityTest';

- end

```

Figure 3.43 : utilisation du paramètre *function* pour l'étude de fiabilité

```

function varargout = fProbabilityTest(this,varargin)
%%%%%%%%%%%%%
% Tirage au sort - probabilité 1/6
%%%%%%%%%%%%%
- switch nargin
- case(1),firstIteration = false; Network = [];
- case(2),firstIteration = varargin(1);Network = [];
- case(3),firstIteration = varargin(1);Network = varargin(2);
- end

- if firstIteration
    ON = rand; %chiffre random de 0 à 1
    if ON < 1/6
        this.ON = false;
    else
        this.ON = true;
    end
- end
doLoadFlowAgain = false;

- varargout(1) = this;
- varargout(2) = doLoadFlowAgain;

```

Figure 3.44 : fonction *fProbabilityTest* pour simuler les pannes d'équipement

La fonction *fProbabilityTest* introduite au paramètre *function* de tous les équipements simule la probabilité d'indisponibilité de l'équipement. La fonction *rand* réalise un tirage au sort de 0 à 1, et si le résultat est inférieur à 1/6, l'équipement est déconnecté du réseau.

Le code principal simule alors 300 fois le réseau, et à chaque passage dans la fonction *fLoadFlow*, l'outil évalue la fonction *fProbabilityTest* sur chacun des équipements et retire ou ajoute, selon le cas, l'équipement en traitement dans le réseau. Les résultats sont ensuite exportés après les 300 simulations dans un fichier Excel (*output.xls*).

```
%Création de la structure Network
- nnetwork4;
%Monte Carlo
- for i = 1:300
-     Network = fLoadFlow(Network);
-     fSave(Network);
-     i
- end
%Impression des résultats dans Excel
- SavedData = fSave;
- fReportxls(SavedData,'output.xls');
```

Figure 3.45 : Simulation Monte-Carlo avec l'outil APREM

Évidemment, l'étude de Monte-Carlo requiert beaucoup plus que 300 simulations pour que les résultats convergent vers des valeurs tangibles qui reflètent l'impact de ces probabilités. Plus d'un millier de simulations serait alors requis et il suffirait simplement d'augmenter le nombre de simulations à 1000, voir 2000, ou encore développer une fonction qui évaluerait la convergence des résultats (voir figure 1.3).

La Figure 3.46 montre le contenu du fichier Excel créé à l'aide de la fonction *fReportxls*. Tous les résultats des écoulements de puissance des 300 simulations Monte-Carlo sont alors disponibles dans ce fichier. Il est ainsi possible d'utiliser des fonctionnalités du logiciel pour l'analyse des résultats.

	A	B	C	D	E	F	G	H	I	J	K
1	CASE N	ID	ON/OFF	FROM	TO	VOLTA	VOLTA	CURRE	CURRE	REAL P	IMAG P
7	Case_1	PQ3	TRUE	B3	GND	275511	-0,38886	2187,83	-17,0881	1E+09	3E+08
15	Case_2	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
23	Case_3	PQ3	TRUE	B3	GND	275511	-0,38886	2187,83	-17,0881	1E+09	3E+08
31	Case_4	PQ3	TRUE	B3	GND	488282,4	-2,67735	1234,473	-19,3766	1E+09	3E+08
39	Case_5	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
47	Case_6	PQ3	TRUE	B3	GND	492158,5	-3,19257	1224,75	-19,8918	1E+09	3E+08
55	Case_7	PQ3	TRUE	B3	GND	493208	-1,53234	1222,144	-18,2316	1E+09	3E+08
63	Case_8	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
71	Case_9	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
79	Case_10	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
87	Case_11	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
95	Case_12	PQ3	TRUE	B3	GND	490807,8	-1,72798	1228,121	-18,4272	1E+09	3E+08
103	Case_13	PQ3	TRUE	B3	GND	490807,8	-1,72798	1228,121	-18,4272	1E+09	3E+08
111	Case_14	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
119	Case_15	PQ3	FALSE	B3	GND	0	0	0	0	0	0
127	Case_16	PQ3	TRUE	B3	GND	495797,5	-0,62264	1215,761	-17,3219	1E+09	3E+08
135	Case_17	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
143	Case_18	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
151	Case_19	PQ3	TRUE	B3	GND	275511	-0,38886	2187,83	-17,0881	1E+09	3E+08
159	Case_20	PQ3	FALSE	B3	GND	0	0	0	0	0	0
167	Case_21	PQ3	FALSE	B3	GND	0	0	0	0	0	0
175	Case_22	PQ3	TRUE	B3	GND	490807,8	-1,72798	1228,121	-18,4272	1E+09	3E+08
183	Case_23	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
191	Case_24	PQ3	TRUE	B3	GND	495910,4	-1,0231	1215,484	-17,7223	1E+09	3E+08
199	Case_25	PQ3	TRUE	R3	GND	492158,5	-1,25741	1224,75	-17,9567	1F+09	3F+08

Figure 3.46 : Exportation des résultats vers Excel : fReportxls

3.7 Validation et résultats

3.7.1 Exemple I : projet de ligne en Bolivie

L'exemple de validation présenté ci-après fait l'analyse d'un projet de ligne de transport de 200 kV au nord de la Bolivie entre l'aménagement hydroélectrique projeté de Cachuela et une petite commune au nord-ouest du pays, Cobija. Le tracé de la ligne projeté d'environ 500 km traverse plusieurs petites localités, soit Puerto Rico, Riberalta et Cachuela. Une extension vers l'est du Barrage est également prévue pour alimenter la ville de Guayaramin. L'ambigüité du projet réside dans la capacité des charges qui sont relativement faibles et leur croissance projetée de 2014 jusqu'à 2034 ne varie pas au même rythme selon la localité.

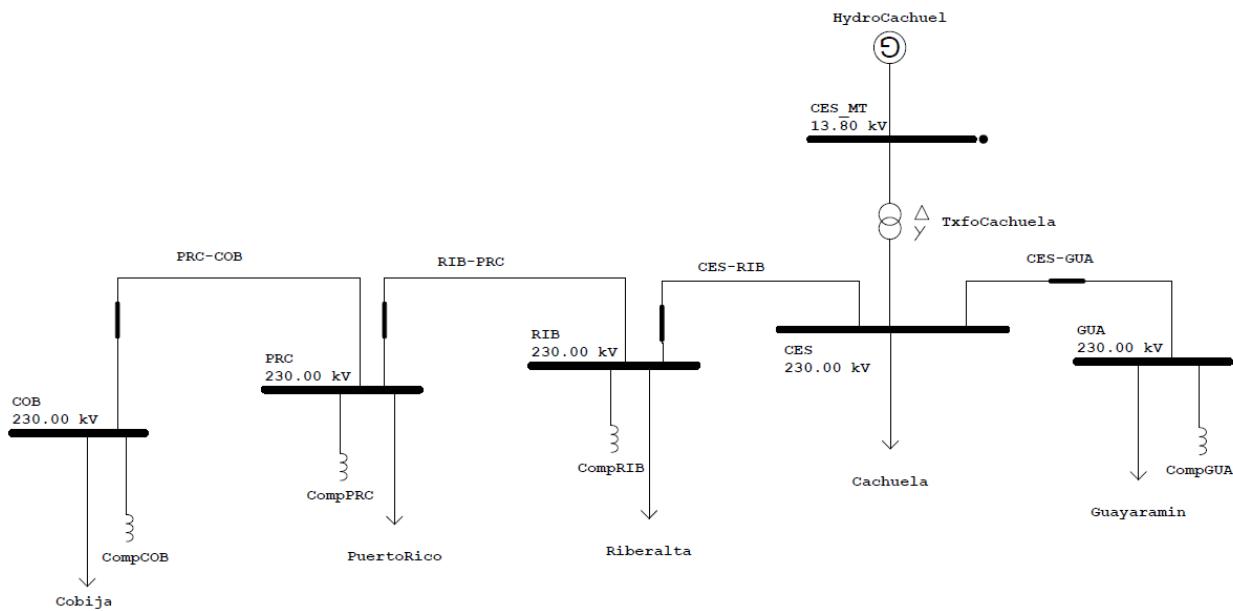


Figure 3.47 : Exemple I : projet de ligne en Bolivie

L'étude réalise une simulation d'évolution de ce réseau suivant des valeurs de charge qui croissent année après année pour chacune des localités. Les données initiales du projet sont dressées dans le tableau 3.1 qui suit.

Tableau 3.1 : Données de base sur l'évolution des charges

Année	Cobija (W)	PuertoRico (W)	Riberalta (W)	Cachuela (W)	Guyaramin (W)	Année	Cobija (W)	PuertoRico (W)	Riberalta (W)	Cachuela (W)	Guyaramin (W)
2015	13 637	741	7 315	74	4 390	2031	29 626	1 826	16 645	165	7 430
2016	20 456	1 111	7 782	78	4 644	2032	30 367	1 872	17 061	174	7 539
2017	20 967	1 176	8 279	83	4 912	2033	31 126	1 919	17 488	182	7 728
2018	21 492	1 244	8 808	88	5 059	2034	31 904	1 967	17 925	191	7 921
2019	22 029	1 281	9 371	92	5 211	2035	32 702	2 016	18 373	201	8 119
2020	22 580	1 319	9 969	97	5 368	2036	33 520	2 066	18 832	211	8 322
2021	23 144	1 359	10 468	102	5 529	2037	34 358	2 118	19 303	222	8 530
2022	23 723	1 400	10 991	107	5 694	2038	35 217	2 171	19 786	233	8 743
2023	24 316	1 442	11 541	112	5 865	2039	36 097	2 225	20 280	244	8 962
2024	24 924	1 485	12 118	118	6 041	2040	36 999	2 281	20 787	257	9 186
2025	25 547	1 530	12 724	123	6 222	2041	37 924	2 338	21 307	269	9 416
2026	26 185	1 575	13 360	130	6 409	2042	38 872	2 396	21 840	283	9 651
2027	26 840	1 623	14 028	136	6 601	2043	39 844	2 456	22 386	297	9 892
2028	27 511	1 671	14 729	143	6 799	2044	40 840	2 518	22 945	312	10 140
2029	28 199	1 721	15 466	150	7 003	2045	41 861	2 581	23 519	327	10 393
2030	28 904	1 773	16 239	157	7 213						

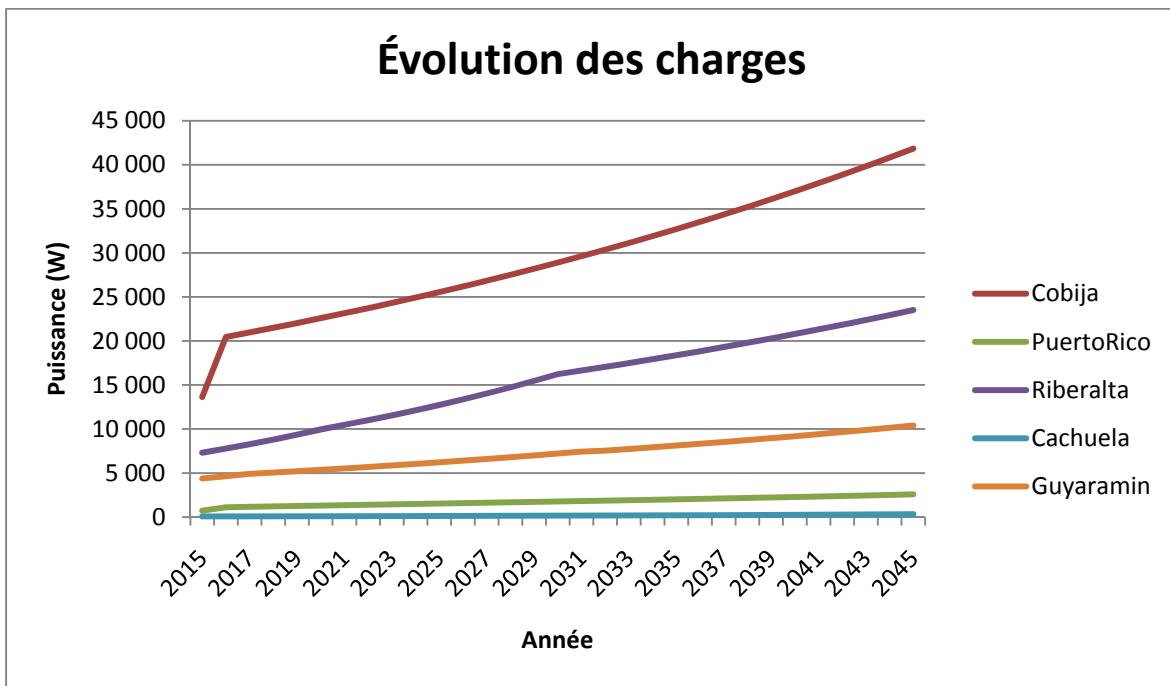


Figure 3.48 : Évolution des charges

Création du réseau

Les lignes de code tirées du fichier *networkDefinition.m* (Figure 3.49) permettent de construire le réseau dans son ensemble. La première section du code (ligne 2 à 26) crée :

- la centrale hydroélectrique de Cachuela, une source de tension de 13.8kV;
- le transformateur élévateur de 65MVA, 13.8/230kV avec une impédance de 11.79% et un ratio X/R de 33.344;
- et la ligne 230 kV de 500 km reliant les cinq localités à l'étude.

```

1 %% SOURCE
2 - Network = struct;
3 - Network.HydroCachuela = cVsource('CES_MT','0',13.8e3,0);
4
5 %% TRANSFORMER
6 - Z1 = 0.1179;
7 - r = 33.344;
8 - R1 = sqrt(Z1^2/(1+r^2));
9 - X1 = r*R1;
10 - Vrated2 = 13.8e3;           % V
11 - Sbase2 = 65e6;             % VA
12 - Zbase2 = Vrated2^2/Sbase2; % Ohm
13 - Z = R1+1i*X1;
14 - Z = Z*Zbase2;
15 - Network.TxfoCachuela = cTxfoZ('CES_MT',0,'CES',0,13.8/230,0,Z);
16
17 %% LINES
18 - R1 = 0.074492;
19 - X1 = 0.410952;
20 - B1 = 2.8000e-6;
21 - Z = R1+1i*X1;
22 - Y = 1i*B1;
23 - l = 40.83; Network.Z_CES_GUA = cLine('CES','GUA',Z*l,Y*l);
24 - l = 83.17; Network.Z_CES_RIB = cLine('CES','RIB',Z*l,Y*l);
25 - l = 235.38; Network.Z_RIB_PRC = cLine('RIB','PRC',Z*l,Y*l);
26 - l = 157.03; Network.Z_PRC_COB = cLine('PRC','COB',Z*l,Y*l);

```

Figure 3.49 : fichier *NetworkDefinition.m* (1/2)

Les valeurs de puissance à utiliser année après année pour chacune des charges sont ensuite stockées dans des tableaux. Ces données en MW sont converties en MVA en appliquant un facteur de puissance de 0.85 (ligne 54 à 56). Les charges sont alors initialisées avec le premier élément de chaque tableau, soit l'année 2015.

```

28 %% LOADS
29 - Cobija = [13.637 20.456 20.967 21.492 22.029...
30   22.580 23.144 23.723 24.316 24.924 25.547...
31   26.185 26.840 27.511 28.199 28.904 29.626...
32   30.367 31.126 31.904 32.702 33.520 34.358...
33   35.217 36.097 36.999 37.924 38.872 39.844...
34   40.840 41.861]*1e6;
35
36 - Riberalta = [7.315 7.782 8.279 8.808 9.371 9.969 10.468 10.991 11.541 ...
37   12.118 12.724 13.360 14.028 14.729 15.466 16.239 16.645 17.061 17.488 ...
38   17.925 18.373 18.832 19.303 19.786 20.280 20.787 21.307 21.840 22.386 ...
39   22.945 23.519]*1e6;
40
41 - PuertoRico = [0.741 1.111 1.176 1.244 1.281 1.319 1.359 1.400 1.442 1.485 ...
42   1.530 1.575 1.623 1.671 1.721 1.773 1.826 1.872 1.919 1.967 2.016 2.066 ...
43   2.118 2.171 2.225 2.281 2.338 2.396 2.456 2.518 2.581]*1e6;
44
45 - Cachuela = [0.074 0.078 0.083 0.088 0.092 0.097 0.102 0.107 0.112 0.118 ...
46   0.123 0.130 0.136 0.143 0.150 0.157 0.165 0.174 0.182 0.191 0.201 ...
47   0.211 0.222 0.233 0.244 0.257 0.269 0.283 0.297 0.312 0.327]*1e6;
48
49 - Guayaramerin = [4.390 4.644 4.912 5.059 5.211 5.368 5.529 5.694 5.865 ...
50   6.041 6.222 6.409 6.601 6.799 7.003 7.213 7.430 7.539 7.728 7.921 ...
51   8.119 8.322 8.530 8.743 8.962 9.186 9.416 9.651 9.892 10.140 ...
52   10.393]*1e6;
53
54 - Cobija = Cobija + 1i*((Cobija/0.85).^2-Cobija.^2).^0.5;
55 - Riberalta = Riberalta + 1i*((Riberalta/0.85).^2-Riberalta.^2).^0.5;
56 - PuertoRico = PuertoRico + 1i*((PuertoRico/0.85).^2-PuertoRico.^2).^0.5;
57 - Cachuela = Cachuela + 1i*((Cachuela/0.85).^2-Cachuela.^2).^0.5;
58 - Guayaramerin = Guayaramerin + 1i*((Guayaramerin/0.85).^2-Guayaramerin.^2).^0.5;
59
60 - Cobija = Cobija.';
61 - Riberalta = Riberalta.';
62 - PuertoRico = PuertoRico.';
63 - Cachuela = Cachuela.';
64 - Guayaramerin = Guayaramerin';
65
66 - Network.Cachuela = cLoad('CES','0',real(Cachuela(1)),imag(Cachuela(1)),Vrated);
67 - Network.Guayaramerin = cLoad('GUA','0',real(Guayaramerin(1)),imag(Guayaramerin(1)),Vrated);
68 - Network.Riberalta = cLoad('RIB','0',real(Riberalta(1)),imag(Riberalta(1)),Vrated);
69 - Network.PuertoRico = cLoad('PRC','0',real(PuertoRico(1)),imag(PuertoRico(1)),Vrated);
70 - Network.Cobija = cLoad('COB','0',real(Cobija(1)),imag(Cobija(1)),Vrated);

```

Figure 3.50 : fichier *NetworkDefinition.m* (2/2)

En exécutant le fichier *NetworkDefinition.m*, on obtient un réseau triphasé équilibré ayant les caractéristiques présentées dans le Tableau 3.2

Tableau 3.2 : Exemple I : données du réseau

cVsource (Barre d'équilibre)											
CASE N°	ID	ON	Bus k	Bus m	function	Vabs (V)	Angle (deg)				
y2015	HydroCachuela	TRUE	CES_MT	GND	0	13 800,00	0,00				
cTxfoZ											
CASE N°	ID	ON	Bus k	Bus m	function	Bus i	Bus j	r	Angle (deg)	R [ohm]	X [ohm]
y2015	TxfoCachuela	TRUE	CES_MT	GND	0	CES	GND	13,8/230	0,00	0,06	0,35
cLine											
CASE N°	ID	ON	Bus k	Bus m	function	R [ohm]	X [ohm]	G[S]	B [S]		
y2015	Z_CES_GUA	TRUE	CES	GUA	0	3,04	16,78	0,00	1,14E-04		
y2015	Z_CES_RIB	TRUE	CES	RIB	0	6,20	34,18	0,00	2,33E-04		
y2015	Z_RIB_PRC	TRUE	RIB	PRC	0	17,53	96,73	0,00	6,59E-04		
y2015	Z_PRC_COB	TRUE	PRC	COB	0	11,70	64,53	0,00	4,40E-04		
cLoad											
CASE N°	ID	ON	Bus k	Bus m	function	P (W)	Q (Var)	Vnom (V)			
y2015	Cachuela	TRUE	CES	GND	0	74 000,00	45 861,08	230 000,00			
y2015	Guayaramerin	TRUE	GUA	GND	0	4 390 000,00	2 720 677,65	230 000,00			
y2015	Riberalta	TRUE	RIB	GND	0	7 315 000,00	4 533 429,84	230 000,00			
y2015	PuertoRico	TRUE	PRC	GND	0	741 000,00	459 230,55	230 000,00			
y2015	Cobija	TRUE	COB	GND	0	13 637 000,00	8 451 453,54	230 000,00			
y2015	CompGUA	TRUE	GUA	GND	0	0,00	0,00	230 000,00			
y2015	CompRIB	TRUE	RIB	GND	0	0,00	20 000 000,00	230 000,00			
y2015	CompPRC	TRUE	PRC	GND	0	0,00	20 000 000,00	230 000,00			
y2015	CompCOB	TRUE	COB	GND	0	0,00	10 000 000,00	230 000,00			

La source idéale *HydroCachuela* de la classe *cVsource* est la barre d'équilibre du réseau. Les classes *cLine* et *cTxfoZ* sont créées à partir des classes APREM pour simplifier le code utilisé pour construire le réseau de l'exemple I. La classe *cLine* permet de créer un modèle de ligne en pi séquence direct en assemblant trois objets *cImpedance*. La classe *cTxfoZ*, quant à elle, permet de modéliser un transformateur non idéal en rajoutant une impédance à son modèle. L'objet créé par la classe *cTxfoZ* est constitué de deux objets, soit un transformateur idéal et une impédance respectivement des classes *cTxfo* et *cImpedance*.

Le même réseau est également modélisé dans le logiciel PSAF [22] avec les mêmes paramètres et caractéristiques électriques pour valider les résultats obtenus (voir annexe 1), bien que le logiciel PSAF travaille en per-unit et APREM avec les valeurs réelles.

Les simulations

Une fois le réseau constitué, les écoulements de puissance peuvent être lancés afin de simuler l'évolution des charges sur 30 années et observer leur effet sur la ligne. Mais avant, il est nécessaire d'ajouter de la compensation sur la ligne dans chaque localité, sans quoi les niveaux de tension observés sont supérieurs à 1.05 p.u., soit >241 kV. La Figure 3.51 présente à cet effet les niveaux de tension obtenus sans compensation.

Bus Data		
Voltage :		
Bus ID	Mag (V)	Ang (deg)
GND	0.000	0.000
CES_MT	13800.000	-0.000
CES	262361.311	-2.748
GUA	262387.941	-2.813
RIB	271886.496	-3.763
PRC	289447.087	-5.487
COB	291135.799	-6.167

Figure 3.51 : niveau de tension sur la ligne sans compensation

Compensation

Dans le fichier principal, Bolivie_MAIN (Figure 3.52), les bancs de compensation (inductances) sont créés sous les identifications *CompRIB*, *CompPRC* et *CompCOB* avec comme valeur initiale 20Mvar afin de réduire la tension de chaque nœud sous 1.05 pu. Il n'est néanmoins pas nécessaire de compenser le nœud Guyaramin d'après les résultats obtenus précédemment. *CompGUY* est donc fixé à 0 var. Le tronçon de ligne qui sépare ce nœud et le site de production étant court, aucune hausse importante de tension n'est observé.

```

5 -  initialisation
6 -  Vrated = 230e3;          * V
7 -  Sbase = 10e6;           * VA
8 -  Zbase = Vrated^2/Sbase; * Ohm
9
10 %% CREATION DU NETWORK
11 - networkDefinition
12 - Network = fLoadFlow(Network);
13 - fSave(Network);
14
15 %% AJOUT DE LA COMPENSATION
16 - Network.CompGUA = cLoad('GUA','0',0,0,Vrated);
17 - Network.CompRIB = cLoad('RIB','0',0,20e6,Vrated);
18 - Network.CompPRC = cLoad('PRC','0',0,20e6,Vrated);
19 - Network.CompCOB = cLoad('COB','0',0,20e6,Vrated);

```

Figure 3.52 : fichier Bolivie_MAIN.m (1/3)

Vecteurs de résultats

Avant le lancement des simulations, des vecteurs résultats sont initialisés (ligne 22 à 33, Figure 3.53) en vue d'accueillir certains résultats d'écoulement de puissance. Ces vecteurs seront ensuite utilisés pour présenter graphiquement les résultats à la fin des simulations.

```

21 %% TABLEAUX RÉSULTATS
22 - Vrib = [] ; % Tension Riberalta
23 - Vprc = [] ; % Tension PuertoRico
24 - Vcob = [] ; % Tension Cobija
25 - Vgua = [] ; % Tension Guayaramerin
26 - Vces = [] ; % Tension Cachuela
27 - Losses_CES_COB = [] ; % Pertes ligne Cachuela-Cobija
28 - Losses_CES_GUA = [] ; % Pertes ligne Cachuela-Guayaramerin
29 - CompRIP = [] ; % Compensation Riberalta
30 - CompPRC = [] ; % Compensation PuertoRico
31 - CompCOB = [] ; % Compensation Cobija
32 - CompGUA = [] ; % Compensation Guayaramerin
33 - Gen = [] ; % Generation

```

Figure 3.53 : fichier Bolivie_MAIN.m (2/3)

Boucle **for**

Après cette configuration, les calculs d’écoulement de puissance sur le réseau sont lancés repris en boucle pour chaque année de simulation. Les écoulements de puissance sont réalisés en appelant la fonction **fLoadFlow** à l’intérieur de la boucle **for** (ligne 39 à 59, Figure 3.54).

Les premières lignes 41 à 45 à l’intérieur de la boucle **for** modifient les charges à chaque passage en assignant les valeurs des tableaux définis dans le fichier *networkDefinition.m*. Le programme lance le calcul d’écoulement de puissance avec **fLoadFlow** et exécute ensuite le fichier *Compensation.m* (ligne 49) qui évalue la compensation requise à chaque nœud pour maintenir le niveau de tension sur la ligne dans la plage 0.95 à 1.05 pu (Figure 3.55). L’algorithme vérifie le niveau de tension à Cobija et Guayaramerin et décrémente la compensation par bloc de 10Mvar en remontant la ligne de Cobija à Riberalta [*CompCOB*, *CompPRC* et *CompRIB*] et à Guayaramerin [*CompGUA*].

```

35 %% BOUCLE DE LOADFLOW
36 % Simule l'augmentation de la charge sur la ligne entre l'année 2015 et
37 % 2045. Un algorithme reproduit le retrait graduel de la compensation afin
38 % de maintenir la tension entre 0.95 et 1.05 pu sur la ligne.
39 - for i = 1:length(Cobija)
40     % Modification de la charge
41 -     Network.Guayaramerin.S = Guayaramerin(i);
42 -     Network.Cachuela.S = Cachuela(i);
43 -     Network.Riberalta.S = Riberalta(i);
44 -     Network.PuertoRico.S = PuertoRico(i);
45 -     Network.Cobija.S = Cobija(i);
46     % LoadFlow
47 -     Network = fLoadFlow(Network);
48     % Algorithme de gestion de la compensation
49 -     Compensation;
50     % Sauvegarde des résultats
51 -     Vgua = [Vgua Network.Guayaramerin.V];
52 -     Vrib = [Vrib Network.Riberalta.V];
53 -     Vprc = [Vprc Network.PuertoRico.V];
54 -     Vcob = [Vcob Network.Cobija.V];
55 -     Vces = [Vces Network.Cachuela.V];
56 -     Gen = [Gen Network.HydroCachuela.S];
57 -     Losses; %% calcul les pertes en % sur la ligne
58     % Sauvegarde de tous les résultats de simulations
59 -     fSave(Network,['y' num2str(2014+i)]);i
60 - end
61 %% CREATION DES REPORTS POUR CHAQUE SIMULATION
62 - fReport(fSave,'Bolivie_230kV_Line_002.txt');
63 - open Bolivie_230kV_Line_002.txt
64 %% IMPRESSION DES RESULTATS
65 - impressionDesResultats
66 %% EXPORT VERS EXCEL
67 - fReportxls(fSave,'Bolivie_230kV_Line.xls');

```

Figure 3.54 : fichier Bolivie_MAIN.m (3/3)

Ces lignes permettent ainsi d'automatiser la régulation de la compensation au fur et à mesure des simulations. Il est néanmoins important de soulever qu'à chaque fois que la compensation à un nœud est ajustée pour rétablir la tension sur la ligne dans la plage 0.95 à 1.05 pu, le programme valide les résultats escomptés en relançant la fonction **fLoadflow** tant et aussi longtemps que le niveau de tension est hors de la plage.

<pre> 1 %% Determine the compensation needed 2 - pu = abs(Network.Guayaramerin.V)/Vrated; 3 - if pu < 0.95 4 - for Comp = 1e-10:10e6:21e6 5 - Network.CompGUA.Q = Comp; 6 - Network = fLoadFlow(Network); 7 - if abs(Network.Guayaramerin.V)/Vrated < 0.95 8 - break 9 - end 10 - end 11 - end 12 - CompGUA = [CompGUA Network.CompGUA.Q]; 13 14 %% Determine the compensation needed 15 - pu = abs(Network.Cobija.V)/Vrated; 16 - if pu < 0.95 && Network.CompCOB.Q ~= 0 17 - for Comp = 20e6:-10e6:0 18 - Network.CompCOB.Q = Comp; 19 - Network = fLoadFlow(Network); 20 - pu = abs(Network.Cobija.V)/Vrated; 21 - if pu > 0.95 22 - break 23 - end 24 - end 25 - end 26 - CompCOB = [CompCOB Network.CompCOB.Q]; 27 28 %% Determine the compensation needed 29 - pu = abs(Network.Cobija.V)/Vrated; 30 - if pu < 0.95 && Network.CompPRC.Q ~= 0 31 - for Comp = 20e6:-10e6:0 32 - Network.CompPRC.Q = Comp; 33 - Network = fLoadFlow(Network); 34 - pu = abs(Network.Cobija.V)/Vrated; 35 - if pu > 0.95 36 - break 37 - end 38 - end 39 - end 40 - CompPRC = [CompPRC Network.CompPRC.Q]; </pre>	<pre> 43 %% Determine the compensation needed 44 - pu = abs(Network.Cobija.V)/Vrated; 45 - if pu < 0.95 && Network.CompRIB.Q ~= 0 46 - for Comp = 20e6:-10e6:0 47 - Network.CompRIB.Q = Comp; 48 - Network = fLoadFlow(Network); 49 - pu = abs(Network.Cobija.V)/Vrated; 50 - if pu > 0.95 51 - break 52 - end 53 - end 54 - end 55 - CompRIP = [CompRIP Network.CompRIB.Q]; 56 57 58 %% Determine the compensation needed 59 - pu = abs(Network.Cobija.V)/Vrated; 60 - if pu < 0.95 61 - for Comp = 0:-10e6:-20e6 62 - Network.CompCOB.Q = Comp; 63 - Network = fLoadFlow(Network); 64 - pu = abs(Network.Cobija.V)/Vrated; 65 - if pu > 0.95 66 - break 67 - end 68 - end 69 - CompCOB(end) = Network.CompCOB.Q; 70 - end </pre>
---	--

Figure 3.55 : fichier compensation.m

Une fois la compensation évaluée, les résultats de l'écoulement de puissance sont ensuite insérés dans les vecteurs V_{gua} , V_{rib} , V_{prc} , V_{cob} , V_{ces} et Gen (ligne 51 à 56). Le programme évalue également les pertes enregistrées sur les lignes pour chaque année de simulation en exécutant le fichier *Losses.m* (ligne 57). Ces pertes sont exprimées en pourcentage de la charge totale raccordée à la ligne.

```

1 - Losses_CES_COB = [Losses_CES_COB...
2     real(Network.Z_CES_RIB.S...
3     + Network.Z_RIB_PRC.S...
4     + Network.Z_PRC_COB.S)...
5     / ...
6     (Network.Riberalta.P...
7     + Network.PuertoRico.P...
8     +Network.Cobija.P)];
9
10 - Losses_CES_GUA = [Losses_CES_GUA...
11     real(Network.Z_CES_GUA.S)...
12     / ...
13     (Network.Guayaramerin.P)];

```

Figure 3.56 : fichier losses.m

Sauvegarde des résultats

La sauvegarde des résultats de l'écoulement de puissance pour le réseau est réalisée par la fonction *fSave* (ligne 59, Bolivie_MAIN.m). Cette dernière transforme chaque objet du réseau en une structure de données Matlab (ligne 28, fSave.m). C'est un format pour lequel Matlab offre des outils d'affichage et de traitement utiles pour l'analyse postérieure des résultats de simulations et qui ne requiert pas l'utilisation de l'outil APREM. Ce format va entre autres être exploité pour la préparation des rapports et l'exportation des données vers d'autres logiciels comme Excel.

```

1 function varargout=fSave(varargin)
2
3 - persistent CaseNumber SavedData
4 - if isempty(CaseNumber)
5 -     CaseNumber = 0; SavedData = [];
6 - end
7 - switch nargin
8 - case 0
9 -     varargout{1} = SavedData;
10 -    return;
11 - case {1,2}
12 -     if ischar(varargin{1}) && strcmp(varargin{1}, 'ini')
13 -         CaseNumber = 0;
14 -         SavedData = [];
15 -         return;
16 -     end
17 -     Network = varargin{1};
18 -     CaseNumber = CaseNumber + 1;
19 -     if nargin == 2
20 -         LF = varargin{2}; else
21 -             LF = ['Case_' num2str(CaseNumber)];
22 -         end
23 -     %% Node voltage
24 -     SavedData.(LF).Node = fNode;
25 -     %% Equipment Data
26 -     index = fieldnames(Network); % returns the names of equipments
27 -     for n = 1:length(index)
28 -         SavedData.(LF).(index(n))=struct(Network.(index(n)));
29 -         SavedData.(LF).(index(n)).class=class(Network.(index(n)));
30 -     end
31 -     varargout{1} = SavedData;
32 - end

```

Figure 3.57 : fonction fSave.m

La Figure 3.58 montre le contenu de la structure SavedData retournée par la fonction *fSave* après la réalisation des 30 années de simulations. Il est ainsi possible d'observer les résultats d'écoulement de puissance de l'année y2028 pour le transformateur élévateur de la centrale Cachuela, *TxfoCachuela*, et la ligne entre Puerto Rico et Cojiba, *Z_PRC_COB*.

ans		ans.y2028		ans.y2028.TxfoCachuela		ans.y2028.Z_PRC_COB	
F...	Value	Field	Value	Field	Value	Field	Value
Case_1	<1x1 struct>	CompCOB	<1x1 struct>	ON	true	ON	true
y2015	<1x1 struct>	CompPRC	<1x1 struct>	k	'CES_MT'	k	'PRC'
y2016	<1x1 struct>	CompRIB	<1x1 struct>	m	'GND'	m	'COB'
y2017	<1x1 struct>	CompGUA	<1x1 struct>	Vk	138000+i*9.9708e-030	Vk	2.3373e+005-i*41735
y2018	<1x1 struct>	Cobija	<1x1 struct>	Vm	0	Vm	2.2954e+005-i*48336
y2019	<1x1 struct>	PuertoRico	<1x1 struct>	V	138000+i*9.9708e-030	V	4184.3+i*6600.7
y2020	<1x1 struct>	Riberalta	<1x1 struct>	I	2164.3+i*403.9	I	-1464.3-i*942.74
y2021	<1x1 struct>	Guayaramerin	<1x1 struct>	S	5.1731e+007-i*9.6543e+006	S	1.6611e+005-i*2.3573e+007
y2022	<1x1 struct>	Cachuela	<1x1 struct>	Sk	5.1731e+007-i*9.6543e+006	Sk	2.7677e+007-i*6.5232e+006
y2023	<1x1 struct>	Z_PRC_COB	<1x1 struct>	Sm	0	Sm	2.7511e+007-i*7.1442e+006
y2024	<1x1 struct>	Z_RIB_PRC	<1x1 struct>	function	"	function	"
y2025	<1x1 struct>	Z_CES_RIB	<1x1 struct>	i	'CES'	R	11.697
y2026	<1x1 struct>	Z_CES_GUA	<1x1 struct>	j	'GND'	X	64.532
y2027	<1x1 struct>	TxfoCachuela	<1x1 struct>	Vi	2.3338e+005-i*21692	G	0
y2028	<1x1 struct>	HydroCachuela	<1x1 struct>	Vj	0	B	0.00043968
y2029	<1x1 struct>	Node	<1x1 struct>	Vij	2.3338e+005-i*21692	class	'cLine'
y2030	<1x1 struct>			lij	-129.86-i*24.234		
y2031	<1x1 struct>			Sij	-5.158e+007+i*1.4675e+007		
y2032	<1x1 struct>			Si	-5.158e+007+i*1.4675e+007		
y2033	<1x1 struct>			Sj	0		
y2034	<1x1 struct>			r	0.06		
y2035	<1x1 struct>			Angle	0		
y2036	<1x1 struct>			n	1		
y2037	<1x1 struct>			R	0.010355		
y2038	<1x1 struct>			X	0.34527		
y2039	<1x1 struct>			class	'cTxfoZ'		

Figure 3.58 : Contenu de la structure SavedData retournée par la fonction fSave

3.7.1.1 Exportation des résultats

Ces structures de données sont exportées intégralement dans le fichier texte *Bolivie_230kV_Line_002.txt* via la fonction **fReport** et vers le fichier Excel *Bolivie_230kV_Line.xls* via **fReportxls** (Figure 3.59). Ces deux fonctions sont bien entendu disponibles dans APREM, mais l'utilisateur est libre de personnaliser ou créer sa propre fonction d'exportation en traitant la structure de données renvoyée par **fSave**.

1	CASE N°	ID	ON/OFF	V	DEG	A	DEG	W	VAR
10	y2015	Cobija	VRAI	231 224	-6	40	-38	13 637 000	8 451 454
11	y2020	Cobija	VRAI	232 285	-9	66	-41	22 580 000	13 993 827
13	y2025	Cobija	VRAI	222 779	-11	78	-43	25 547 000	15 832 609
14	y2030	Cobija	VRAI	229 974	-13	85	-45	28 904 000	17 913 090
15	y2040	Cobija	VRAI	223 501	-17	112	-49	36 999 000	22 929 921
16	y2045	Cobija	VRAI	219 111	-19	130	-51	41 861 000	25 943 118
22	y2015	CompPRC	VRAI	233 809	-5	49	-95	0	20 000 000
23	y2020	CompPRC	VRAI	234 095	-8	49	-98	0	20 000 000
25	y2025	CompPRC	VRAI	225 659	-9	51	-99	0	20 000 000
27	y2030	CompPRC	VRAI	233 338	-11	25	-101	0	10 000 000
28	y2040	CompPRC	VRAI	229 108	-14	0	0	0	0
94	y2045	CompPRC	VRAI	226 171	-16	0	0	0	0
95	y2015	ComprIB	VRAI	233 898	-4	49	-94	0	20 000 000
96	y2020	ComprIB	VRAI	233 423	-5	49	-95	0	20 000 000
102	y2025	ComprIB	VRAI	227 555	-6	51	-96	0	20 000 000
103	y2030	ComprIB	VRAI	231 311	-7	50	-97	0	20 000 000
105	y2040	ComprIB	VRAI	227 128	-10	51	-100	0	20 000 000
107	y2045	ComprIB	VRAI	226 915	-11	25	-101	0	10 000 000
187	y2015	Guayaramerin	VRAI	234 166	-3	13	-35	4 390 000	2 720 678
188	y2020	Guayaramerin	VRAI	233 779	-4	16	-36	5 368 000	3 326 788
254	y2025	Guayaramerin	VRAI	229 329	-5	18	-37	6 222 000	3 856 049
255	y2030	Guayaramerin	VRAI	232 003	-6	21	-38	7 213 000	4 470 216
256	y2040	Guayaramerin	VRAI	228 593	-8	27	-39	9 186 000	5 692 971
263	y2045	Guayaramerin	VRAI	228 217	-9	31	-40	10 393 000	6 441 003
265	y2015	HydroCachuela	VRAI	13 800	0	1 183	-159	-26 375 386	10 214 379
267	y2020	HydroCachuela	VRAI	13 800	0	1 710	-167	-39 869 402	9 021 913
268	y2025	HydroCachuela	VRAI	13 800	0	1 963	178	-46 879 927	-1 851 193
414	y2030	HydroCachuela	VRAI	13 800	0	2 319	-176	-55 260 832	4 168 484
415	y2040	HydroCachuela	VRAI	13 800	0	2 985	176	-71 179 024	-5 081 885
416	y2045	HydroCachuela	VRAI	13 800	0	3 394	175	-80 823 233	-6 865 357

Figure 3.59 : aperçu du fichier Bolovie_230kV_Line.xls

En utilisant Matlab, l'utilisateur de l'outil APREM a également accès à tous les outils graphiques disponibles sur Matlab avec lesquels il sera en mesure de présenter ses résultats. Les

Figure 3.60 et 3.61 proposent quelques exemples de résultats graphiques tirés du projet de ligne en Bolivie.

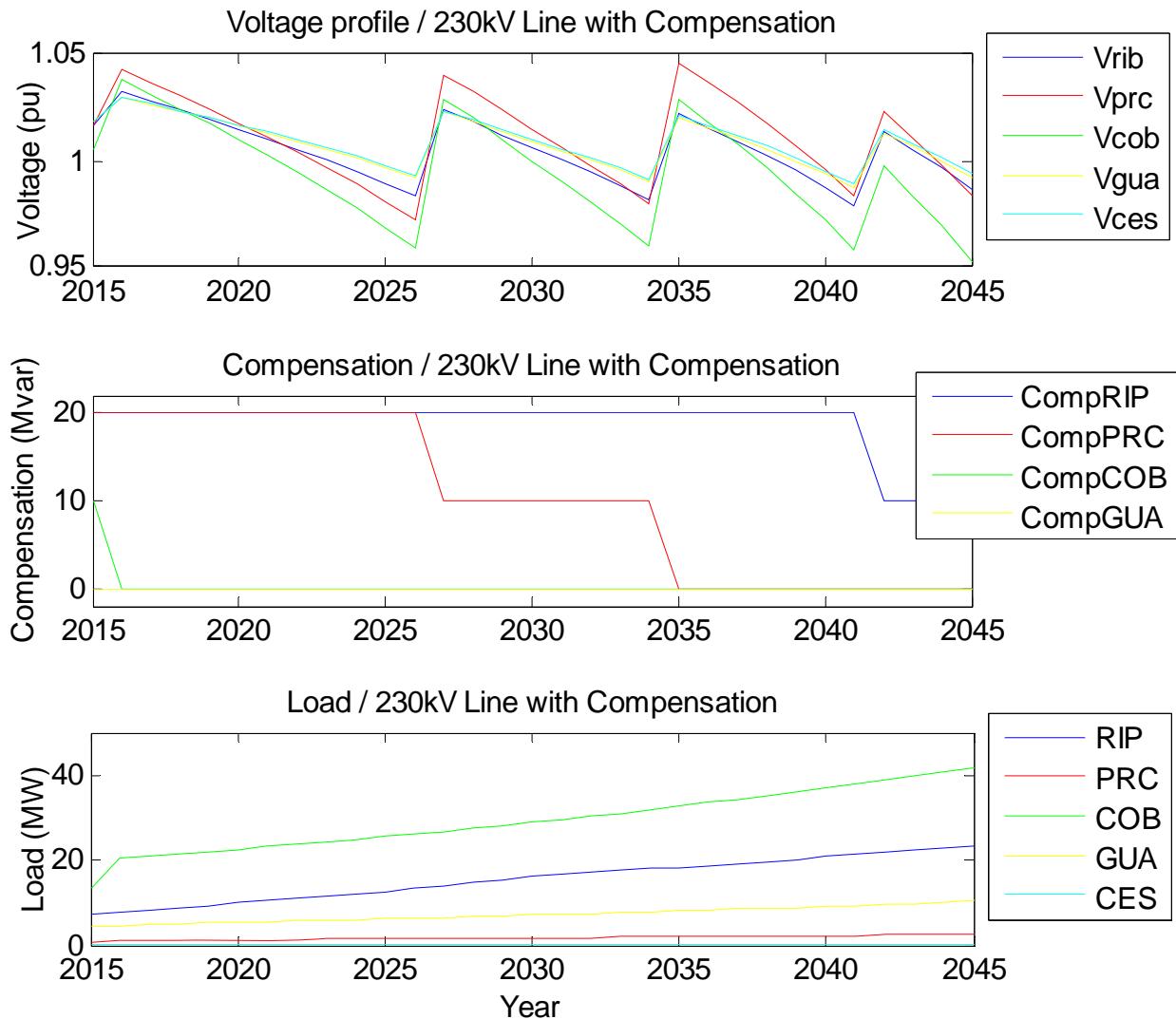


Figure 3.60 : Profile de tension, compensation et charge (ex : ligne 230kV, Bolivie)

À la Figure 3.60, il est alors possible d'observer l'évolution des différentes charges et remarquer que chacune d'entre elle croît à un rythme différent. Les graphiques permettent de visualiser le retrait de la compensation réactive aux différents nœuds et leur effet sur le niveau de tension de la ligne.

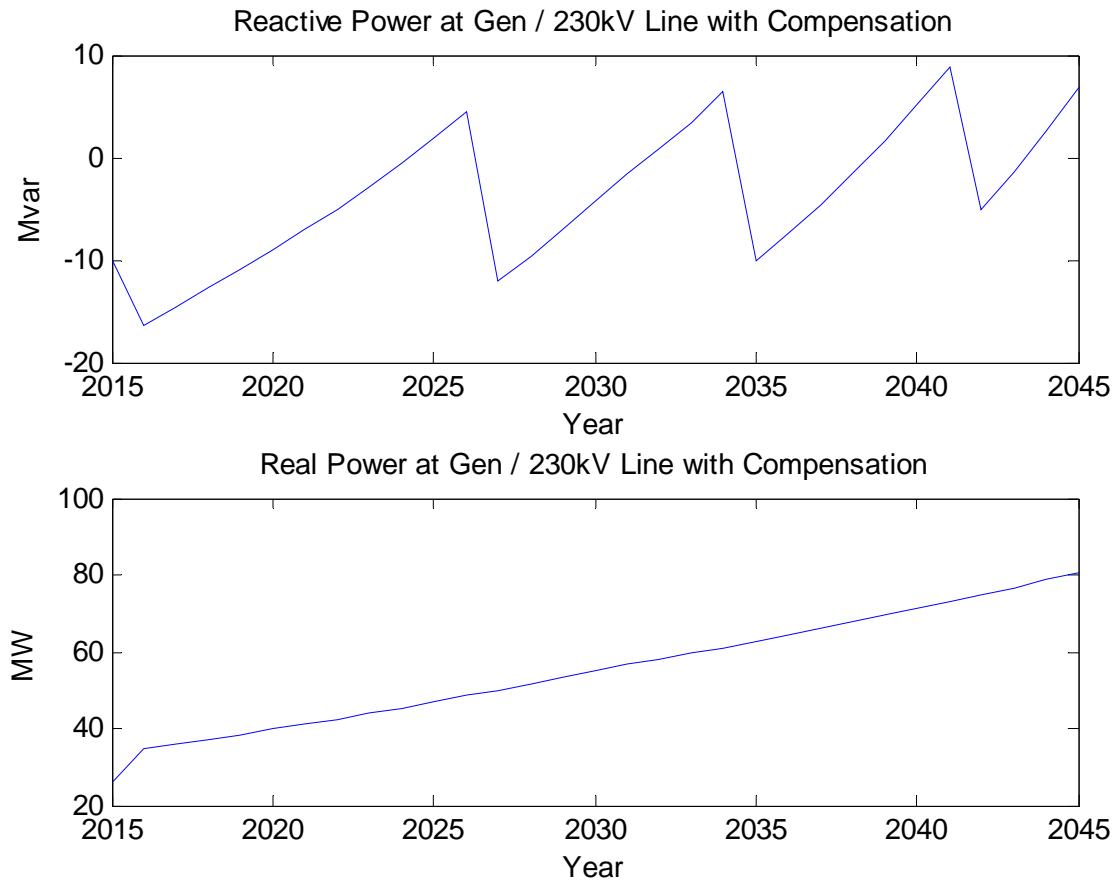


Figure 3.61 : Puissance réelle et réactive à la centrale (ex : ligne 230kV, Bolivie)

La Figure 3.61 montre l'accroissement année après année de la production au niveau de la centrale hydroélectrique de Cachuela. Elle double sur la période de 30 ans. On remarque également les variations de la puissance réactive au niveau de la production dont les changements marqués correspondent au retrait de la compensation. L'information permet ainsi d'avoir une idée sur les conditions d'opération moyen du/des groupes en régime permanent.

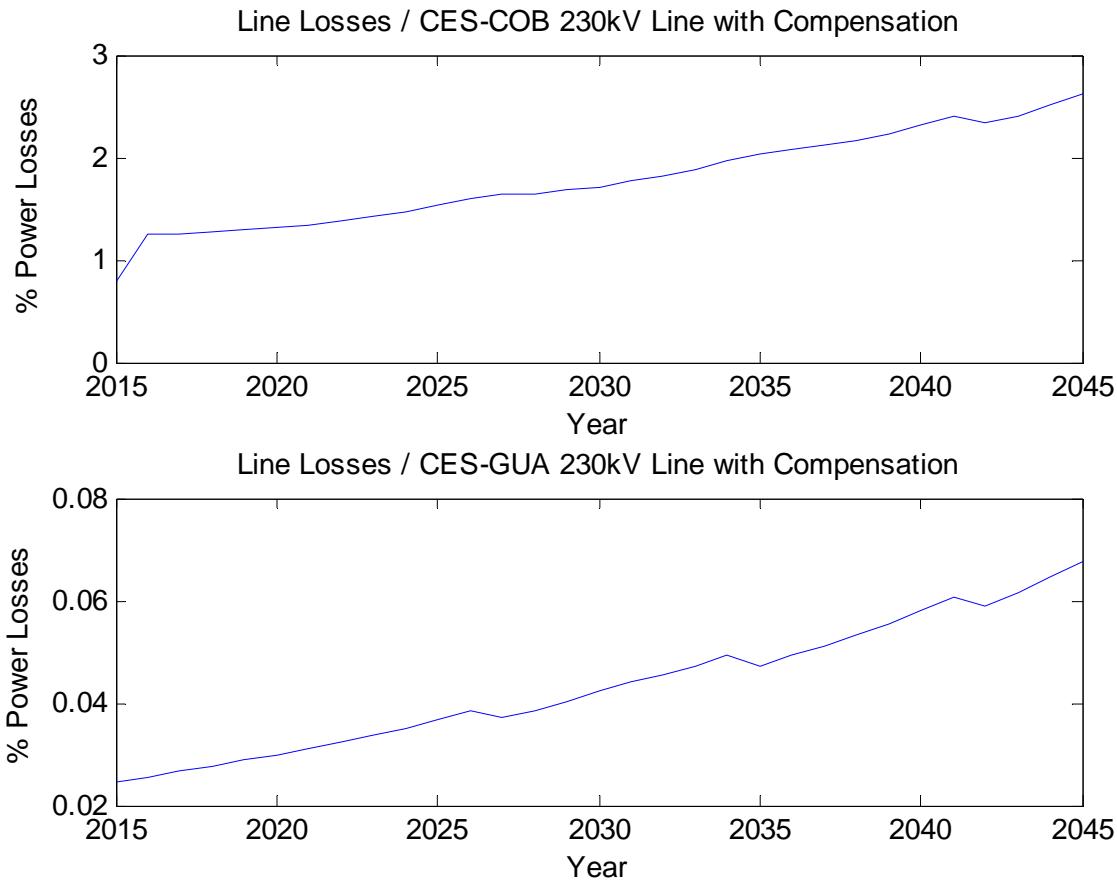


Figure 3.62 : Pertes sur la ligne 230kV (ex : ligne 230kV, Bolivie)

L'autre Figure 3.62, elle, présente les pertes enregistrées sur les lignes. Elles sont exprimées en % de la puissance transmise. Les pertes sur la ligne entre Cachuela et Guyaramin se situent en deçà de 1% et sur le tronçon Cachuela-Cobija en deçà de 3% sur la période de 30 ans.

L'utilisation de Matlab pour la production de graphique permet ainsi de personnaliser complètement la présentation des résultats suivant les besoins de l'utilisateur.

Le lecteur comprendra par cette exemple qu'il aurait été possible d'entrer dans les détails mensuels, voire horaire, de la courbe de charge de chaque localité et ainsi apprécier le rôle de la compensation suivant un échantillonnage plus important. Cette même précision peut également être apportée à l'algorithme de compensation, qui, dans l'exemple est simplifié pour les besoins de démonstration. Cependant, un impact important sur la durée des simulations est à prévoir plus le volume de donnée à traiter est important. Les possibilités qu'offre Matlab dans la modélisation

des systèmes sont immenses. Libre à l'utilisateur d'user de ces fonctionnalités pour répondre à ses besoins.

3.7.1.2 Validation des résultats avec le logiciel PSAF

Afin de valider les résultats obtenus avec l'outil APREM, le réseau de l'exemple I est simulé sur PSAF pour l'année 2015. Les résultats obtenus avec PSAF et l'outil APREM sont présentés respectivement aux annexes 1 et 2. Les tableaux 3.3 et 3.4 présentent les niveaux de tensions des barres et le transit de puissance sur les lignes obtenus sur chacun des logiciels en appliquant la méthode de Newton-Raphson pour solutionner l'écoulement de puissance. Les résultats, incluant ceux fournis en annexe, sont semblables et confirment donc la justesse des résultats obtenus avec l'outil APREM.

Tableau 3.3 : Exemple I : niveaux de tension sur les barres obtenus sur APREM et PSAF

	APREM		PSAF	
	V sol [kV]	Ang sol [deg]	V sol [kV]	Ang sol [deg]
CES_MT	13,800	0,000	13,80	0,00
CES	234,194	-2,723	234,19	-2,70
GUÀ	234,166	-2,801	234,17	-2,80
RIB	233,898	-3,516	233,90	-3,50
PRC	233,809	-5,026	233,81	-5,00
COB	231,224	-5,876	231,22	-5,90

Tableau 3.4 : Exemple I : transits de puissance obtenus sur APREM et PSAF

1	APREM		PSAF	
	P [MW]	Q [MVAR]	P [MW]	Q [MVAR]
Z_PRC_COB	4,391	-3,543	4,39	-3,54
Z_RIB_PRC	21,867	-8,168	21,87	-8,17
Z_CES_GUA	14,497	-20,245	14,50	-20,24
Z_CES_RIB	13,688	-5,042	13,69	-5,04

3.7.2 Exemple II : projet étudiant - cours de fiabilité

Dans le cadre du cours de fiabilité ELE6412 dispensé à l'école polytechnique de Montréal, les étudiants devaient étudier l'effet de la reconfiguration d'un réseau suite à une défaillance de ce dernier. Le réseau de distribution monophasé de séquence direct étudié est un réseau en étoile composé de quatre (4) artères principales et d'une ligne de relève (Figure 3.63). Les caractéristiques sont présentées au tableau 3.5. Les résultats de l'écoulement de puissance en condition normal d'opération sont disponibles en annexe 3.

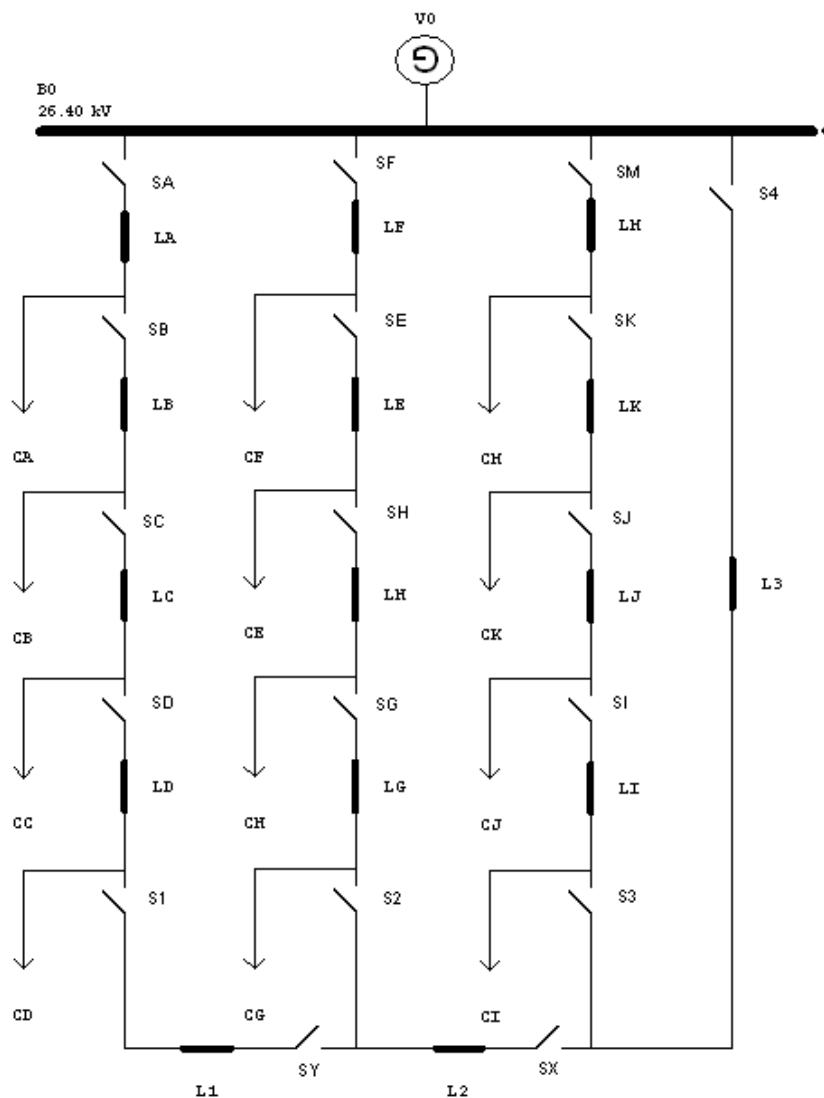


Figure 3.63 : exemple II, étude de fiabilité d'un réseau de distribution [39]

Source : F. Chartrand, "Étude de fiabilité d'un réseau de distribution électrique avec une méthode probabiliste," Projet étudiant, cours ELE6412, École Polytechnique de Montréal

Tableau 3.5 : Exemple II : données du réseau

cVsource							
ID	ON	Bus k	Bus m	function	Vabs (V)	Angle (deg)	n
V0	VRAI	B0	GND	0	26 400.00	0.00	1

cImpedance						
ID	ON	Bus k	Bus m	function	R (ohm)	X (ohm)
LA	VRAI	Sa	Ca	FOR01	0.27878	0.27878
LB	VRAI	Sb	Cb	FOR01	0.27878	0.27878
LC	VRAI	Sc	Cc	FOR01	0.27878	0.27878
LD	VRAI	Sd	Cd	FOR01	0.27878	0.27878
LE	VRAI	Se	Ce	FOR01	0.27878	0.27878
LF	VRAI	Sf	Cf	FOR01	0.27878	0.27878
LG	VRAI	Sg	Cg	FOR01	0.27878	0.27878
LH	VRAI	Sh	Ch	FOR01	0.27878	0.27878
LI	VRAI	Si	Ci	FOR01	0.27878	0.27878
LJ	VRAI	Sj	Cj	FOR01	0.27878	0.27878
LK	VRAI	Sk	Ck	FOR01	0.27878	0.27878
LM	VRAI	Sm	Cm	FOR01	0.27878	0.27878
L1	VRAI	S1	Sy	FOR01	0.27878	0.27878
L2	VRAI	S2	Sx	FOR01	0.27878	0.27878
L3	VRAI	S3	S4	FOR01	0.27878	0.27878

cSwitch						
ID	ON	Bus k	Bus m	function	state	n
SA	VRAI	B0	Sa	FOR005	CLOSE	1.00
SB	VRAI	Ca	Sb	FOR005	CLOSE	2.00
SC	VRAI	Cb	Sc	FOR005	CLOSE	3.00
SD	VRAI	Cc	Sd	FOR005	CLOSE	4.00
SE	VRAI	Cf	Se	FOR005	CLOSE	5.00
SF	VRAI	B0	Sf	FOR005	CLOSE	6.00
SG	VRAI	Ch	Sg	FOR005	CLOSE	7.00
SH	VRAI	Ce	Sh	FOR005	CLOSE	8.00
SI	VRAI	Cj	Si	FOR005	CLOSE	9.00
SJ	VRAI	Ck	Sj	FOR005	CLOSE	10.00
SK	VRAI	Cm	Sk	FOR005	CLOSE	11.00
SM	VRAI	B0	Sm	FOR005	CLOSE	12.00
S1	VRAI	Cd	S1	FOR005	OPEN	13.00

S2	VRAI	Cg	S2	FOR005	OPEN	14.00
S3	VRAI	Ci	S3	FOR005	OPEN	15.00
S4	VRAI	B0	S4	FOR005	OPEN	16.00
SX	VRAI	Sx	S3	FOR005	OPEN	17.00
SY	VRAI	Sy	S2	FOR005	OPEN	18.00

cLoad							
ID	ON	Bus k	Bus m	function	P (W)	Q (Var)	Vnom (V)
CA	VRAI	Ca	GND	0.00	4 700 000.00	0.00	26 400.00
CB	VRAI	Cb	GND	0.00	3 300 000.00	0.00	26 400.00
CC	VRAI	Cc	GND	0.00	8 200 000.00	0.00	26 400.00
CD	VRAI	Cd	GND	0.00	4 800 000.00	0.00	26 400.00
CE	VRAI	Ce	GND	0.00	4 600 000.00	0.00	26 400.00
CF	VRAI	Cf	GND	0.00	3 600 000.00	0.00	26 400.00
CG	VRAI	Cg	GND	0.00	5 600 000.00	0.00	26 400.00
CH	VRAI	Ch	GND	0.00	4 500 000.00	0.00	26 400.00
CI	VRAI	Ci	GND	0.00	5 800 000.00	0.00	26 400.00
CJ	VRAI	Cj	GND	0.00	3 200 000.00	0.00	26 400.00
CK	VRAI	Ck	GND	0.00	5 500 000.00	0.00	26 400.00
CM	VRAI	Cm	GND	0.00	6 200 000.00	0.00	26 400.00

Le projet étudiant de Francis Chartrand [39], repris ici, montre l'impact de la topologie dans l'analyse de la fiabilité. Le réseau étudié est composé de la source de tension idéale Vs, des lignes L_x, des interrupteurs S_x et des charges C_x. Les interrupteurs S₁ à S₄ de la ligne de relève sont normalement ouverts.

La méthode utilisée par l'auteur pour étudier la fiabilité du réseau est l'approche Monte-Carlo qui consiste à réaliser un nombre important de simulations au cours desquelles les équipements sont mis en défaut aléatoirement, tout en respectant une probabilité de défaillance qui leur est assignée. Un FOR de 0,01 et 0,005 est respectivement retenu pour les câbles et les interrupteurs afin de simuler les pannes survenant durant l'exploitation. Ainsi, lorsqu'un câble est en défaut, les interrupteurs aux extrémités s'ouvrent pour isoler le défaut et lorsqu'un interrupteur est en défaut, la panne se traduit par l'ouverture de l'interrupteur. La source et les charges sont quant à eux toujours actives.

Quatre différents scénarios sont étudiés à l'aide de l'outil APREM (dénommé MATLF dans l'article), mais seulement deux sont présentés ici. Le premier consiste à désactiver la reconfiguration automatique du réseau en cas de défaut. Ainsi, lorsqu'un interrupteur ou un câble est défaillant, les charges en aval du défaut ne sont plus alimentées. Le deuxième scénario, lui, exploite la ligne de relève en utilisant un algorithme de reconfiguration automatique qui permet d'isoler l'équipement en défaut et réalimenter les artères par leur extrémité.

Pour assurer une convergence des simulations de Monte-Carlo, l'auteur fixe le nombre de simulations à 5000. C'est le nombre approximatif de simulations qu'il détermine en observant la variation du taux de défaut (ΔLOLP) sur le nombre de simulations réalisées.

Les résultats obtenus pour les deux scénarios étudiés sont repris dans les tableau 3.6 et 3.7. La formule utilisée pour évaluer le LOLP de chacune des charges va comme suit :

$$\text{LOLP} = \frac{\text{nombre de fois que la charge } C_x \text{ n'est pas alimentée}}{\text{nombre total de simulations réalisées}} \quad (3.208)$$

Les colonnes *câble*, *interr* et *combiné* représentent les valeurs de LOLP obtenues lorsque l'indisponibilité de la charge C_x est respectivement causée par une défaillance des câbles, une défaillance des interrupteurs et la combinaison des deux. La ligne *Système* correspond quant à

elle aux valeurs de LOLP calculées pour toute indisponibilité qui survient dans le réseau, peu importe la charge.

	câble	interr	combiné
Ca	0.009	0.006	0.014
Cb	0.020	0.010	0.029
Cc	0.027	0.014	0.040
Cd	0.035	0.018	0.053
Cf	0.009	0.005	0.013
Ce	0.019	0.010	0.028
Ch	0.026	0.014	0.040
Cg	0.035	0.019	0.054
Cm	0.010	0.005	0.015
Ck	0.021	0.010	0.031
Cj	0.031	0.014	0.045
Ci	0.042	0.019	0.061
Système	0.1078	0.0546	0.1574

Tableau 3.6 : Exemple II : LOLP sans reconfiguration

	câble	interr	combiné
Ca	0.012	0.001	0.012
Cb	0.013	0.002	0.014
Cc	0.012	0.002	0.013
Cd	0.011	0.002	0.012
Cf	0.009	0.000	0.010
Ce	0.009	0.001	0.010
Ch	0.013	0.001	0.013
Cg	0.010	0.001	0.010
Cm	0.009	0.000	0.009
Ck	0.010	0.001	0.011
Cj	0.013	0.001	0.013
Ci	0.011	0.001	0.011
Système	0.1154	0.0058	0.1182

Tableau 3.7 : Exemple II :LOLP avec reconfiguration

Validation des résultats

L'auteur compare alors quelques résultats obtenus par la méthode de Monte-Carlo et l'évaluation théorique qui découle de la formulation du problème. Par exemple, la probabilité d'indisponibilité de la charge C_a correspond à l'expression analytique suivante :

$$P|C_a| = P|S_a| \cup P|L_a| = P|S_a| + P|L_a| - P|S_a| \times P|L_a| \quad (3.209)$$

$$P|C_a| = 0.005 + 0.01 - 0.005 \times 0.01 = 0.01495 \quad (3.210)$$

La valeur obtenue est alors comparable à celle obtenue par la méthode de Monte-Carlo, soit 0.014, dans le cas du premier scénario en l'absence de reconfiguration automatique.

Les résultats obtenus dans les tableaux 3.5 et 3.6 démontrent indéniablement l'effet bénéfique de l'utilisation de la ligne de relève dans la fiabilité du réseau de distribution. L'impact

est encore plus marqué pour les charges éloignées de la source. Par exemple, le LOLP de la charge C_d passe de 0.054 à 0.012, ce qui correspond à une réduction de 4 fois la probabilité de voir une indisponibilité à ce point sur le réseau. Le même constat s'applique pour les charges C_g et C_i .

L'exercice a ainsi permis de démontrer les possibilités qu'offre l'outil APREM pour l'étude de la fiabilité selon l'approche de Monte-Carlo, mais également sa convivialité de puisque le projet a été réalisé alors que l'outil ne disposait que de très peu de documentation sur son fonctionnement.

3.7.3 Exemple III : outil APREM versus PSS/E

Un autre exercice de validation réalisé par Mourad Houari, étudiant au BAC à l'École polytechnique [40], a consisté à comparer les résultats d'écoulement de puissance obtenus sur APREM avec ceux tirés de PSS/E. Le réseau étudié est schématiquement présenté dans la Figure 3.64

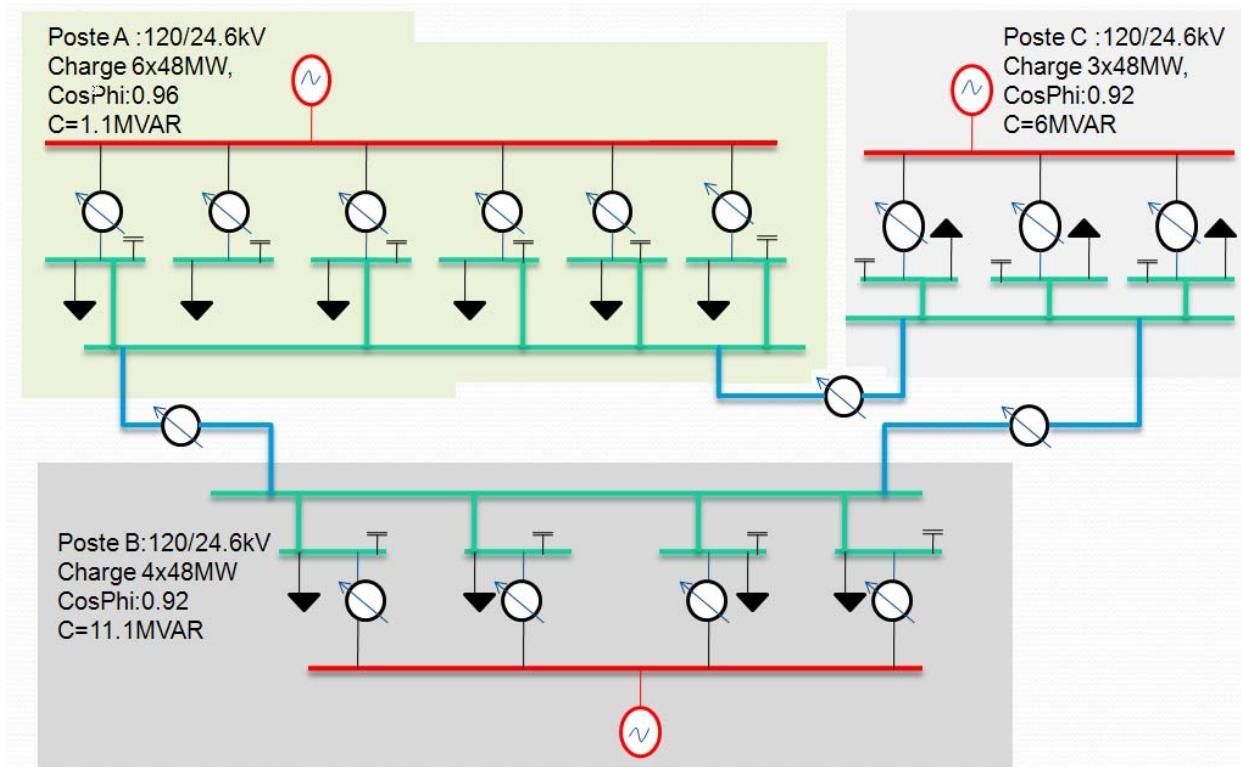


Figure 3.64 : exemple III, outil APREM versus PSS/E

La barre d'équilibre est au poste A et les postes B et C sont connectés à des sources PV. Le réseau est triphasé et équilibré. Les données du réseau modélisé et les résultats de l'écoulement de puissance sur PSS/E et l'outil APREM sont présentés respectivement aux annexes 4 et 5. Le réseau de transport est donc modélisé sur PSS/E et dans l'outil APREM avec les paramètres et caractéristiques électriques, sauf que PSS/E travaille en per-unit et APREM directement avec les valeurs réelles. Ces données sont présentées au tableau 3.10. La méthode de Newton-Raphson est employée dans les deux cas afin d'évaluer et comparer la justesse des calculs d'écoulement de

puissance de l'outil APREM. Les tableaux 3.8 et 3.9 présentent quelques résultats et il est alors possible d'observer la similitude entre les résultats. L'étude comparative confirme ainsi l'exactitude de l'écoulement de puissance fourni par l'outil APREM.

Tableau 3.8 : Exemple III : résultats comparatifs entre l'outil APREM et PSS/E

Performances de convergence			Transit entre les postes (MW/MVAR)		
Paramètres	PSS/E	APREM	Connexion	PSS/E	APREM
Méthode	Newton	Newton	B vers A	27.8/3.00	27.7/3.02
Précision (convergence)	10^{-6}	10^{-6}	C vers A	31.1/6.6	31.1/6.6
Nombre d'itérations	3	4	C vers B	19.8/5.1	19.7/5.16

Niveau de tension (en kV)			Production des groupes (MW/MVAR)		
Nom du Poste	PSS/E	APREM	Nom groupe	PSS/E	APREM
A (HT/MT)	120/24.6	120/24.596	A	239.3/56	293.3/56
B (HT/MT)	120/24.6	120/24.596	B	200/14	200/14
B (HT/MT)	120/24.6	120/24.596	C	195/17.3	195/17.3

Tableau 3.9 : Exemple III : niveaux de tension obtenus sur APREM et PSS/E

	APREM		PSS/E	
	V sol [kV]	Ang sol [deg]	V sol [kV]	Ang sol [deg]
POST_AB	24,618	0,120	24,617	0,120
POST_AC	24,619	0,122	24,620	0,120
POST_BA	24,620	0,136	24,620	0,140
POST_BC	24,648	1,658	24,647	1,660
POST_CA	24,622	0,139	24,622	0,140
POST_CB	24,648	1,670	24,647	1,670
POSTEA	120,000	0,000	120,000	0,000
POSTEA811_1	24,596	-0,021	24,595	-0,020
POSTEA811_2	24,596	-0,021	24,595	-0,020
POSTEA811_3	24,596	-0,021	24,595	-0,020
POSTEA811_4	24,596	-0,021	24,595	-0,020
POSTEA811_5	24,596	-0,021	24,595	-0,020
POSTEA811_6	24,596	-0,021	24,595	-0,020
POSTEA911	24,616	0,105	24,615	0,100
POSTEB	120,000	1,155	120,000	1,160
POSTEB811_1	24,597	1,127	24,598	1,130
POSTEB811_2	24,597	1,127	24,598	1,130
POSTEB811_3	24,597	1,127	24,598	1,130
POSTEB811_4	24,597	1,127	24,598	1,130
POSTEB911	24,620	1,095	24,620	1,100
POSTEC	120,000	2,207	120,000	2,200
POSTEC811_1	24,595	2,170	24,595	2,170
POSTEC811_2	24,595	2,170	24,595	2,170
POSTEC811_3	24,595	2,170	24,595	2,170
POSTEC911	24,627	1,937	24,627	1,930

Tableau 3.10 : Exemple III : données du réseau

cVsource (Barre d'équilibre)										
ID	ON	Bus k	Bus m	function	Vabs (V)	Angle (deg)				
V1	TRUE	POSTEA	GND	0	120 000,00	0,00				
cPVsource										
ID	ON	Bus k	Bus m	function	P (W)	Vabs (V)				
V2	TRUE	POSTEB	GND	0	200 000 000,00	120 000,00				
V3	TRUE	POSTEC	GND	0	195 000 000,00	120 000,00				
cTxfoZ										
ID	ON	Bus k	Bus m	function	Bus i	Bus j	r	Angle (deg)	R (ohm)	X (ohm)
TC1	TRUE	POSTEC	GND	0	POSTEC811_1	GND	120/24,6	0,00	4,88	0,14
TC2	TRUE	POSTEC	GND	0	POSTEC811_2	GND	120/24,6	0,00	4,88	0,14
TC3	TRUE	POSTEC	GND	0	POSTEC811_3	GND	120/24,6	0,00	4,88	0,14
TC4	TRUE	POSTE BC	GND	0	POSTE_CB	GND	1,00	0,00	1,00	0,01
TB1	TRUE	POSTEB	GND	0	POSTEB811_1	GND	120/24,6	0,00	4,88	0,14
TB2	TRUE	POSTEB	GND	0	POSTEB811_2	GND	120/24,6	0,00	4,88	0,14
TB3	TRUE	POSTEB	GND	0	POSTEB811_3	GND	120/24,6	0,00	4,88	0,14
TB4	TRUE	POSTEB	GND	0	POSTEB811_4	GND	120/24,6	0,00	4,88	0,14
TA1	TRUE	POSTEA811_1	GND	0	POSTEA	GND	24,6/120	0,00	0,21	0,01
TA2	TRUE	POSTEA811_2	GND	0	POSTEA	GND	24,6/120	0,00	0,21	0,01
TA3	TRUE	POSTEA811_3	GND	0	POSTEA	GND	24,6/120	0,00	0,21	0,01
TA4	TRUE	POSTEA811_4	GND	0	POSTEA	GND	24,6/120	0,00	0,21	0,01
TA5	TRUE	POSTEA811_5	GND	0	POSTEA	GND	24,6/120	0,00	0,21	0,01
TA6	TRUE	POSTEA811_6	GND	0	POSTEA	GND	24,6/120	0,00	0,21	0,01
TA7	TRUE	POSTA911	GND	0	POSTE_AB	GND	1,00	0,00	1,00	0,01
TA10	TRUE	POSTA911	GND	0	POSTE_AC	GND	1,00	0,00	1,00	0,01
TA8	TRUE	POSTE_AB	GND	0	POSTE_BA	GND	1,00	0,00	1,00	0,01
TA9	TRUE	POSTE_AC	GND	0	POSTE_CA	GND	1,00	0,00	1,00	0,01
cLine										
ID	ON	Bus k	Bus m	function	R (ohm)	X (ohm)	G(S)	B(S)		
L4	TRUE	POSTEA811_1	POSTA911	0	0,0139	0,139	0,000	3,30E-03		
L5	TRUE	POSTEA811_2	POSTA911	0	0,0139	0,139	0,000	3,30E-03		
L6	TRUE	POSTEA811_3	POSTA911	0	0,0139	0,139	0,000	3,30E-03		
L7	TRUE	POSTEA811_4	POSTA911	0	0,0139	0,139	0,000	3,30E-03		
L8	TRUE	POSTEA811_5	POSTA911	0	0,0139	0,139	0,000	3,30E-03		
L9	TRUE	POSTEA811_6	POSTA911	0	0,0139	0,139	0,000	3,30E-03		
L10A	TRUE	POSTE_BA	POSTB911	0	0,0363	0,363	0,000	1,98E-02		
L10C	TRUE	POSTE_CA	POSTC911	0	0,0605	0,605	0,000	3,30E-02		

LB1	TRUE	POSTEB811_1	POSTB911	0	0,0139	0,139	0,000	3,30E-03
LB2	TRUE	POSTEB811_2	POSTB911	0	0,0139	0,139	0,000	3,30E-03
LB3	TRUE	POSTEB811_3	POSTB911	0	0,0139	0,139	0,000	3,30E-03
LB4	TRUE	POSTEB811_4	POSTB911	0	0,0139	0,139	0,000	3,30E-03
L10BC	TRUE	POSTB911	POSTE BC	0	0,0303	0,303	0,000	1,65E-02
L10CB	TRUE	POSTC911	POSTE CB	0	0,0139	0,139	0,000	3,30E-03
LC1	TRUE	POSTEC811_1	POSTC911	0	0,0139	0,139	0,000	3,30E-03
LC2	TRUE	POSTEC811_2	POSTC911	0	0,0139	0,139	0,000	3,30E-03
LC3	TRUE	POSTEC811_3	POSTC911	0	0,0139	0,139	0,000	3,30E-03

cLoad

ID	ON	Bus k	Bus m	function	P (W)	Q (Var)	Vnom (V)
PQC1	TRUE	POSTEC811_1	GND	0	48 000 000,00	20 000 000,00	24 600,00
PQC2	TRUE	POSTEC811_2	GND	0	48 000 000,00	20 000 000,00	24 600,00
PQC3	TRUE	POSTEC811_3	GND	0	48 000 000,00	20 000 000,00	24 600,00
PQA811	TRUE	POSTEA811_1	GND	0	48 000 000,00	14 000 000,00	24 600,00
PQA812	TRUE	POSTEA811_2	GND	0	48 000 000,00	14 000 000,00	24 600,00
PQA813	TRUE	POSTEA811_3	GND	0	48 000 000,00	14 000 000,00	24 600,00
PQA814	TRUE	POSTEA811_4	GND	0	48 000 000,00	14 000 000,00	24 600,00
PQA815	TRUE	POSTEA811_5	GND	0	48 000 000,00	14 000 000,00	24 600,00
PQA816	TRUE	POSTEA811_6	GND	0	48 000 000,00	14 000 000,00	24 600,00
PQB811	TRUE	POSTEB811_1	GND	0	48 000 000,00	20 000 000,00	24 600,00
PQB812	TRUE	POSTEB811_2	GND	0	48 000 000,00	20 000 000,00	24 600,00
PQB813	TRUE	POSTEB811_3	GND	0	48 000 000,00	20 000 000,00	24 600,00
PQB814	TRUE	POSTEB811_4	GND	0	48 000 000,00	20 000 000,00	24 600,00
Comp2	TRUE	POSTEA811_1	GND	0	0,00	-1 100 000,00	24 600,00
Comp3	TRUE	POSTEA811_2	GND	0	0,00	-1 100 000,00	24 600,00
Comp4	TRUE	POSTEA811_3	GND	0	0,00	-1 100 000,00	24 600,00
Comp5	TRUE	POSTEA811_4	GND	0	0,00	-1 100 000,00	24 600,00
Comp6	TRUE	POSTEA811_5	GND	0	0,00	-1 100 000,00	24 600,00
Comp7	TRUE	POSTEA811_6	GND	0	0,00	-1 100 000,00	24 600,00
Comp10	TRUE	POSTEB811_1	GND	0	0,00	-11 100 000,00	24 600,00
Comp11	TRUE	POSTEB811_2	GND	0	0,00	-11 100 000,00	24 600,00
Comp12	TRUE	POSTEB811_3	GND	0	0,00	-11 100 000,00	24 600,00
Comp13	TRUE	POSTEB811_4	GND	0	0,00	-11 100 000,00	24 600,00
Comp15	TRUE	POSTEC811_1	GND	0	0,00	-6 000 000,00	24 600,00
Comp16	TRUE	POSTEC811_2	GND	0	0,00	-6 000 000,00	24 600,00
Comp17	TRUE	POSTEC811_3	GND	0	0,00	-6 000 000,00	24 600,00

L'exercice est également repris en variant les charges aux postes A et B afin de valider de nouveau la similitude des résultats entre l'outil APREM et le logiciel PSS/E. Les résultats sont présentés graphiquement à la Figure 3.65

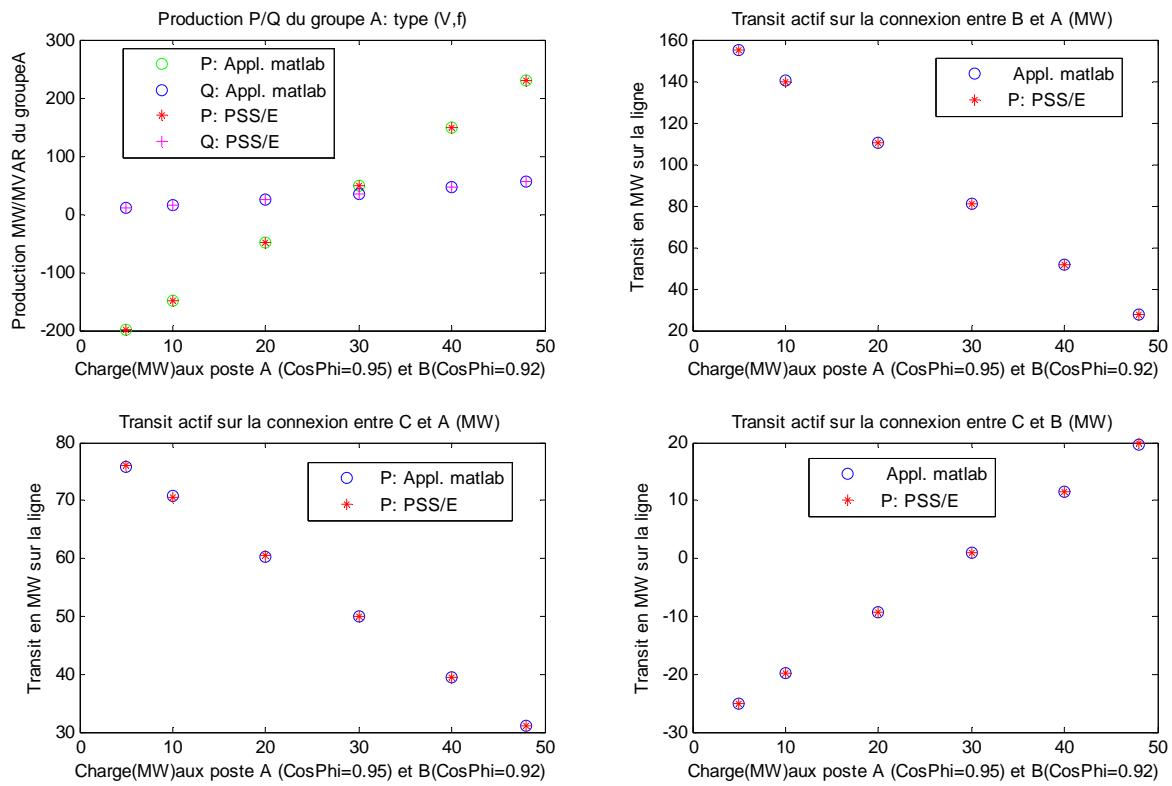


Figure 3.65 : résultats comparatifs entre l'outil APREM et PSS/E

CHAPITRE 4 CONLUSION ET RECOMMANDATIONS POUR TRAVAUX FUTURS

L'intérêt de ma recherche était principalement l'étude de la topologie et des paramètres des réseaux électriques. Ce mémoire a présenté une nouvelle approche simplifiée pour faire l'analyse paramétrique des réseaux électriques. L'utilisation du langage Orienté Objet sur Matlab a permis de développer un outil d'analyse avec lequel l'utilisateur a la possibilité de programmer des boucles de simulations de réseaux dans un langage de haut niveau. Il peut étudier l'évolution des réseaux électriques et aborder l'étude de la fiabilité des réseaux selon des méthodes probabilistes telles que Monte-Carlo. L'outil propose un environnement simple d'utilisation dans lequel l'usager peut développer ses propres scénarios et inclure des paramètres économiques, techniques, voir environnementales dans ces études.

L'outil APREM offre de grandes possibilités pour l'analyse des réseaux électriques. Les résultats sont très convaincants et le logiciel a prouvé sa justesse dès ces premiers essais et lors des comparaisons de résultats obtenus avec des logiciels commerciaux bien établis dans le domaine de l'analyse de réseau électrique.

L'analyse paramétrique des réseaux électriques sur Matlab propose d'étudier les réseaux électriques d'une nouvelle façon qui permet d'inclure des notions de fiabilité et de probabilité. L'approche visée cherche à fournir à l'usager un environnement de haut niveau dans lequel l'évolution des réseaux peut être programmée et ainsi réduire considérablement le nombre d'interventions directes de l'usager sur le logiciel durant l'exécution des différentes simulations.

L'outil APREM est la base d'une plateforme de développement. Parce que le logiciel utilise le langage de programmation Matlab, l'usager averti dispose de tous les moyens à sa disposition pour développer de nouvelles classes ou de nouvelles fonctions pour l'outil APREM et personnaliser chaque étude. De plus, le large déploiement du logiciel Matlab dans la communauté scientifique assurera le renouvellement de l'outil APREM et l'échange de données entre Matlab et tous les logiciels spécialisés compatibles. L'exportation des résultats vers Excel en est un exemple.

Plusieurs développements doivent néanmoins être envisagés, surtout dans l'optimisation des algorithmes de calculs. Bien que Matlab utilise un langage de haut niveau optimisé, le temps

de calculs du logiciel est non-négligeable et les performances des simulations sont ainsi très limitées. Le transfert de certaines opérations vers un langage plus performant n'est pas à rejeter.

Aussi, l'intégration d'une interface graphique pour la construction des réseaux est à étudier et il serait même intéressant, dans un premier temps, d'utiliser l'interface d'un logiciel existant à partir duquel les données seraient exportées vers l'outil APREM.

CHAPITRE 5 RÉFÉRENCES

- [1] CEATI, "Electric Distribution Utility Roadmap," 2004.
- [2] "Transforming the Grid to Revolutionize Electric Power in North America," United States Department of Energy, Office of Electric Transmission and Distribution2004.
- [3] G. Simard, *et al.*, "Hydro-Quebec's vision of the Distribution Automation," in *Transmission and Distribution Conference and Exhibition, 2005/2006 IEEE PES*, 2006, pp. 333-338.
- [4] "Plan d'évolution du réseau de distribution 2008-2023," Hydro-Québec, Distribution2008.
- [5] Available: http://www.hydroottawa.com/smartmeter/index.cfm?lang=e&template_id=326
- [6] R. N. Allan, *et al.*, "Bibliography on the application of probability methods in power system reliability evaluation," *IEEE Transactions on Power Systems*, vol. 14, pp. 51-57, 1999.
- [7] C. Bo, *et al.*, "Title," unpublished|.
- [8] R. Billinton and R. N. Allan, *Reliability Evaluation of Power Systems*, 2nd ed. New York: Plenum Press, 1996.
- [9] X. Luo, *et al.*, "Power system reliability evaluation using learning vector quantization and Monte Carlo simulation," *Electric Power Systems Research*, vol. 66, pp. 163-169, 2003.
- [10] H. Liu, *et al.*, "A novel state selection technique for power system reliability evaluation," *Electric Power Systems Research*, vol. 78, pp. 1019-1027, 2008.
- [11] R. Billinton and A. Sankarakrishnan, "Comparison of Monte Carlo simulation techniques for composite power system reliability assessment," in *Proceedings of the 1995 IEEE WESCANEX Communications, Power, and Computing Conference. Part 1 (of 2), May 15, 1995 - May 16, 1995*, Winnipeg, Manit, Can, 1995, pp. 145-150.
- [12] Y. Xingbin and S. Chanan, "Power system reliability analysis considering protection failures," in *Power Engineering Society Summer Meeting, 2002 IEEE*, 2002, pp. 963-968 vol.2.
- [13] I. Kurihara, *et al.*, "Development of power system reliability analysis program with consideration of system operation," *Electrical Engineering in Japan*, vol. 153, pp. 20-31, 2005.
- [14] W. Kersting, *Distribution systems modeling and analysis*. CA, USA: CRC Press LLC, 2002.
- [15] CYME International T&D. *CYMDIST Version 4.7 Reference Manual*. Available: <http://www.cyme.com>
- [16] J. A. Martinez and J. Mahseredjian, "Load Flow Calculations in Distribution Systems with Distributed Ressources, A Review.," in *Submitted to IEEE PES-2010 Conference*.
- [17] J. Mahseredjian, *et al.*, "On a new approach for the simulation of transients in power systems," *Electric Power Systems Research*, vol. 77, pp. 1514-20, 2007.

- [18] J. Mahseredjian, "Simulation des transitoires électromagnétiques dans les réseaux électriques," *Les Techniques de l'Ingénieur, Dossier D4130. 2008, Invited publication with funding*, p. 12, February 10, 2008.
- [19] J. Mahseredjian, "Comportement des réseaux électriques," *École polytechnique de Montréal, Notes du cours ELE8457 sur l'écoulement de puissance multiphasé*, 2010.
- [20] *PSS®E - Transmission System Analysis and Planning*. Available: <http://www.energy.siemens.com/hq/en/services/power-transmission-distribution/power-technologies-international/software-solutions/pss-e.htm>
- [21] *CYMDIST - Analyse de réseaux de distribution* Available: <http://www.cyme.com/fr/software/cymdist/>
- [22] *PSAF - Logiciel intégré d'analyse de réseaux électriques*. Available: <http://www.cyme.com/fr/software/psaf/>
- [23] *EMTP-RV - ElectroMagnetic Transients Program*. Available: <http://www.emtp.com/>
- [24] J. H. Chow and K. W. Cheung, "A toolbox for power system dynamics and control engineering education and research," *Power Systems, IEEE Transactions on*, vol. 7, pp. 1559-1564, 1992.
- [25] R. D. Zimmerman. (2010), *Matpower 4.0b5, User's Manual*. Available: <http://www.pserc.cornell.edu/matpower/>
- [26] S. Ayasun, *et al.*, "Voltage Stability Toolbox for Power System Education and Research," vol. 49, ed, 2006, pp. 442-442.
- [27] J. Mahseredjian and F. Alvarado, "Creating an Electromagnetic Transients Program in MATLAB: MatEMTP," vol. 12, ed, 1997, pp. 388-388.
- [28] (2010), *SimPowerSystems™ 5, User's Guide published under sublicense from Hydro-Québec, and The MathWorks, Inc.*. Available: <http://www.mathworks.com/products/simpower/>
- [29] K. Schoder, *et al.*, "PAT: A Power Analysis Toolbox for MATLAB/Simulink," vol. 18, ed, 2003, p. 42.
- [30] C. D. Vournas, *et al.*, "An Educational Simulation Tool for Power System Control and Stability," vol. 19, ed, 2004, pp. 48-55.
- [31] F. Milano, "An open source power system analysis toolbox," in *Power Engineering Society General Meeting, 2006. IEEE*, 2006, p. 1 pp.
- [32] J. Mahseredjian and F. Alvarado, "Creating an electromagnetic transients program in MATLAB: MatEMTP," *IEEE Transactions on Power Delivery*, vol. 12, pp. 380-388, 1997.
- [33] *MEX-files Guide*. Available: <http://www.mathworks.com/support/tech-notes/1600/1605.html>
- [34] "Parallel Computing Toolbox - Perform parallel computations on multicore computers, GPUs, and computer clusters."

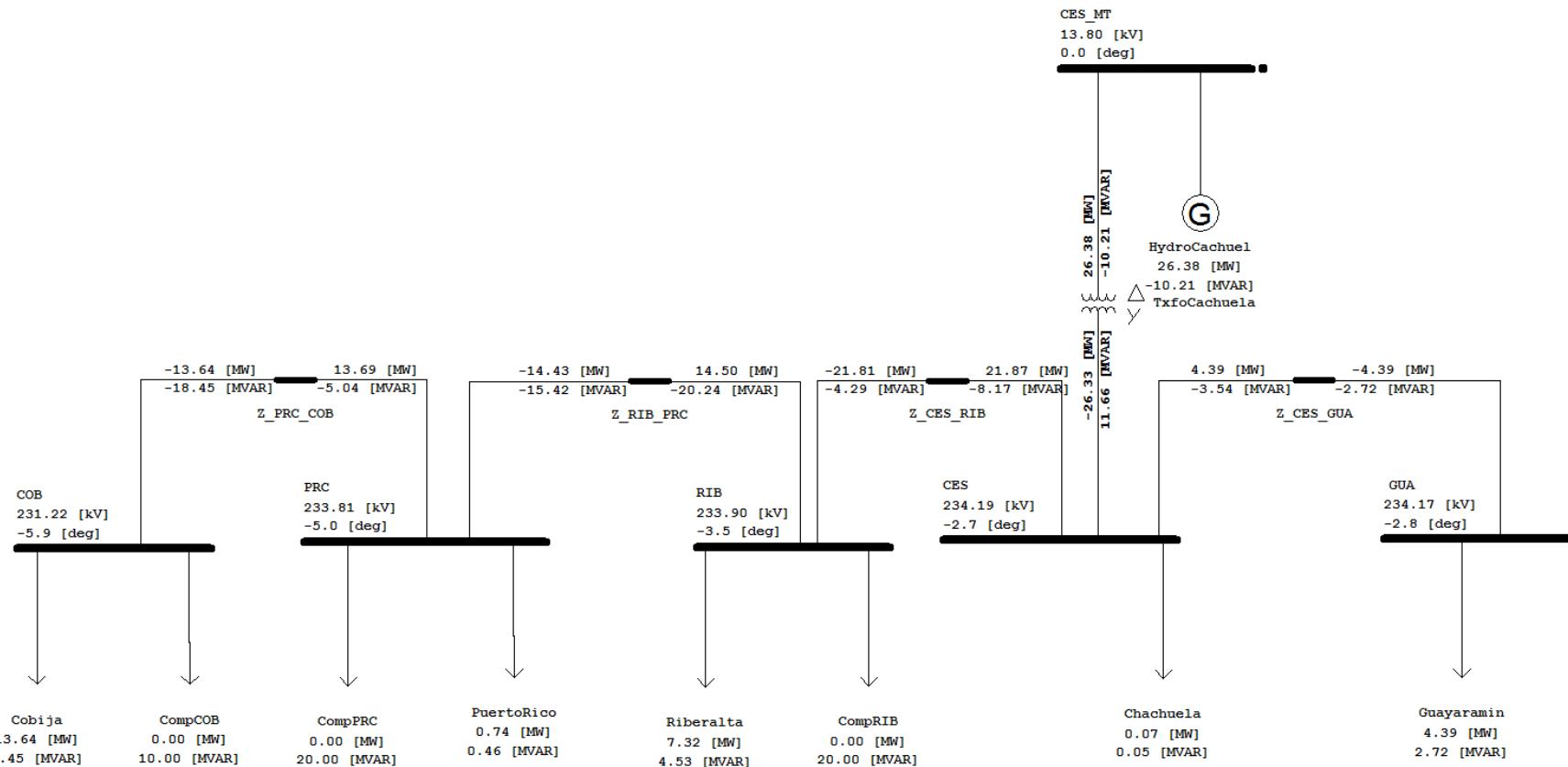
- [35] *MATLAB - Object-Oriented Programming in MATLAB*. Available: http://www.mathworks.com/products/matlab/object_oriented_programming.html
- [36] A. H. Register, *A guide to MATLAB object-oriented programming*. Boca Raton, Flo.: Chapman & Hall/CRC : SCITECH Pub., 2007.
- [37] *MATLAB Cell Arrays*. Available: http://www.mathworks.com/help/techdoc/matlab_prog/br04bw6-98.html
- [38] J. J. McConnell, *Analysis of algorithms : an active learning approach*. Sudbury, Mass.: Jones and Bartlett Publishers, 2001.
- [39] F. Chartrand, "Étude de fiabilité d'un réseau de distribution électrique avec une méthode probabiliste," Projet étudiant, cours ELE6412, École Polytechnique de Montréal, 2009.
- [40] M. Houari, "Validation des calculs d'écoulement de puissance de l'outil APREM," Projet étudiant, École Polytechnique de Montréal, 2009.

ANNEXE 1**Exemple I : données du réseau et résultats d'écoulement de puissance sur PSAF**

DONNÉES DU RÉSEAU DE L'EXEMPLE I SUR PSAF

Barre																
Nom équip.	État	kV Base	kV Opér.	Zone	Vmin [p.u.]	Vmax [p.u.]	RQQ1 [p.u.]	XQQ1 [p.u.]	Gén. P [MW]	Gén. Q [MVAR]	Angle [deg]	Q Max [MVAR]	Q Min [MVAR]	Barre contrôlée		
GUA	ON	230.000	230.000	0	0.900	1.100	0.00000	0.00000								
RIB	ON	230.000	230.000	0	0.900	1.100	0.00000	0.00000								
COB	ON	230.000	230.000	0	0.900	1.100	0.00000	0.00000								
PRC	ON	230.000	230.000	0	0.900	1.100	0.00000	0.00000								
CES	ON	230.000	230.000	0	0.900	1.100	0.00000	0.00000								
CES_MT	ON	13.800	13.800	0	0.900	1.100	0.00000	0.00000								
Génératrice																
Nom équip.	Nom BD	État	Duplic	De la barre	kV Nominal	MVA nominal [MVA]	Type gén.	Gén. P [MW]	Gén. Q [MVAR]	Angle [deg]	Q Max [MVAR]	Q Min [MVAR]	Barre contrôlée			
HydroCachuel	13.8_50	ON	1	CES_MT	13.800	60.0000	à barre d'équilibre	40.000	0.000	0.00	30.000	-15.000	CES_MT			
Charge Statique																
Nom équip.	Nom BD	État	Duplic	Duplic Info	De la barre	Charge P [MW]	Charge Q [MVAR]									
Chachuela	CONSTANT_S	ON	100	Données combinées	CES	0.0740	0.0459									
Cobija	CONSTANT_S	ON	100	Données combinées	COB	13.6370	8.4515									
Guayaramin	CONSTANT_S	ON	100	Données combinées	GUA	4.3900	2.7207									
PuertoRico	CONSTANT_S	ON	100	Données combinées	PRC	0.7410	0.4592									
Riberalta	CONSTANT_S	ON	100	Données combinées	RIB	7.3150	4.5334									
CompCOB	CONSTANT_S	ON	100	Données combinées	COB	0.0000	10.0000									
CompPRC	CONSTANT_S	ON	100	Données combinées	PRC	1.1.1 0.0000	1.1.2 20.0000									
CompRIB	CONSTANT_S	ON	100	Données combinées	RIB	0.0000	20.0000									
Ligne																
Nom équip.	Nom BD	État	Duplic	De la barre	A la barre	Niveau kV	Longueur	Unité longeur	R1 [ohm]	X1 [ohm]	B1 [uS]	Limite de charge normale [A]	Limite de charge d'urgence [A]			
Z_CES_GUA	230KV_795MCM	ON	1	CES	GUA	230.000	40.8300	[KM]	3.04151	16.77917	114.3240	900.00	1000.00			
Z_CES_RIB	230KV_795MCM	ON	1	CES	RIB	230.000	83.1700	[KM]	6.19550	34.17888	232.8760	900.00	1000.00			
Z_PRC_COB	230KV_795MCM	ON	1	PRC	COB	230.000	157.0300	[KM]	11.69748	64.53179	439.6840	900.00	1000.00			
Z_RIB_PRC	230KV_795MCM	ON	1	RIB	PRC	230.000	235.3800	[KM]	17.53393	96.72988	659.0640	900.00	1000.00			
Transformateur à prise fixe																
Nom équip.	Nom BD	État	Duplic	De la barre	A la barre	MVA nominal [MVA]	Primaire [kV]	Secondaire [kV]	Enroulement primaire	Enroulement secondaire	Déphasage	Z1 [ohm]	X/R Positive	Limite de charge normale [MVA]	Limite de charge d'urgence [MVA]	
TxfoCachuela	13.8/230_65	ON	1	CES_MT	CES	65.000	13.800	230.000	D	Y	30.00	0.3454	33.344	Shell type	65.000	75.000

RÉSULTATS DE L'ÉCOULEMENT DE PUISSANCE DU RÉSEAU DE L'EXEMPLE I SUR PSAF (ANNÉE 2015)



Barre														
Nom équip.	Zone	KV Base	V sol [kV]	Ang sol [deg]	Vmin [kV]	Vmax [kV]	Gén. P [MW]	Gén. Q [MVAR]	Charge P [MW]	Charge Q [MVAR]	P motor [MW]	Q motor [MVAR]	P shunt [MW]	Q shunt [MVAR]
CES	0	230.00	234.19	-2.7	207.00	253.00	0.00	0.00	0.07	0.05	0.00	0.00	0.00	0.00
CES_MT	0	13.80	13.80	0.0	12.42	15.18	26.38	-10.21	0.00	0.00	0.00	0.00	0.00	0.00
COB	0	230.00	231.22	-5.9	207.00	253.00	0.00	0.00	13.64	18.45	0.00	0.00	0.00	0.00
GUA	0	230.00	234.17	-2.8	207.00	253.00	0.00	0.00	4.39	2.72	0.00	0.00	0.00	0.00
PRC	0	230.00	233.81	-5.0	207.00	253.00	0.00	0.00	0.74	20.46	0.00	0.00	0.00	0.00
RIB	0	230.00	233.90	-3.5	207.00	253.00	0.00	0.00	7.32	24.53	0.00	0.00	0.00	0.00

Générateur														
Nom équip.	Nom de barre	Nom BD	Type	MVA nominal [MVA]	kV Nominal	Type gén.	P [MW]	Q [MVAR]	S [MVA]	Fact.puiss. [%]	I [pu]	Q max. [MVAR]	Q min. [MVAR]	Barre contrôlée/ V [kV]
HydroCachuel	CES_MT	13.8_50	Générateur	60.00	13.80	SW	26.38	-10.21	28.28	93.3	0.283	30.00	-15.00	CES_MT 13.80

Charge Statique														
Nom équip.	Nom de barre	Nom BD	P [MW]	Q [MVAR]	S [MVA]	Fact.puiss. [%]								
Cobija	COB	CONSTANT_S	13.64	8.45	16.04	85.0								
Chachuela	CES	CONSTANT_S	0.07	0.05	0.09	85.0								
Guayaramin	GUA	CONSTANT_S	4.39	2.72	5.16	85.0								
PuertoRico	PRC	CONSTANT_S	0.74	0.46	0.87	85.0								
CompRIB	RIB	CONSTANT_S	0.00	20.00	20.00	0.0								
CompCOB	COB	CONSTANT_S	0.00	10.00	10.00	0.0								
CompPRC	PRC	CONSTANT_S	0.00	20.00	20.00	0.0								
Riberalta	RIB	CONSTANT_S	7.32	4.53	8.61	85.0								

Ligne														
Nom équip.	De barre	Vers barre	Nom BD	Type	kV Nominal	Longueur	P [MW]	Q [MVAR]	S [MVA]	Fact.puiss. [%]	I [pu]	I angle [deg]	Pertes P [MW]	Pertes Q [MVAR]
Z_PRC_COB	PRC	COB	230KV_795MC M	Ligne	230.00	157.0300	13.69	-5.04	14.59	93.8	0.143	15.2	0.05	-23.49
Z_RIB_PRC	RIB	PRC	230KV_795MC M	Ligne	230.00	235.3800	14.50	-20.24	24.90	58.2	0.245	50.9	0.07	-35.66
Z_CES_GUA	CES	GUA	230KV_795MC M	Ligne	230.00	40.8300	4.39	-3.54	5.64	77.8	0.055	36.2	0.00	-6.26
Z_CES_RIB	CES	RIB	230KV_795MC M	Ligne	230.00	83.1700	21.87	-8.17	23.34	93.7	0.229	17.8	0.05	-12.46

Transformateur à prise fixe																	
Nom transfo	De barre	Vers barre	Nom BD	Type	MVA nominal [MVA]	kV Nominal Primaire	kV Nominal Secondaire	P [MW]	Q [MVAR]	S [MVA]	Fact.puiss. [%]	I [pu]	I angle [deg]	Pertes P [MW]	Pertes Q [MVAR]	Ratio prise %	POS PRISE [kV]
TxfoCachuela	CES_MT	CES	13.8/230_65	Transfo à prise fixe	65.00	13.80	230.00	26.38	-10.21	28.28	93.3	0.283	21.2	0.04	1.45	100	13.80

ANNEXE 2**Exemple I : code Matlab du réseau et résultats d'écoulement
de puissance sur APREM**

CODE APREM POUR CRÉER LE RÉSEAU DE L'EXEMPLE I

```

%% SOURCE
Network = struct;
Network.HydroCachuela = cVsource('CES_MT','0',13.8e3,0);

%% TRANSFORMER
Z1 = 0.1179;
r = 33.344;
R1 = sqrt(Z1^2/(1+r^2));
X1 = r*R1;
Vrated2 = 13.8e3; % V
Sbase2 = 65e6; % VA
Zbase2 = Vrated2^2/Sbase2; % Ohm
Z = R1+li*X1;
Z = Z*Zbase2;
Network.TxfoCachuela = cTxfoZ('CES_MT',0,'CES',0,13.8/230,0,Z);

%% LINES
R1 = 0.074492;
X1 = 0.410952;
B1 = 2.8000e-6;
Z = R1+li*X1;
Y = li*B1;
l = 40.83; Network.Z_CES_GUA = cLine('CES','GUA',Z*l,Y*l);
l = 83.17; Network.Z_CES_RIB = cLine('CES','RIB',Z*l,Y*l);
l = 235.38; Network.Z_RIB_PRC = cLine('RIB','PRC',Z*l,Y*l);
l = 157.03; Network.Z_PRC_COB = cLine('PRC','COB',Z*l,Y*l);

%% LOADS
Cobija = [13.637 20.456 20.967 21.492 22.029 ...
           22.580 23.144 23.723 24.316 24.924 25.547 ...
           26.185 26.840 27.511 28.199 28.904 29.626 ...
           30.367 31.126 31.904 32.702 33.520 34.358 ...
           35.217 36.097 36.999 37.924 38.872 39.844 ...
           40.840 41.861]*1e6;

Riberalta = [7.315 7.782 8.279 8.808 9.371 9.969 10.468 10.991 11.541 ...
             12.118 12.724 13.360 14.028 14.729 15.466 16.239 16.645 17.061 17.488 ...
             17.925 18.373 18.832 19.303 19.786 20.280 20.787 21.307 21.840 22.386 ...
             22.945 23.519]*1e6;

PuertoRico = [0.741 1.111 1.176 1.244 1.281 1.319 1.359 1.400 1.442 1.485 ...
              1.530 1.575 1.623 1.671 1.721 1.773 1.826 1.872 1.919 1.967 2.016 2.066 ...
              2.118 2.171 2.225 2.281 2.338 2.396 2.456 2.518 2.581]*1e6;

Cachuela = [0.074 0.078 0.083 0.088 0.092 0.097 0.102 0.107 0.112 0.118 ...
             0.123 0.130 0.136 0.143 0.150 0.157 0.165 0.174 0.182 0.191 0.201 ...
             0.211 0.222 0.233 0.244 0.257 0.269 0.283 0.297 0.312 0.327]*1e6;

Guayaramerin = [4.390 4.644 4.912 5.059 5.211 5.368 5.529 5.694 5.865 ...
                6.041 6.222 6.409 6.601 6.799 7.003 7.213 7.430 7.539 7.728 7.921 ...
                8.119 8.322 8.530 8.743 8.962 9.186 9.416 9.651 9.892 10.140 ...
                10.393]*1e6;

Cobija = Cobija + li*((Cobija/0.85).^2-Cobija.^2).^0.5;
Riberalta = Riberalta + li*((Riberalta/0.85).^2-Riberalta.^2).^0.5;
PuertoRico = PuertoRico + li*((PuertoRico/0.85).^2-PuertoRico.^2).^0.5;
Cachuela = Cachuela + li*((Cachuela/0.85).^2-Cachuela.^2).^0.5;
Guayaramerin = Guayaramerin + li*((Guayaramerin/0.85).^2-Guayaramerin.^2).^0.5;
Cobija = Cobija.';
Riberalta = Riberalta.';
PuertoRico = PuertoRico.';
Cachuela = Cachuela.';
Guayaramerin = Guayaramerin.';
Network.Cachuela = cLoad('CES','0',real(Cachuela(1)),imag(Cachuela(1)),Vrated);
Network.Guayaramerin = cLoad('GUA','0',real(Guayaramerin(1)),imag(Guayaramerin(1)),Vrated);
Network.Riberalta = cLoad('RIB','0',real(Riberalta(1)),imag(Riberalta(1)),Vrated);
Network.PuertoRico = cLoad('PRC','0',real(PuertoRico(1)),imag(PuertoRico(1)),Vrated);
Network.Cobija = cLoad('COB','0',real(Cobija(1)),imag(Cobija(1)),Vrated);

```

RÉSULTATS DE L'ÉCOULEMENT DE PUISSANCE DU RÉSEAU DE L'EXEMPLE I SUR APREM (ANNÉE 2015)

18-Jan-2011 20:56:26 y2015

Bus Data												
Voltage :												
Bus ID	Mag(V)	Ang(deg)										
GND	0.000	0.000										
CES_MT	13800.000	0.000										
CES	234193.962	-2.723										
GUA	234166.370	-2.801										
RIB	233897.896	-3.516										
PRC	233809.009	-5.026										
COB	231224.174	-5.876										
Branch Data												
Branch ID	From Bus k	To Bus m	Voltage: Mag(V)	Ang(deg)	Current: Mag(A)	Ang(deg)	Power: P(W)	Q(VAr)	Bus k: P(W)	Q(VAr)	Bus m : P(W)	Q(VAr)
HydroCachuela	1	CES_MT	GND	1.380e+004	0.000	1.183e+003	-158.830	-2.64e+007	1.02e+007	-2.64e+007	1.02e+007	0.00e+000
TxfoCachuela	1	CES_MT	GND	1.380e+004	0.000	1.183e+003	21.170	2.64e+007	-1.02e+007	2.64e+007	-1.02e+007	0.00e+000
	1	CES	GND	2.342e+005	-2.723	7.100e+001	-158.830	-2.63e+007	1.17e+007	-2.63e+007	1.17e+007	0.00e+000
Z_CES_GUA	1	CES	GUA	3.211e+002	82.308	1.391e+001	36.175	1.08e+003	-6.26e+006	4.39e+006	-3.54e+006	-4.39e+006
Z_CES_RIB	1	CES	RIB	3.254e+003	81.660	5.755e+001	17.759	5.44e+004	-1.25e+007	2.19e+007	-8.17e+006	-2.18e+007
Z_RIB_PRC	1	RIB	PRC	6.164e+003	84.902	6.146e+001	50.877	6.89e+004	-3.57e+007	1.45e+007	-2.02e+007	-1.44e+007
Z_PRC_COB	1	PRC	COB	4.309e+003	47.692	3.602e+001	15.195	5.05e+004	-2.35e+007	1.37e+007	-5.04e+006	-1.36e+007
Cachuela	1	CES	GND	2.342e+005	-2.723	2.146e-001	-34.511	7.40e+004	4.59e+004	7.40e+004	4.59e+004	0.00e+000
Guayaramerin	1	GUA	GND	2.342e+005	-2.801	1.273e+001	-34.590	4.39e+006	2.72e+006	4.39e+006	2.72e+006	0.00e+000
Riberalta	1	RIB	GND	2.339e+005	-3.516	2.124e+001	-35.305	7.31e+006	4.53e+006	7.31e+006	4.53e+006	0.00e+000
PuertoRico	1	PRC	GND	2.338e+005	-5.026	2.153e+000	-36.815	7.41e+005	4.59e+005	7.41e+005	4.59e+005	0.00e+000
Cobija	1	COB	GND	2.312e+005	-5.876	4.006e+001	-37.664	1.36e+007	8.45e+006	1.36e+007	8.45e+006	0.00e+000
CompGUA	1	GUA	GND	2.342e+005	-2.801	0.000e+000	0.000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000
CompRIB	1	RIB	GND	2.339e+005	-3.516	4.937e+001	-93.516	-2.33e-010	2.00e+007	-2.33e-010	2.00e+007	0.00e+000
CompPRC	1	PRC	GND	2.338e+005	-5.026	4.939e+001	-95.026	0.00e+000	2.00e+007	0.00e+000	2.00e+007	0.00e+000
CompCOB	1	COB	GND	2.312e+005	-5.876	2.497e+001	-95.876	0.00e+000	1.00e+007	0.00e+000	1.00e+007	0.00e+000
TOTAL:												
							-4.121e-008	-2.38e-007				

ANNEXE 3**Exemple II : code Matlab du réseau et résultats
d'écoulement de puissance sur APREM**

CODE APREM POUR CRÉER LE RÉSEAU DE L'EXEMPLE II

```

%% FICHIER ELE6412.m
%% Build the Network Structure
addpath('Ossoucah_2009-08-14');
addpath('Ossoucah_2009-08-14\lib');
initialisation;

Network = [];

Vbase = 26.4e3;
Sbase = 25e6;
Zbase = (Vbase^2)/Sbase;
Charges = [4.7 3.3 8.2 4.8 4.6 3.6 5.6 4.5 5.8 3.2 5.5 6.2]*1e6;

Network.LA = cImpedance('Sa','Ca', 0.01*Zbase,0.01*Zbase);
Network.LB = cImpedance('Sb','Cb', 0.01*Zbase,0.01*Zbase);
Network.LC = cImpedance('Sc','Cc', 0.01*Zbase,0.01*Zbase);
Network.LD = cImpedance('Sd','Cd', 0.01*Zbase,0.01*Zbase);
Network.LE = cImpedance('Se','Ce', 0.01*Zbase,0.01*Zbase);
Network.LF = cImpedance('Sf','Cf', 0.01*Zbase,0.01*Zbase);
Network.LG = cImpedance('Sg','Cg', 0.01*Zbase,0.01*Zbase);
Network.LH = cImpedance('Sh','Ch', 0.01*Zbase,0.01*Zbase);
Network.LI = cImpedance('Si','Ci', 0.01*Zbase,0.01*Zbase);
Network.LJ = cImpedance('Sj','Cj', 0.01*Zbase,0.01*Zbase);
Network.LK = cImpedance('Sk','Ck', 0.01*Zbase,0.01*Zbase);
Network.LM = cImpedance('Sm','Cm', 0.01*Zbase,0.01*Zbase);
Network.L2 = cImpedance('S2','Sx', 0.01*Zbase,0.01*Zbase);
Network.Ll = cImpedance('S1','Sy', 0.01*Zbase,0.01*Zbase);
Network.L3 = cImpedance('S3','S4', 0.01*Zbase,0.01*Zbase);

Network.CA = cLoad('Ca','0',Charges(1),0,Vbase);
Network.CB = cLoad('Cb','0',Charges(2),0,Vbase);
Network.CC = cLoad('Cc','0',Charges(3),0,Vbase);
Network.CD = cLoad('Cd','0',Charges(4),0,Vbase);
Network.CE = cLoad('Ce','0',Charges(5),0,Vbase);
Network.CF = cLoad('Cf','0',Charges(6),0,Vbase);
Network.CG = cLoad('Cg','0',Charges(7),0,Vbase);
Network.CH = cLoad('Ch','0',Charges(8),0,Vbase);
Network.CI = cLoad('Ci','0',Charges(9),0,Vbase);
Network.CJ = cLoad('Cj','0',Charges(10),0,Vbase);
Network.CK = cLoad('Ck','0',Charges(11),0,Vbase);
Network.CM = cLoad('Cm','0',Charges(12),0,Vbase);

Network.SA = cSwitch('B0','Sa',1);
Network.SB = cSwitch('Ca','Sb',1);
Network.SC = cSwitch('Cb','Sc',1);
Network.SD = cSwitch('Cc','Sd',1);
Network.SE = cSwitch('Cf','Se',1);
Network.SF = cSwitch('B0','Sf',1);
Network.SG = cSwitch('Ch','Sg',1);
Network.SH = cSwitch('Ce','Sh',1);
Network.SI = cSwitch('Cj','Si',1);
Network.SJ = cSwitch('Ck','Sj',1);
Network.SK = cSwitch('Cm','Sk',1);
Network.SM = cSwitch('B0','Sm',1);
Network.S1 = cSwitch('Cd','S1',0);
Network.S2 = cSwitch('Cg','S2',0);
Network.S3 = cSwitch('Ci','S3',0);
Network.S4 = cSwitch('B0','S4',0);
Network.SX = cSwitch('Sx','S3',0);
Network.SY = cSwitch('Sy','S2',0);

Network.V0 = cVsource('B0','0',Vbase,0);

fLoadFlow(Network);

Network.LA.function = 'FOR01';
Network.LB.function = 'FOR01';
Network.LC.function = 'FOR01';

```

```
Network.LD.function = 'FOR01';
Network.LE.function = 'FOR01';
Network.LF.function = 'FOR01';
Network.LG.function = 'FOR01';
Network.LH.function = 'FOR01';
Network.LI.function = 'FOR01';
Network.LJ.function = 'FOR01';
Network.LK.function = 'FOR01';
Network.LM.function = 'FOR01';
Network.L2.function = 'FOR01';
Network.L1.function = 'FOR01';
Network.L3.function = 'FOR01';

Network.SA.function = 'FOR005';
Network.SB.function = 'FOR005';
Network.SC.function = 'FOR005';
Network.SD.function = 'FOR005';
Network.SE.function = 'FOR005';
Network.SF.function = 'FOR005';
Network.SG.function = 'FOR005';
Network.SH.function = 'FOR005';
Network.SI.function = 'FOR005';
Network.SJ.function = 'FOR005';
Network.SK.function = 'FOR005';
Network.SM.function = 'FOR005';
Network.SX.function = 'FOR005';
Network.SY.function = 'FOR005';
Network.S1.function = 'FOR005';
Network.S2.function = 'FOR005';
Network.S3.function = 'FOR005';
Network.S4.function = 'FOR005';

fSave(Network);
fReport(fSave,'ELE6412.txt');
open ELE6412.txt
fReportxls(fSave,'ELE6412_2.xls');
```

```

%% Fichier FOR01.m
%% Fonction de probabilité FOR = 0.01
%% Tirage au sort pour retirer l'équipement du réseau
function varargout = FOR01(this,varargin)

switch nargin
    case{1}
        firstIteration = false;
        Network = [];
    case{2}
        firstIteration = varargin{1};
        Network = [];
    case{3}
        firstIteration = varargin{1};
        Network = varargin{2};
end

if firstIteration
    ON = rand;
    if ON < 0.01
        this.ON = false;
    else
        this.ON = true;
    end
end
doLoadFlowAgain = false;

varargout{1} = this;
varargout{2} = doLoadFlowAgain;

%% Fichier FOR005.m
%% Fonction de probabilité FOR = 0.005
%% Tirage au sort pour retirer l'équipement du réseau
function varargout = FOR005(this,varargin)

switch nargin
    case{1}
        firstIteration = false;
        Network = [];
    case{2}
        firstIteration = varargin{1};
        Network = [];
    case{3}
        firstIteration = varargin{1};
        Network = varargin{2};
end

if firstIteration
    ON = rand;
    if ON < 0.005
        this.ON = false;
    else
        this.ON = true;
    end
end
doLoadFlowAgain = false;

varargout{1} = this;
varargout{2} = doLoadFlowAgain;

```


CA	1	Ca	GND	2.617e+004	-0.485	1.796e+002	-0.485	4.70e+006	0.00e+000	4.70e+006	0.00e+000	0.00e+000	0.00e+000
CB	1	Cb	GND	2.599e+004	-0.868	1.270e+002	-0.868	3.30e+006	0.00e+000	3.30e+006	0.00e+000	0.00e+000	0.00e+000
CC	1	Cc	GND	2.585e+004	-1.177	3.172e+002	-1.177	8.20e+006	2.91e-011	8.20e+006	2.91e-011	0.00e+000	0.00e+000
CD	1	Cd	GND	2.580e+004	-1.292	1.860e+002	-1.292	4.80e+006	0.00e+000	4.80e+006	0.00e+000	0.00e+000	0.00e+000
CE	1	Ce	GND	2.604e+004	-0.767	1.766e+002	-0.767	4.60e+006	0.00e+000	4.60e+006	0.00e+000	0.00e+000	0.00e+000
CF	1	Cf	GND	2.620e+004	-0.423	1.374e+002	-0.423	3.60e+006	-3.64e-012	3.60e+006	-3.64e-012	0.00e+000	0.00e+000
CG	1	Cg	GND	2.587e+004	-1.139	2.164e+002	-1.139	5.60e+006	1.46e-011	5.60e+006	1.46e-011	0.00e+000	0.00e+000
CH	1	Ch	GND	2.593e+004	-1.006	1.735e+002	-1.006	4.50e+006	0.00e+000	4.50e+006	0.00e+000	0.00e+000	0.00e+000
CI	1	Ci	GND	2.586e+004	-1.170	2.243e+002	-1.170	5.80e+006	0.00e+000	5.80e+006	0.00e+000	0.00e+000	0.00e+000
CJ	1	Cj	GND	2.592e+004	-1.032	1.234e+002	-1.032	3.20e+006	0.00e+000	3.20e+006	0.00e+000	0.00e+000	0.00e+000
CK	1	Ck	GND	2.602e+004	-0.819	2.114e+002	-0.819	5.50e+006	0.00e+000	5.50e+006	0.00e+000	0.00e+000	0.00e+000
CM	1	Cm	GND	2.618e+004	-0.478	2.369e+002	-0.478	6.20e+006	-7.28e-012	6.20e+006	-7.28e-012	0.00e+000	0.00e+000
SA	1	B0	Sa	0.000e+000	0.000	8.097e+002	-1.002	0.00e+000	0.00e+000	2.14e+007	3.74e+005	2.14e+007	3.74e+005
SB	1	Ca	Sb	0.000e+000	0.000	6.302e+002	-1.149	0.00e+000	0.00e+000	1.65e+007	1.91e+005	1.65e+007	1.91e+005
SC	1	Cb	Sc	0.000e+000	0.000	5.032e+002	-1.220	0.00e+000	0.00e+000	1.31e+007	8.02e+004	1.31e+007	8.02e+004
SD	1	Cc	Sd	0.000e+000	0.000	1.860e+002	-1.292	0.00e+000	0.00e+000	4.81e+006	9.65e+003	4.81e+006	9.65e+003
SE	1	Cf	Se	0.000e+000	0.000	5.666e+002	-0.982	0.00e+000	0.00e+000	1.48e+007	1.45e+005	1.48e+007	1.45e+005
SF	1	B0	Sf	2.626e-031	90.000	7.040e+002	-0.873	-2.82e-030	1.85e-028	1.86e+007	2.83e+005	1.86e+007	2.83e+005
SG	1	Ch	Sg	0.000e+000	0.000	2.164e+002	-1.139	0.00e+000	0.00e+000	5.61e+006	1.31e+004	5.61e+006	1.31e+004
SH	1	Ce	Sh	0.000e+000	0.000	3.900e+002	-1.080	0.00e+000	0.00e+000	1.02e+007	5.55e+004	1.02e+007	5.55e+004
SI	1	Cj	Si	0.000e+000	0.000	2.243e+002	-1.170	0.00e+000	0.00e+000	5.81e+006	1.40e+004	5.81e+006	1.40e+004
SJ	1	Ck	Sj	0.000e+000	0.000	3.477e+002	-1.121	0.00e+000	0.00e+000	9.05e+006	4.77e+004	9.05e+006	4.77e+004
SK	1	Cm	Sk	0.000e+000	0.000	5.591e+002	-1.007	0.00e+000	0.00e+000	1.46e+007	1.35e+005	1.46e+007	1.35e+005
SM	1	B0	Sm	1.756e-030	90.000	7.960e+002	-0.849	-2.07e-029	1.40e-027	2.10e+007	3.12e+005	2.10e+007	3.12e+005
S1	1	Cd	S1	2.580e+004	-1.292	0.000e+000	0.000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000
S2	1	Cg	S2	2.587e+004	-1.139	0.000e+000	0.000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000
S3	1	Ci	S3	2.586e+004	-1.170	0.000e+000	0.000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000
S4	1	B0	S4	2.640e+004	0.000	0.000e+000	0.000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000
SX	1	Sx	S3	0.000e+000	0.000	0.000e+000	0.000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000
SY	1	Sy	S2	0.000e+000	0.000	0.000e+000	0.000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000	0.00e+000
V0	1	B0	GND	2.640e+004	0.000	2.310e+003	179.090	-6.10e+007	-9.68e+005	-6.10e+007	-9.68e+005	0.00e+000	0.00e+000

TOTAL: 0.000e+000 4.66e-010

ANNEXE 4**Exemple III : données du réseau et résultats d'écoulement de puissance sur PSS/E**

DONNÉES DU RÉSEAU DE L'EXEMPLE III SUR PSS/E

BUS									
Bus Number	Bus Name	Base kV	Voltage (pu)	Angle (deg)					
1	POSTEA	120	1	0					
2	POSTEA811_1	24,6	0,9998	-0,02					
3	POSTEA811_2	24,6	0,9998	-0,02					
4	POSTEA811_3	24,6	0,9998	-0,02					
5	POSTEA811_4	24,6	0,9998	-0,02					
6	POSTEA811_5	24,6	0,9998	-0,02					
7	POSTEA811_6	24,6	0,9998	-0,02					
8	POSTEA911	24,6	1,0006	0,1					
9	POSTEB	120	1	1,16					
10	POSTEB811_1	24,6	0,9999	1,13					
11	POSTEB811_2	24,6	0,9999	1,13					
12	POSTEB811_3	24,6	0,9999	1,13					
13	POSTEB811_4	24,6	0,9999	1,13					
14	POSTEC	120	1	2,2					
15	POSTEC811_1	24,6	0,9998	2,17					
16	POSTEC811_2	24,6	0,9998	2,17					
17	POSTEC811_3	24,6	0,9998	2,17					
18	POSTEC911	24,6	1,0011	1,93					
19	POSTEB911	24,6	1,0008	1,1					
20	POST_AB	24,6	1,0007	0,12					
21	POST_BA	24,6	1,0008	0,14					
22	POST_AC	24,6	1,0008	0,12					
23	POST_CA	24,6	1,0009	0,14					
24	POST_BC	24,6	1,0019	1,66					
25	POST_CB	24,6	1,0019	1,67					

MACHINE									
Bus Number	Bus Name	Code	VSched (pu)	Pgen (MW)	Pmax (MW)	Pmin (MW)	Qgen (Mvar)	Qmax (Mvar)	Qmin (Mvar)
								Mbase (MVA)	R Source (pu)
1	POSTEA	120,00	3	1	229,288	280	0	56,0399	300
9	POSTEB	120,00	2	1	200	230	0	14,0417	120
14	POSTEC	120,00	2	1	195	230	0	17,318	100

LOAD									
Bus Number	Bus Name	Pload (MW)	Qload (Mvar)	IPload (MW)	IQload (Mvar)	YPload (MW)	YQload (Mvar)		
2	POSTEA811_1	24,600	48	14	0	0	0	0	0
3	POSTEA811_2	24,600	48	14	0	0	0	0	0
4	POSTEA811_324,600	48	14	0	0	0	0	0	0
5	POSTEA811_424,600	48	14	0	0	0	0	0	0
6	POSTEA811_524,600	48	14	0	0	0	0	0	0
7	POSTEA811_624,600	48	14	0	0	0	0	0	0
10	POSTEB811_124,600	48	20	0	0	0	0	0	0
11	POSTEB811_224,600	48	20	0	0	0	0	0	0
12	POSTEB811_324,600	48	20	0	0	0	0	0	0
13	POSTEB811_424,600	48	20	0	0	0	0	0	0
15	POSTEC811_124,600	48	20	0	0	0	0	0	0
16	POSTEC811_224,600	48	20	0	0	0	0	0	0
17	POSTEC811_324,600	48	20	0	0	0	0	0	0

FIXED SHUNT									
Bus Number	Bus Name	G-Shunt (MW)	B-Shunt (Mvar)						
2	POSTEA811_1	24,600	0		1,1				
3	POSTEA811_2	24,600	0		1,1				
4	POSTEA811_324,600	0		1,1					
5	POSTEA811_424,600	0		1,1					

6	POSTEA811_524,600	0	1,1
7	POSTEA811_624,600	0	1,1
10	POSTEB811_124,600	0	11,1
11	POSTEB811_224,600	0	11,1
12	POSTEB811_324,600	0	11,1
13	POSTEB811_424,600	0	11,1
15	POSTEC811_124,600	0	6
16	POSTEC811_224,600	0	6
17	POSTEC811_324,600	0	6

BRANCH

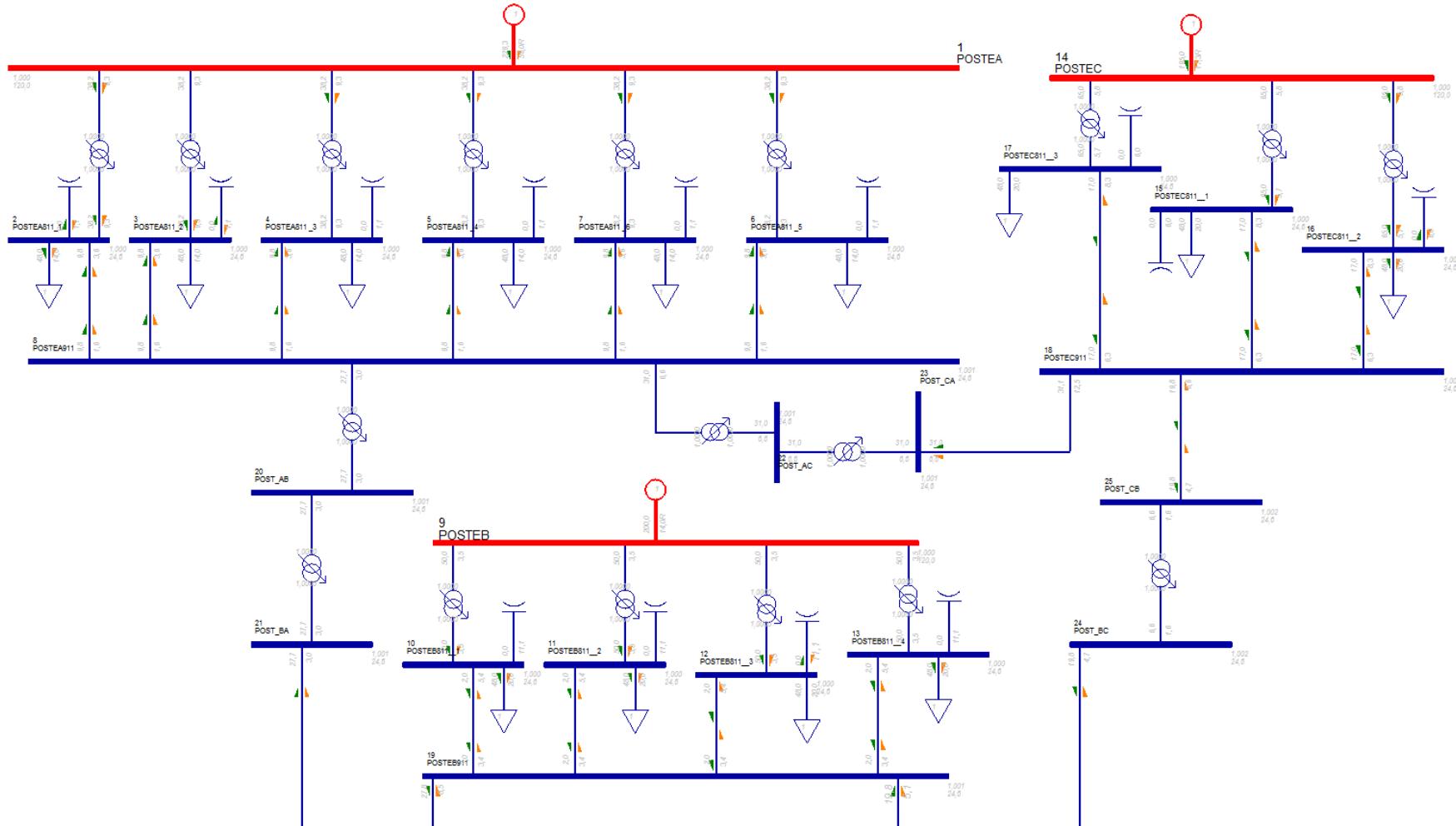
From Bus Number	From Bus Name	To Bus Number	To Bus Name	Id	Line R (pu)	Line X (pu)	Charging (pu)
2	POSTEA811_124,600	8	POSTEA911_24,600	1	0,0023	0,023	0,02
3	POSTEA811_224,600	8	POSTEA911_24,600	2	0,0023	0,023	0,02
4	POSTEA811_324,600	8	POSTEA911_24,600	3	0,0023	0,023	0,02
5	POSTEA811_424,600	8	POSTEA911_24,600	2	0,0023	0,023	0,02
6	POSTEA811_524,600	8	POSTEA911_24,600	2	0,0023	0,023	0,02
7	POSTEA811_624,600	8	POSTEA911_24,600	2	0,0023	0,023	0,02
10	POSTEB811_124,600	19	POSTEB911_24,600	1	0,0023	0,023	0,02
11	POSTEB811_224,600	19	POSTEB911_24,600	2	0,0023	0,023	0,02
12	POSTEB811_324,600	19	POSTEB911_24,600	2	0,0023	0,023	0,02
13	POSTEB811_424,600	19	POSTEB911_24,600	2	0,0023	0,023	0,02
15	POSTEC811_124,600	18	POSTEC911_24,600	2	0,0023	0,023	0,02
16	POSTEC811_224,600	18	POSTEC911_24,600	2	0,0023	0,023	0,02
17	POSTEC811_324,600	18	POSTEC911_24,600	2	0,0023	0,023	0,02
18	POSTEC911_24,600	23	POST_CA_24,600	2	0,01	0,1	0,2
18	POSTEC911_24,600	25	POST_CB_24,600	1	0,0023	0,023	0,02
19	POSTEB911_24,600	21	POST_BA_24,600	2	0,006	0,06	0,12
19	POSTEB911_24,600	24	POST_BB_24,600	2	0,005	0,05	0,1

2 WINDING

From Bus Number	From Bus Name	To Bus Number	To Bus Name	Winding I/O Code	Impedance I/O Code	Admittance I/O Code	Specified R (pu or watts)	Specified X (pu)	Winding MVA Base	Wnd 1 Ratio (pu or kV)	Wnd 1 Nominal kV	Wnd 1 Angle (degrees)	Wnd 2 Ratio (pu or kV)	Wnd 2 Nominal kV
1	POSTEA 120,00	2	POSTEA811_124,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120
1	POSTEA 120,00	3	POSTEA811_224,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120
1	POSTEA 120,00	4	POSTEA811_324,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120
1	POSTEA 120,00	5	POSTEA811_424,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120
1	POSTEA 120,00	6	POSTEA811_524,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120
1	POSTEA 120,00	7	POSTEA811_624,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120
8	POSTEA911_24,600	20	POST_AB_24,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	24,6
8	POSTEA911_24,600	22	POST_AC_24,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	24,6
9	POSTEB 120,00	10	POSTEB811_124,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120

9	POSTEB	120,00		11	POSTEB811_224,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120				
9	POSTEB	120,00		12	POSTEB811_324,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120				
9	POSTEB	120,00		13	POSTEB811_424,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120				
14	POSTEC	120,00		15	POSTEC811_124,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	5.1.1	24,6	5.1.2	0	5.1.3	1	5.1.4	120
14	POSTEC	120,00		16	POSTEC811_224,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120				
14	POSTEC	120,00		17	POSTEC811_324,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	120				
20	POST_AB	24,600		21	POST_BA 24,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	24,6				
22	POST_AC	24,600		23	POST_CA 24,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	24,6				
24	POST_BC	24,600		25	POST_CB 24,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	24,6				
24	POST_BC	24,600		25	POST_CB 24,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	24,6				
24	POST_BC	24,600		25	POST_CB 24,600	Turns ratio (pu on bus base kV)	Zpu (system base)	Y pu (system base)	0,00021	0,001	100	1	24,6	0	1	24,6				

RÉSULTATS DE L'ÉCOULEMENT DE PUISSANCE DU RÉSEAU DE L'EXEMPLE III SUR PSS/E



PTI INTERACTIVE POWER SYSTEM SIMULATOR--PSS®E										MON, JAN 17 2011 21:55							
X----- FROM BUS -----X AREA VOLT				GEN	LOAD	SHUNT	X----- TO BUS -----X	TRANSFORMER						RATING			
BUS#	X-- NAME --X BASKV	ZONE	PU/KV	ANGLE	MW/MVAR	MW/MVAR	MW/MVAR	BUS#	X-- NAME --X BASKV	AREA	CKT	MW	MVAR	RATIO	ANGLE	AMPS	% SET A
1 POSTEA	120.00	1	1.0000	0.0	229.3	0.0	0.0	-	-	-	-	-	-	-	-	-	
		1	120.00		56.0R	0.0	0.0	2	POSTEA811_1	24.600	1	1	38.2	9.3 1.000UN	189	82	48M
								3	POSTEA811_2	24.600	1	2	38.2	9.3 1.000UN	189	82	48M
								4	POSTEA811_3	24.600	1	3	38.2	9.3 1.000UN	189	82	48M
								5	POSTEA811_4	24.600	1	4	38.2	9.3 1.000UN	189	82	48M
								6	POSTEA811_5	24.600	1	5	38.2	9.3 1.000UN	189	82	48M
								7	POSTEA811_6	24.600	1	6	38.2	9.3 1.000UN	189	82	48M
2 POSTEA811_1	24.600	1	0.9998	0.0	0.0	48.0	0.0	-	-	-	-	-	-	-	-	-	
		1	24.596		0.0	14.0	-1.1	1	POSTEA	120.00	1	1	-38.2	-9.3 1.000HI	923	82	48M
								8	POSTEA911	24.600	1	1	-9.8	-3.6	245	26	939A
3 POSTEA811_2	24.600	1	0.9998	0.0	0.0	48.0	0.0	-	-	-	-	-	-	-	-	-	
		1	24.596		0.0	14.0	-1.1	1	POSTEA	120.00	1	2	-38.2	-9.3 1.000HI	923	82	48M
								8	POSTEA911	24.600	1	2	-9.8	-3.6	245	26	939A
4 POSTEA811_3	24.600	1	0.9998	0.0	0.0	48.0	0.0	-	-	-	-	-	-	-	-	-	
		1	24.596		0.0	14.0	-1.1	1	POSTEA	120.00	1	3	-38.2	-9.3 1.000HI	923	82	48M
								8	POSTEA911	24.600	1	3	-9.8	-3.6	245	26	939A
5 POSTEA811_4	24.600	1	0.9998	0.0	0.0	48.0	0.0	-	-	-	-	-	-	-	-	-	
		1	24.596		0.0	14.0	-1.1	1	POSTEA	120.00	1	4	-38.2	-9.3 1.000HI	923	82	48M
								8	POSTEA911	24.600	1	2	-9.8	-3.6	245	26	939A
6 POSTEA811_5	24.600	1	0.9998	0.0	0.0	48.0	0.0	-	-	-	-	-	-	-	-	-	
		1	24.596		0.0	14.0	-1.1	1	POSTEA	120.00	1	5	-38.2	-9.3 1.000HI	923	82	48M
								8	POSTEA911	24.600	1	2	-9.8	-3.6	245	26	939A
7 POSTEA811_6	24.600	1	0.9998	0.0	0.0	48.0	0.0	-	-	-	-	-	-	-	-	-	
		1	24.596		0.0	14.0	-1.1	1	POSTEA	120.00	1	6	-38.2	-9.3 1.000HI	923	82	48M
								8	POSTEA911	24.600	1	2	-9.8	-3.6	245	26	939A
8 POSTEA911	24.600	1	1.0006	0.1	0.0	0.0	0.0	-	-	-	-	-	-	-	-	-	
		1	24.616		0.0	0.0	0.0	2	POSTEA811_1	24.600	1	1	9.8	1.6	233	25	939A

							3 POSTEA811_2 24. 600	1 2	9. 8	1. 6		233	25	939A	
							4 POSTEA811 _324. 600	1 3	9. 8	1. 6		233	25	939A	
							5 POSTEA811 _424. 600	1 2	9. 8	1. 6		233	25	939A	
							6 POSTEA811 _524. 600	1 2	9. 8	1. 6		233	25	939A	
							7 POSTEA811 _624. 600	1 2	9. 8	1. 6		233	25	939A	
							20 POST_AB 24. 600	1 7	-27. 7	-3. 0 1. 000UN		654	42	67M	
							22 POST_AC 24. 600	1 8	-31. 0	-6. 6 1. 000UN		744	47	67M	
9 POSTEB	120. 00	1 1. 0000	1. 2	200. 0	0. 0	0. 0									
		1 120. 00		14. OR	0. 0	0. 0	10 POSTEB811__124. 600	1 9	50. 0	3. 5 1. 000UN		241	75	67M	
							11 POSTEB811__224. 600	1 10	50. 0	3. 5 1. 000UN		241	75	67M	
							12 POSTEB811__324. 600	1 12	50. 0	3. 5 1. 000UN		241	75	67M	
							13 POSTEB811__424. 600	1 13	50. 0	3. 5 1. 000UN		241	75	67M	
10 POSTEB811__124. 600	1 0. 9999	1. 1	0. 0	48. 0	0. 0										
		1 24. 597		0. 0	20. 0	-11. 1	9 POSTEB	120. 00	1 9	-50. 0	-3. 5 1. 000HI		1176	75	67M
							19 POSTEB911	24. 600	1 1	2. 0	-5. 4		136	14	939A
11 POSTEB811__224. 600	1 0. 9999	1. 1	0. 0	48. 0	0. 0										
		1 24. 597		0. 0	20. 0	-11. 1	9 POSTEB	120. 00	1 10	-50. 0	-3. 5 1. 000HI		1176	75	67M
							19 POSTEB911	24. 600	1 2	2. 0	-5. 4		136	14	939A
12 POSTEB811__324. 600	1 0. 9999	1. 1	0. 0	48. 0	0. 0										
		1 24. 597		0. 0	20. 0	-11. 1	9 POSTEB	120. 00	1 12	-50. 0	-3. 5 1. 000HI		1176	75	67M
							19 POSTEB911	24. 600	1 2	2. 0	-5. 4		136	14	939A
13 POSTEB811__424. 600	1 0. 9999	1. 1	0. 0	48. 0	0. 0										
		1 24. 597		0. 0	20. 0	-11. 1	9 POSTEB	120. 00	1 13	-50. 0	-3. 5 1. 000HI		1176	75	67M
							19 POSTEB911	24. 600	1 2	2. 0	-5. 4		136	14	939A
14 POSTEC	120. 00	1 1. 0000	2. 2	195. 0	0. 0	0. 0									
		1 120. 00		17. 3R	0. 0	0. 0	15 POSTEC811__124. 600	1 14	65. 0	5. 8 1. 000UN		314	97	67M	
							16 POSTEC811__224. 600	1 15	65. 0	5. 8 1. 000UN		314	97	67M	
							17 POSTEC811__324. 600	1 16	65. 0	5. 8 1. 000UN		314	97	67M	
15 POSTEC811__124. 600	1 0. 9998	2. 2	0. 0	48. 0	0. 0										
		1 24. 595		0. 0	20. 0	-6. 0	14 POSTEC	120. 00	1 14	-65. 0	-5. 7 1. 000HI		1532	97	67M
							18 POSTEC911	24. 600	1 2	17. 0	-8. 3		444	47	939A
16 POSTEC811__224. 600	1 0. 9998	2. 2	0. 0	48. 0	0. 0										
		1 24. 595		0. 0	20. 0	-6. 0	14 POSTEC	120. 00	1 15	-65. 0	-5. 7 1. 000HI		1532	97	67M

17	POSTEC811	_324.600	1	0.9998	2.2	0.0	48.0	0.0	-----	18	POSTEC911	24.600	1	2	17.0	-8.3		444	47	939A
			1	24.595		0.0	20.0	-6.0	-----	14	POSTEC	120.00	1	16	-65.0	-5.7	1.000HI	1532	97	67M
									-----	18	POSTEC911	24.600	1	2	17.0	-8.3		444	47	939A
18	POSTEC911	24.600	1	1.0011	1.9	0.0	0.0	0.0	-----	15	POSTEC811	_124.600	1	2	-17.0	6.3		425	45	939A
			1	24.627		0.0	0.0	0.0	-----	16	POSTEC811	_224.600	1	2	-17.0	6.3		425	45	939A
									-----	17	POSTEC811	_324.600	1	2	-17.0	6.3		425	45	939A
										23	POST_CA	24.600	1	2	31.1	-12.5		786	84	939A
										25	POST_CB	24.600	1	1	19.8	-6.6		489	52	939A
19	POSTEB911	24.600	1	1.0008	1.1	0.0	0.0	0.0	-----											
			1	24.620		0.0	0.0	0.0	-----	10	POSTEB811	_124.600	1	1	-2.0	3.4		93	10	939A
									-----	11	POSTEB811	_224.600	1	2	-2.0	3.4		93	10	939A
										12	POSTEB811	_324.600	1	2	-2.0	3.4		93	10	939A
										13	POSTEB811	_424.600	1	2	-2.0	3.4		93	10	939A
										21	POST_BA	24.600	1	2	27.8	-8.5		681	73	939A
										24	POST_BC	24.600	1	2	-19.8	-5.1		479	51	939A
20	POST_AB	24.600	1	1.0007	0.1	0.0	0.0	0.0	-----											
			1	24.618		0.0	0.0	0.0	-----	8	POSTEA911	24.600	1	7	27.7	3.0	1.000LK	654	42	67M
										21	POST_BA	24.600	1	19	-27.7	-3.0	1.000UN	654	42	67M
21	POST_BA	24.600	1	1.0008	0.1	0.0	0.0	0.0	-----											
			1	24.620		0.0	0.0	0.0	-----	19	POSTEB911	24.600	1	2	-27.7	-3.0		654	70	939A
										20	POST_AB	24.600	1	19	27.7	3.0	1.000LK	654	42	67M
22	POST_AC	24.600	1	1.0008	0.1	0.0	0.0	0.0	-----											
			1	24.619		0.0	0.0	0.0	-----	8	POSTEA911	24.600	1	8	31.0	6.6	1.000LK	744	47	67M
										23	POST_CA	24.600	1	20	-31.0	-6.6	1.000UN	744	47	67M
23	POST_CA	24.600	1	1.0009	0.1	0.0	0.0	0.0	-----											
			1	24.622		0.0	0.0	0.0	-----	18	POSTEC911	24.600	1	2	-31.0	-6.6		744	79	939A
										22	POST_AC	24.600	1	20	31.0	6.6	1.000LK	744	47	67M
24	POST_BC	24.600	1	1.0019	1.7	0.0	0.0	0.0	-----											
			1	24.648		0.0	0.0	0.0	-----	19	POSTEB911	24.600	1	2	19.8	-4.7		477	51	939A
										25	POST_CB	24.600	1	1	-6.6	1.6	1.000UN	159	10	67M
										25	POST_CB	24.600	1	2	-6.6	1.6	1.000UN	159	10	67M

25 POST_CB	24.600	1 1.0019	1.7	0.0	0.0	0.0	25 POST_CB	24.600	1 20	-6.6	1.6 1.000UN	159 10	67M
		1 24.648		0.0	0.0	0.0	-----						
							18 POSTEC911	24.600	1 1	-19.8	4.7	477 51	939A
							24 POST_BC	24.600	1 1	6.6	-1.6 1.000LK	159 10	67M
							24 POST_BC	24.600	1 2	6.6	-1.6 1.000LK	159 10	67M
							24 POST_BC	24.600	1 20	6.6	-1.6 1.000LK	159 10	67M

ANNEXE 5**Exemple III : données du réseau et résultats d'écoulement de puissance sur l'outil APREM**

CODE APREM POUR CRÉER LE RÉSEAU DE L'EXEMPLE III

```
% Build the Network Structure
Network = [];
% Zbase = (24.6e3)^2/100e6; % puissance de base

Zline = (0.0023+li*0.023)*Zbase ;
ZlineCA = (0.001+li*0.01)*Zbase ;
ZlineCB = (0.001+li*0.01)*Zbase ;
ZlineBA = (0.001+li*0.01)*Zbase ;

Charging = li*0.02/Zbase ;
ChargingCA = li*0.02/Zbase ;
ChargingCB = li*0.02/Zbase ;
ChargingBA = li*0.02/Zbase ;

%%%%%%%%%%%%%%Ajout des lignes%%%%%%%%%%%%%%%
Network.L4 = cLine('POSTEA811_1', 'POSTA911', Zline, Charging);
Network.L5 = cLine('POSTEA811_2', 'POSTA911', Zline, Charging);
Network.L6 = cLine('POSTEA811_3', 'POSTA911', Zline, Charging);
Network.L7 = cLine('POSTEA811_4', 'POSTA911', Zline, Charging);
Network.L8 = cLine('POSTEA811_5', 'POSTA911', Zline, Charging);
Network.L9 = cLine('POSTEA811_6', 'POSTA911', Zline, Charging);
Network.L10A = cLine('POSTE_BA', 'POSTB911', 6*ZlineBA, 6*ChargingBA);
Network.L10C = cLine('POSTE_CA', 'POSTC911', 10*ZlineCA, 10*ChargingCA);

Network.LB1 = cLine('POSTEB811_1', 'POSTB911', Zline, Charging);
Network.LB2 = cLine('POSTEB811_2', 'POSTB911', Zline, Charging);
Network.LB3 = cLine('POSTEB811_3', 'POSTB911', Zline, Charging);
Network.LB4 = cLine('POSTEB811_4', 'POSTB911', Zline, Charging);

Network.L10BC = cLine('POSTB911', 'POSTE_BC', 5*ZlineCB, 5*ChargingCB);
Network.L10CB = cLine('POSTC911', 'POSTE_CB', Zline, Charging);

Network.LC1 = cLine('POSTEC811_1', 'POSTC911', Zline, Charging);
Network.LC2 = cLine('POSTEC811_2', 'POSTC911', Zline, Charging);
Network.LC3 = cLine('POSTEC811_3', 'POSTC911', Zline, Charging);

%%%%%%%%%%%%%%Tranformateur 120/24.6 au poste C
Ztxfo = 0.00021 + li*0.0010;
Ztxfo120 = ((120e3)^2/100e6)*Ztxfo;% %%%%((24.6e3)^2/100e6)*Ztxfo;
Ztxfo24 = ((24.6e3)^2/100e6)*Ztxfo;
Network.TC1 = cTxfoZ ('POSTEC', '0', 'POSTEC811_1', '0', 120/24.6, 0, Ztxfo120);%
Network.TC2 = cTxfoZ ('POSTEC', '0', 'POSTEC811_2', '0', 120/24.6, 0, Ztxfo120);%
Network.TC3 = cTxfoZ ('POSTEC', '0', 'POSTEC811_3', '0', 120/24.6, 0, Ztxfo120 );%
Network.TC4 = cTxfoZ ('POSTE_BC', '0', 'POSTE_CB', '0', 24.6/24.6, 0, Ztxfo24 );%
%%%%%%%%%%%%%%Tranformateur 120/24.6 au posteB
Network.TB1 = cTxfoZ ('POSTEB', '0', 'POSTEB811_1', '0', 120/24.6, 0, Ztxfo120 );%
Network.TB2 = cTxfoZ ('POSTEB', '0', 'POSTEB811_2', '0', 120/24.6, 0, Ztxfo120 );%
Network.TB3 = cTxfoZ ('POSTEB', '0', 'POSTEB811_3', '0', 120/24.6, 0, Ztxfo120 );%
Network.TB4 = cTxfoZ ('POSTEB', '0', 'POSTEB811_4', '0', 120/24.6, 0, Ztxfo120 );%
%%%%%%%%%%%%%%Tranformateur 24.6/120 au posteA
Network.TA1 = cTxfoZ ('POSTEA811_1', '0', 'POSTEA', '0', 24.6/120,0 , Ztxfo24 );%
Network.TA2 = cTxfoZ ('POSTEA811_2', '0', 'POSTEA', '0', 24.6/120,0 , Ztxfo24);%
Network.TA3 = cTxfoZ ('POSTEA811_3', '0', 'POSTEA', '0', 24.6/120,0 , Ztxfo24);%
Network.TA4 = cTxfoZ ('POSTEA811_4', '0', 'POSTEA', '0', 24.6/120,0 , Ztxfo24);%
Network.TA5 = cTxfoZ ('POSTEA811_5', '0', 'POSTEA', '0', 24.6/120,0 , Ztxfo24);%
Network.TA6 = cTxfoZ ('POSTEA811_6', '0', 'POSTEA', '0', 24.6/120,0 , Ztxfo24);%
Network.TA7 = cTxfoZ ('POSTA911', '0', 'POSTE_AB', '0', 24.6/24.6,0 , Ztxfo24);%
Network.TA10 = cTxfoZ ('POSTA911', '0', 'POSTE_AC', '0', 24.6/24.6,0 , Ztxfo24);%
Network.TA8 = cTxfoZ ('POSTE_AB', '0', 'POSTE_BA', '0', 24.6/24.6,0 , Ztxfo24);%
Network.TA9 = cTxfoZ ('POSTE_AC', '0', 'POSTE_CA', '0', 24.6/24.6,0 , Ztxfo24);%

%Charge au poste A
Network.PQA811 = cLoad('POSTEA811_1', '0', 48e6,14e6,24.6e3);
Network.PQA812 = cLoad('POSTEA811_2', '0', 48e6,14e6,24.6e3);
Network.PQA813 = cLoad('POSTEA811_3', '0', 48e6,14e6,24.6e3);
Network.PQA814 = cLoad('POSTEA811_4', '0', 48e6,14e6,24.6e3);
Network.PQA815 = cLoad('POSTEA811_5', '0', 48e6,14e6,24.6e3);
Network.PQA816 = cLoad('POSTEA811_6', '0', 48e6,14e6,24.6e3);
%Charge du posteB
Network.PQB811 = cLoad('POSTEB811_1', '0', 48e6,20e6,24.6e3);
Network.PQB812 = cLoad('POSTEB811_2', '0', 48e6,20e6,24.6e3);
Network.PQB813 = cLoad('POSTEB811_3', '0', 48e6,20e6,24.6e3);
Network.PQB814 = cLoad('POSTEB811_4', '0', 48e6,20e6,24.6e3);
%Charge au poste C
Network.PQC1 = cLoad('POSTEC811_1', '0', 48e6,20e6,24.6e3);
Network.PQC2 = cLoad('POSTEC811_2', '0', 48e6,20e6,24.6e3);
Network.PQC3 = cLoad('POSTEC811_3', '0', 48e6,20e6,24.6e3);
```

```

%%%%%%%%% Production %%%%%%%%
Network.V1 = cVsource('POSTEA','0',120e3,0);      % Slack bus
Network.V2 = cPVsource('POSTEB','0',200e6,120e3);    % Générateur PV (puissance P régulée)
Network.V3 = cPVsource('POSTEC','0',195e6,120e3);

%%%%%%%%% Compensation %%%%%%%%
Network.Comp2 = cLoad( 'POSTEA811_1','0',0,-1.1e6, 24.6e3);
Network.Comp3 = cLoad( 'POSTEA811_2','0',0,-1.1e6, 24.6e3);
Network.Comp4 = cLoad( 'POSTEA811_3','0',0,-1.1e6, 24.6e3);
Network.Comp5 = cLoad( 'POSTEA811_4','0',0,-1.1e6, 24.6e3);
Network.Comp6 = cLoad( 'POSTEA811_5','0',0,-1.1e6, 24.6e3);
Network.Comp7 = cLoad( 'POSTEA811_6','0',0,-1.1e6, 24.6e3);
Network.Comp10 = cLoad( 'POSTEB811_1','0',0,-11.10e6, 24.6e3);
Network.Comp11 = cLoad( 'POSTEB811_2','0',0,-11.10e6, 24.6e3);
Network.Comp12 = cLoad( 'POSTEB811_3','0',0,-11.10e6, 24.6e3);
Network.Comp13 = cLoad( 'POSTEB811_4','0',0,-11.10e6, 24.6e3);
Network.Comp15 = cLoad( 'POSTEC811_1','0',0,-6.00e6, 24.6e3);
Network.Comp16 = cLoad( 'POSTEC811_2','0',0,-6.00e6, 24.6e3);
Network.Comp17 = cLoad( 'POSTEC811_3','0',0,-6.00e6, 24.6e3);

```

DONNÉES DU RÉSEAU DANS APREM POUR L'EXEMPLE III

cVsource								
ID	ON	k	m	function	Vabs (V)	Angle (deg)	n	
V1	TRUE	POSTEA	GND	0	120 000,00	0,00	1	
cPVsource								
ID	ON	k	m	function	P (W)	Vabs (V)	n	
V2	TRUE	POSTEB	GND	0	200 000 000,00	120 000,00	1	
V3	TRUE	POSTEC	GND	0	195 000 000,00	120 000,00	2	
cTxfoz								
ID	ON	k	m	function	i	j	r	Angle (deg) R (ohm) X (ohm)
TC1	TRUE	POSTEC	GND	0	POSTEC811_1	GND	4,88	0,00 1 4,88 0,14
TC2	TRUE	POSTEC	GND	0	POSTEC811_2	GND	4,88	0,00 2 4,88 0,14
TC3	TRUE	POSTEC	GND	0	POSTEC811_3	GND	4,88	0,00 3 4,88 0,14
TC4	TRUE	POSTE_BC	GND	0	POSTE_CB	GND	1,00	0,00 4 1,00 0,01
TB1	TRUE	POSTEB	GND	0	POSTEB811_1	GND	4,88	0,00 5 4,88 0,14
TB2	TRUE	POSTEB	GND	0	POSTEB811_2	GND	4,88	0,00 6 4,88 0,14
TB3	TRUE	POSTEB	GND	0	POSTEB811_3	GND	4,88	0,00 7 4,88 0,14
TB4	TRUE	POSTEB	GND	0	POSTEB811_4	GND	4,88	0,00 8 4,88 0,14
TA1	TRUE	POSTEA811_1	GND	0	POSTEA	GND	0,21	0,00 9 0,21 0,01
TA2	TRUE	POSTEA811_2	GND	0	POSTEA	GND	0,21	0,00 10 0,21 0,01
TA3	TRUE	POSTEA811_3	GND	0	POSTEA	GND	0,21	0,00 11 0,21 0,01
TA4	TRUE	POSTEA811_4	GND	0	POSTEA	GND	0,21	0,00 12 0,21 0,01
TA5	TRUE	POSTEA811_5	GND	0	POSTEA	GND	0,21	0,00 13 0,21 0,01
TA6	TRUE	POSTEA811_6	GND	0	POSTEA	GND	0,21	0,00 14 0,21 0,01
TA7	TRUE	POSTA911	GND	0	POSTE_AB	GND	1,00	0,00 15 1,00 0,01
TA10	TRUE	POSTA911	GND	0	POSTE_AC	GND	1,00	0,00 16 1,00 0,01
TA8	TRUE	POSTE_AB	GND	0	POSTE_BA	GND	1,00	0,00 17 1,00 0,01
TA9	TRUE	POSTE_AC	GND	0	POSTE_CA	GND	1,00	0,00 18 1,00 0,01
cline								
ID	ON	k	m	function	R (ohm)	X (ohm)	G(S)	B(S)
L4	TRUE	POSTEA811_1	POSTA911	0	0,0139	0,139	0,000	3,30E-03
L5	TRUE	POSTEA811_2	POSTA911	0	0,0139	0,139	0,000	3,30E-03
L6	TRUE	POSTEA811_3	POSTA911	0	0,0139	0,139	0,000	3,30E-03
L7	TRUE	POSTEA811_4	POSTA911	0	0,0139	0,139	0,000	3,30E-03
L8	TRUE	POSTEA811_5	POSTA911	0	0,0139	0,139	0,000	3,30E-03
L9	TRUE	POSTEA811_6	POSTA911	0	0,0139	0,139	0,000	3,30E-03
L10A	TRUE	POSTE_BA	POSTB911	0	0,0363	0,363	0,000	1,98E-02
L10C	TRUE	POSTE_CA	POSTC911	0	0,0605	0,605	0,000	3,30E-02
LB1	TRUE	POSTEB811_1	POSTB911	0	0,0139	0,139	0,000	3,30E-03
LB2	TRUE	POSTEB811_2	POSTB911	0	0,0139	0,139	0,000	3,30E-03
LB3	TRUE	POSTEB811_3	POSTB911	0	0,0139	0,139	0,000	3,30E-03
LB4	TRUE	POSTEB811_4	POSTB911	0	0,0139	0,139	0,000	3,30E-03
L10BC	TRUE	POSTB911	POSTE_BC	0	0,0303	0,303	0,000	1,65E-02
L10CB	TRUE	POSTC911	POSTE_CB	0	0,0139	0,139	0,000	3,30E-03
LC1	TRUE	POSTEC811_1	POSTC911	0	0,0139	0,139	0,000	3,30E-03
LC2	TRUE	POSTEC811_2	POSTC911	0	0,0139	0,139	0,000	3,30E-03
LC3	TRUE	POSTEC811_3	POSTC911	0	0,0139	0,139	0,000	3,30E-03
cload								
ID	ON	k	m	function	P (W)	Q (Var)	Vnom (V)	n
PQC1	TRUE	POSTEC811_1	GND	0	48 000 000,00	20 000 000,00	24 600,00	14
PQC2	TRUE	POSTEC811_2	GND	0	48 000 000,00	20 000 000,00	24 600,00	15
PQC3	TRUE	POSTEC811_3	GND	0	48 000 000,00	20 000 000,00	24 600,00	16
PQA811	TRUE	POSTEA811_1	GND	0	48 000 000,00	14 000 000,00	24 600,00	4
PQA812	TRUE	POSTEA811_2	GND	0	48 000 000,00	14 000 000,00	24 600,00	5
PQA813	TRUE	POSTEA811_3	GND	0	48 000 000,00	14 000 000,00	24 600,00	6
PQA814	TRUE	POSTEA811_4	GND	0	48 000 000,00	14 000 000,00	24 600,00	7

PQA815	TRUE	POSTEA811_5	GND	0	48 000 000,00	14 000 000,00	24 600,00	8
PQA816	TRUE	POSTEA811_6	GND	0	48 000 000,00	14 000 000,00	24 600,00	9
PQ8811	TRUE	POSTEB811_1	GND	0	48 000 000,00	20 000 000,00	24 600,00	10
PQ8812	TRUE	POSTEB811_2	GND	0	48 000 000,00	20 000 000,00	24 600,00	11
PQ8813	TRUE	POSTEB811_3	GND	0	48 000 000,00	20 000 000,00	24 600,00	12
PQ8814	TRUE	POSTEB811_4	GND	0	48 000 000,00	20 000 000,00	24 600,00	13
Comp2	TRUE	POSTEA811_1	GND	0	0,00	-1 100 000,00	24 600,00	17
Comp3	TRUE	POSTEA811_2	GND	0	0,00	-1 100 000,00	24 600,00	18
Comp4	TRUE	POSTEA811_3	GND	0	0,00	-1 100 000,00	24 600,00	19
Comp5	TRUE	POSTEA811_4	GND	0	0,00	-1 100 000,00	24 600,00	20
Comp6	TRUE	POSTEA811_5	GND	0	0,00	-1 100 000,00	24 600,00	21
Comp7	TRUE	POSTEA811_6	GND	0	0,00	-1 100 000,00	24 600,00	22
Comp10	TRUE	POSTEB811_1	GND	0	0,00	-11 100 000,00	24 600,00	23
Comp11	TRUE	POSTEB811_2	GND	0	0,00	-11 100 000,00	24 600,00	24
Comp12	TRUE	POSTEB811_3	GND	0	0,00	-11 100 000,00	24 600,00	25
Comp13	TRUE	POSTEB811_4	GND	0	0,00	-11 100 000,00	24 600,00	26
Comp15	TRUE	POSTEC811_1	GND	0	0,00	-6 000 000,00	24 600,00	27
Comp16	TRUE	POSTEC811_2	GND	0	0,00	-6 000 000,00	24 600,00	28
Comp17	TRUE	POSTEC811_3	GND	0	0,00	-6 000 000,00	24 600,00	29

RÉSULTATS DE L'ÉCOULEMENT DE PUISSANCE DU RÉSEAU DE L'EXEMPLE III SUR APREM

18-Jan-2011 21:47:57 Case_1

Bus Data												
Voltage :												
Bus ID	Mag(V)	Ang(deg)										
GND	0.000	0.000										
POSTEA811_1	24595.730	-0.021										
POSTA911	24615.906	0.105										
POSTEA811_2	24595.730	-0.021										
POSTEA811_3	24595.730	-0.021										
POSTEA811_4	24595.730	-0.021										
POSTEA811_5	24595.730	-0.021										
POSTEA811_6	24595.730	-0.021										
POSTE_BA	24620.244	0.136										
POSTB911	24620.439	1.095										
POSTE_CA	24622.358	0.139										
POSTC911	24626.961	1.937										
POSTEB811_1	24596.557	1.127										
POSTEB811_2	24596.557	1.127										
POSTEB811_3	24596.557	1.127										
POSTEB811_4	24596.557	1.127										
POSTE_BC	24647.718	1.658										
POSTE_CB	24647.589	1.670										
POSTEC811_1	24595.227	2.170										
POSTEC811_2	24595.227	2.170										
POSTEC811_3	24595.227	2.170										
POSTEC	120000.000	2.207										
POSTEB	120000.000	1.155										
POSTEA	120000.000	0.000										
POSTE_AB	24618.074	0.120										
POSTE_AC	24619.131	0.122										
Branch Data												
Branch ID	ON/OFF	From Bus k	To Bus m	Voltage: Mag(V)	Current: Ang(deg)	Power: P(W)	Q(Var)	Bus k: P(W)	Bus m: P(W)	Bus k: Q(Var)	Bus m: Q(Var)	
L4	1	POSTEA811_1	POSTA911	5.756e+001	-110.476	2.446e+002	159.912	2.36e+003	-1.98e+006	-9.79e+006	-3.58e+006	9.79e+006 1.60e+006
L5	1	POSTEA811_2	POSTA911	5.756e+001	-110.476	2.446e+002	159.912	2.36e+003	-1.98e+006	-9.79e+006	-3.58e+006	9.79e+006 1.60e+006
L6	1	POSTEA811_3	POSTA911	5.756e+001	-110.476	2.446e+002	159.912	2.36e+003	-1.98e+006	-9.79e+006	-3.58e+006	9.79e+006 1.60e+006
L7	1	POSTEA811_4	POSTA911	5.756e+001	-110.476	2.446e+002	159.912	2.36e+003	-1.98e+006	-9.79e+006	-3.58e+006	9.79e+006 1.60e+006
L8	1	POSTEA811_5	POSTA911	5.756e+001	-110.476	2.446e+002	159.912	2.36e+003	-1.98e+006	-9.79e+006	-3.58e+006	9.79e+006 1.60e+006
L9	1	POSTEA811_6	POSTA911	5.756e+001	-110.476	2.446e+002	159.912	2.36e+003	-1.98e+006	-9.79e+006	-3.58e+006	9.79e+006 1.60e+006
L10A	1	POSTE_BA	POSTB911	4.123e+002	-89.412	6.524e+002	173.896	4.63e+002	-1.16e+007	-2.77e+007	-3.02e+006	2.77e+007 -8.53e+006
L10C	1	POSTE_CA	POSTC911	7.728e+002	-89.303	7.455e+002	168.153	9.77e+004	-1.91e+007	-3.11e+007	-6.60e+006	3.12e+007 -1.25e+007
LB1	1	POSTEB811_1	POSTB911	2.757e+001	151.123	1.355e+002	70.918	5.41e+002	-2.00e+006	1.99e+006	-5.42e+006	-1.99e+006 3.42e+006
LB2	1	POSTEB811_2	POSTB911	2.757e+001	151.123	1.355e+002	70.918	5.41e+002	-2.00e+006	1.99e+006	-5.42e+006	-1.99e+006 3.42e+006
LB3	1	POSTEB811_3	POSTB911	2.757e+001	151.123	1.355e+002	70.918	5.41e+002	-2.00e+006	1.99e+006	-5.42e+006	-1.99e+006 3.42e+006
LB4	1	POSTEB811_4	POSTB911	2.757e+001	151.123	1.355e+002	70.918	5.41e+002	-2.00e+006	1.99e+006	-5.42e+006	-1.99e+006 3.42e+006
L10BC	1	POSTB911	POSTE_BC	2.436e+002	-95.052	4.781e+002	166.438	1.94e+004	-9.83e+006	1.97e+007	-5.16e+006	1.97e+007 -4.67e+006
L10CB	1	POSTC911	POSTE_CB	1.166e+002	101.994	4.881e+002	20.360	9.67e+003	-1.91e+006	1.98e+007	-6.58e+006	-1.97e+007 4.67e+006
LC1	1	POSTEC811_1	POSTC911	1.051e+002	109.626	4.436e+002	28.124	7.86e+003	-1.92e+006	1.70e+007	-8.27e+006	-1.70e+007 6.35e+006
LC2	1	POSTEC811_2	POSTC911	1.051e+002	109.626	4.436e+002	28.124	7.86e+003	-1.92e+006	1.70e+007	-8.27e+006	-1.70e+007 6.35e+006
LC3	1	POSTEC811_3	POSTC911	1.051e+002	109.626	4.436e+002	28.124	7.86e+003	-1.92e+006	1.70e+007	-8.27e+006	-1.70e+007 6.35e+006
PQC1	1	POSTEC811_1	GND	2.460e+004	2.170	1.221e+003	-20.450	4.80e+007	2.00e+007	4.80e+007	2.00e+007	0.00e+000 0.00e+000
POC2	1	POSTEC811_2	GND	2.460e+004	2.170	1.221e+003	-20.450	4.80e+007	2.00e+007	4.80e+007	2.00e+007	0.00e+000 0.00e+000
PQC3	1	POSTEC811_3	GND	2.460e+004	2.170	1.221e+003	-20.450	4.80e+007	2.00e+007	4.80e+007	2.00e+007	0.00e+000 0.00e+000
TC1	1	POSTEC	GND	1.200e+005	2.207	3.140e+002	-2.868	6.50e+007	5.77e+006	6.50e+007	5.77e+006	0.00e+000 0.00e+000
	1	POSTEC811_1	GND	2.460e+004	2.170	1.532e+003	177.132	-6.50e+007	-5.73e+006	-6.50e+007	-5.73e+006	0.00e+000 0.00e+000

TC2	1	POSTEC	GND	1.200e+005	2.207	3.140e+002	-2.868	6.50e+007	5.77e+006	6.50e+007	5.77e+006	0.00e+000	0.00e+000
	1	POSTEC811_2	GND	2.460e+004	2.170	1.532e+003	177.132	-6.50e+007	-5.73e+006	-6.50e+007	-5.73e+006	0.00e+000	0.00e+000
TC3	1	POSTEC	GND	1.200e+005	2.207	3.140e+002	-2.868	6.50e+007	5.77e+006	6.50e+007	5.77e+006	0.00e+000	0.00e+000
	1	POSTEC811_3	GND	2.460e+004	2.170	1.532e+003	177.132	-6.50e+007	-5.73e+006	-6.50e+007	-5.73e+006	0.00e+000	0.00e+000
TC4	1	POSTE_BC	GND	2.465e+004	1.658	4.753e+002	-165.021	-1.97e+007	4.67e+006	-1.97e+007	4.67e+006	0.00e+000	0.00e+000
	1	POSTE_CB	GND	2.465e+004	1.670	4.753e+002	14.979	1.97e+007	-4.67e+006	1.97e+007	-4.67e+006	0.00e+000	0.00e+000
TB1	1	POSTEB	GND	1.200e+005	1.155	2.412e+002	-2.856	5.00e+007	3.51e+006	5.00e+007	3.51e+006	0.00e+000	0.00e+000
	1	POSTEB811_1	GND	2.460e+004	1.127	1.176e+003	177.144	-5.00e+007	-3.48e+006	-5.00e+007	-3.48e+006	0.00e+000	0.00e+000
TB2	1	POSTEB	GND	1.200e+005	1.155	2.412e+002	-2.856	5.00e+007	3.51e+006	5.00e+007	3.51e+006	0.00e+000	0.00e+000
	1	POSTEB811_2	GND	2.460e+004	1.127	1.176e+003	177.144	-5.00e+007	-3.48e+006	-5.00e+007	-3.48e+006	0.00e+000	0.00e+000
TB3	1	POSTEB	GND	1.200e+005	1.155	2.412e+002	-2.856	5.00e+007	3.51e+006	5.00e+007	3.51e+006	0.00e+000	0.00e+000
	1	POSTEB811_3	GND	2.460e+004	1.127	1.176e+003	177.144	-5.00e+007	-3.48e+006	-5.00e+007	-3.48e+006	0.00e+000	0.00e+000
TB4	1	POSTEB	GND	1.200e+005	1.155	2.412e+002	-2.856	5.00e+007	3.51e+006	5.00e+007	3.51e+006	0.00e+000	0.00e+000
	1	POSTEB811_4	GND	2.460e+004	1.127	1.176e+003	177.144	-5.00e+007	-3.48e+006	-5.00e+007	-3.48e+006	0.00e+000	0.00e+000
PQA811	1	POSTEA811_1	GND	2.460e+004	-0.021	1.174e+003	-16.281	4.80e+007	1.40e+007	4.80e+007	1.40e+007	0.00e+000	0.00e+000
PQA812	1	POSTEA811_2	GND	2.460e+004	-0.021	1.174e+003	-16.281	4.80e+007	1.40e+007	4.80e+007	1.40e+007	0.00e+000	0.00e+000
PQA813	1	POSTEA811_3	GND	2.460e+004	-0.021	1.174e+003	-16.281	4.80e+007	1.40e+007	4.80e+007	1.40e+007	0.00e+000	0.00e+000
PQA814	1	POSTEA811_4	GND	2.460e+004	-0.021	1.174e+003	-16.281	4.80e+007	1.40e+007	4.80e+007	1.40e+007	0.00e+000	0.00e+000
PQA815	1	POSTEA811_5	GND	2.460e+004	-0.021	1.174e+003	-16.281	4.80e+007	1.40e+007	4.80e+007	1.40e+007	0.00e+000	0.00e+000
PQA816	1	POSTEA811_6	GND	2.460e+004	-0.021	1.174e+003	-16.281	4.80e+007	1.40e+007	4.80e+007	1.40e+007	0.00e+000	0.00e+000
PQB811	1	POSTEB811_1	GND	2.460e+004	1.127	1.221e+003	-21.493	4.80e+007	2.00e+007	4.80e+007	2.00e+007	0.00e+000	0.00e+000
PQB812	1	POSTEB811_2	GND	2.460e+004	1.127	1.221e+003	-21.493	4.80e+007	2.00e+007	4.80e+007	2.00e+007	0.00e+000	0.00e+000
PQB813	1	POSTEB811_3	GND	2.460e+004	1.127	1.221e+003	-21.493	4.80e+007	2.00e+007	4.80e+007	2.00e+007	0.00e+000	0.00e+000
PQB814	1	POSTEB811_4	GND	2.460e+004	1.127	1.221e+003	-21.493	4.80e+007	2.00e+007	4.80e+007	2.00e+007	0.00e+000	0.00e+000
TA1	1	POSTEA811_1	GND	2.460e+004	-0.021	9.233e+002	166.266	-3.82e+007	-9.32e+006	-3.82e+007	-9.32e+006	0.00e+000	0.00e+000
	1	POSTEA	GND	1.200e+005	0.000	1.893e+002	-13.734	3.82e+007	9.34e+006	3.82e+007	9.34e+006	0.00e+000	0.00e+000
TA2	1	POSTEA811_2	GND	2.460e+004	-0.021	9.233e+002	166.266	-3.82e+007	-9.32e+006	-3.82e+007	-9.32e+006	0.00e+000	0.00e+000
	1	POSTEA	GND	1.200e+005	0.000	1.893e+002	-13.734	3.82e+007	9.34e+006	3.82e+007	9.34e+006	0.00e+000	0.00e+000
TA3	1	POSTEA811_3	GND	2.460e+004	-0.021	9.233e+002	166.266	-3.82e+007	-9.32e+006	-3.82e+007	-9.32e+006	0.00e+000	0.00e+000
	1	POSTEA	GND	1.200e+005	0.000	1.893e+002	-13.734	3.82e+007	9.34e+006	3.82e+007	9.34e+006	0.00e+000	0.00e+000
TA4	1	POSTEA811_4	GND	2.460e+004	-0.021	9.233e+002	166.266	-3.82e+007	-9.32e+006	-3.82e+007	-9.32e+006	0.00e+000	0.00e+000
	1	POSTEA	GND	1.200e+005	0.000	1.893e+002	-13.734	3.82e+007	9.34e+006	3.82e+007	9.34e+006	0.00e+000	0.00e+000
TA5	1	POSTEA811_5	GND	2.460e+004	-0.021	9.233e+002	166.266	-3.82e+007	-9.32e+006	-3.82e+007	-9.32e+006	0.00e+000	0.00e+000
	1	POSTEA	GND	1.200e+005	0.000	1.893e+002	-13.734	3.82e+007	9.34e+006	3.82e+007	9.34e+006	0.00e+000	0.00e+000
TA6	1	POSTEA811_6	GND	2.460e+004	-0.021	9.233e+002	166.266	-3.82e+007	-9.32e+006	-3.82e+007	-9.32e+006	0.00e+000	0.00e+000
	1	POSTEA	GND	1.200e+005	0.000	1.893e+002	-13.734	3.82e+007	9.34e+006	3.82e+007	9.34e+006	0.00e+000	0.00e+000
TA7	1	POSTA911	GND	2.462e+004	0.105	6.524e+002	173.896	-2.77e+007	-3.01e+006	-2.77e+007	-3.01e+006	0.00e+000	0.00e+000
	1	POSTE_AB	GND	2.462e+004	0.120	6.524e+002	-6.104	2.77e+007	3.02e+006	2.77e+007	3.02e+006	0.00e+000	0.00e+000
TA10	1	POSTA911	GND	2.462e+004	0.105	7.455e+002	168.153	-3.11e+007	-6.58e+006	-3.11e+007	-6.58e+006	0.00e+000	0.00e+000
	1	POSTE_AC	GND	2.462e+004	0.122	7.455e+002	-11.847	3.11e+007	6.59e+006	3.11e+007	6.59e+006	0.00e+000	0.00e+000
TA8	1	POSTE_AB	GND	2.462e+004	0.120	6.524e+002	173.896	-2.77e+007	-3.02e+006	-2.77e+007	-3.02e+006	0.00e+000	0.00e+000
	1	POSTE_BA	GND	2.462e+004	0.136	6.524e+002	-6.104	2.77e+007	3.02e+006	2.77e+007	3.02e+006	0.00e+000	0.00e+000
TA9	1	POSTE_AC	GND	2.462e+004	0.122	7.455e+002	168.153	-3.11e+007	-6.59e+006	-3.11e+007	-6.59e+006	0.00e+000	0.00e+000
	1	POSTE_CA	GND	2.462e+004	0.139	7.455e+002	-11.847	3.11e+007	6.60e+006	3.11e+007	6.60e+006	0.00e+000	0.00e+000
V1	1	POSTEA	GND	1.200e+005	0.000	1.136e+003	166.266	-2.29e+008	-5.60e+007	-2.29e+008	-5.60e+007	0.00e+000	0.00e+000
V2	1	POSTEB	GND	1.200e+005	1.155	9.646e+002	177.144	2.00e+008	-1.40e+007	2.00e+008	-1.40e+007	0.00e+000	0.00e+000
V3	1	POSTEC	GND	1.200e+005	2.207	9.419e+002	177.132	-1.95e+008	-1.73e+007	-1.95e+008	-1.73e+007	0.00e+000	0.00e+000
Comp2	1	POSTEA811_1	GND	2.460e+004	-0.021	2.582e+001	89.979	0.00e+000	-1.10e+006	0.00e+000	-1.10e+006	0.00e+000	0.00e+000
Comp3	1	POSTEA811_2	GND	2.460e+004	-0.021	2.582e+001	89.979	-5.68e-014	-1.10e+006	-5.68e-014	-1.10e+006	0.00e+000	0.00e+000
Comp4	1	POSTEA811_3	GND	2.460e+004	-0.021	2.582e+001	89.979	-5.68e-014	-1.10e+006	-5.68e-014	-1.10e+006	0.00e+000	0.00e+000
Comp5	1	POSTEA811_4	GND	2.460e+004	-0.021	2.582e+001	89.979	-5.68e-014	-1.10e+006	-5.68e-014	-1.10e+006	0.00e+000	0.00e+000
Comp6	1	POSTEA811_5	GND	2.460e+004	-0.021	2.582e+001	89.979	0.00e+000	-1.10e+006	0.00e+000	-1.10e+006	0.00e+000	0.00e+000
Comp7	1	POSTEA811_6	GND	2.460e+004	-0.021	2.582e+001	89.979	-5.68e-014	-1.10e+006	-5.68e-014	-1.10e+006	0.00e+000	0.00e+000
Comp10	1	POSTEB811_1	GND	2.460e+004	1.127	2.605e+002	91.127	0.00e+000	-1.11e+007	0.00e+000	-1.11e+007	0.00e+000	0.00e+000
Comp11	1	POSTEB811_2	GND	2.460e+004	1.127	2.605e+002	91.127	-2.91e-011	-1.11e+007	-2.91e-011	-1.11e+007	0.00e+000	0.00e+000
Comp12	1	POSTEB811_3	GND	2.460e+004	1.127	2.605e+002	91.127	-2.91e-011	-1.11e+007	-2.91e-011	-1.11e+007	0.00e+000	0.00e+000
Comp13	1	POSTEB811_4	GND	2.460e+004	1.127	2.605e+002	91.127	-2.91e-011	-1.11e+007	-2.91e-011	-1.11e+007	0.00e+000	0.00e+000
Comp15	1	POSTEC811_1	GND	2.460e+004	2.170	1.408e+002	92.170	0.00e+000	-6.00e+006	0.00e+000	-6.00e+006	0.00e+000	0.00e+000
Comp16	1	POSTEC811_2	GND	2.460e+004	2.170	1.408e+002	92.170	-2.91e-011	-6.00e+006	-2.91e-011	-6.00e+006	0.00e+000	0.00e+000
Comp17	1	POSTEC811_3	GND	2.460e+004	2.170	1.408e+002	92.170	-2.91e-011	-6.00e+006	-2.91e-011	-6.00e+006	0.00e+000	0.00e+000

----- -----
TOTAL: 1.931e-005 3.48e-005