



HAL
open science

Applications de l'intelligence artificielle à la détection et l'isolation de pannes multiples dans un réseau de télécommunications

Serge Romaric Tembo Mouafo

► **To cite this version:**

Serge Romaric Tembo Mouafo. Applications de l'intelligence artificielle à la détection et l'isolation de pannes multiples dans un réseau de télécommunications. Réseaux et télécommunications [cs.NI]. Ecole nationale supérieure Mines-Télécom Atlantique, 2017. Français. NNT : 2017IMTA0004 . tel-01781480

HAL Id: tel-01781480

<https://theses.hal.science/tel-01781480>

Submitted on 30 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

**UNIVERSITE
BRETAGNE
LOIRE**

THÈSE / IMT Atlantique

sous le sceau de l'Université Bretagne Loire
pour obtenir le grade de

DOCTEUR D'IMT Atlantique

Mention : Informatique

École Doctorale Matisse

Présentée par

Serge Romaric Tembo Mouafo

Préparée dans le département Informatique
Laboratoire Irisa

Applications de l'intelligence artificielle à la détection et l'isolation de pannes multiples dans un réseau de télécommunications

Thèse soutenue le 23 janvier 2017

devant le jury composé de :

Bernard Cousin

Professeur, Esir - Université de Rennes 1 / président

Hind Castel

Professeur, Télécom SudParis / rapporteur

Olivier Brun

Chargé de recherche (HDR), LAAS-CNRS - Toulouse / rapporteur

Eric Fabre

Directeur de recherche, Irisa - Rennes / examinateur

Yassine Ghamri-Doudane

Professeur, Université de la Rochelle / examinateur

Sandrine Vaton

Professeur, IMT Atlantique / directrice de thèse

Jean-Luc Courant

Ingénieur, Orange Labs - Lannion / invité

Stéphane Gosselin

Ingénieur, Orange Labs - Lannion / invité

Table des matières

1	Introduction	5
1.1	Énoncé du problème	6
1.2	Contribution	7
1.3	Structure de la dissertation	8
2	État de l'art dans le domaine du diagnostic des réseaux de télécommunication	11
2.1	Introduction	12
2.2	Système expert à base de règles	12
2.3	Système expert à base de modèles	17
2.4	Le diagnostic comme système d'apprentissage artificiel	21
2.5	Réseau de Petri	29
2.6	Graphe de dépendances	31
2.7	Réseaux bayésiens	35
2.8	Discussions et conclusion	41
3	Algorithmes d'inférence exacte sur un réseau bayésien	44
3.1	Introduction	45
3.2	Quelques bases en théorie des probabilités	45
3.3	Opération de marginalisation	47
3.4	Algorithme de somme-produit	48
3.5	Algorithme d'inférence sur un arbre de jonction	57
3.6	Discussion et conclusion	63
4	Apprentissage artificiel des paramètres d'un réseau bayésien	64
4.1	Introduction	65
4.2	Estimation paramétrique au sens du Maximum de vraisemblance (MLE)	65
4.3	L'algorithme MLE avec des données incomplètes	68
4.4	L'algorithme EM sur un réseau bayésien	71
4.5	Discussion et conclusion	78
5	Modèle générique pour le diagnostic automatique d'un réseau de télécommunication	79
5.1	Introduction	80
5.2	Propagation de pannes dans un réseau	80
5.3	Description du modèle générique	81
5.4	Formalisme du modèle générique	85
5.5	Calcul de diagnostic du modèle générique	86
5.6	Capacité de reconfiguration du modèle générique	87
5.7	Conclusion et perspectives	91

6	Modélisation probabiliste du réseau FTTH (Fiber To The Home) de type GPON (Gigabit Passive Optical Network)	93
6.1	Introduction	94
6.2	Modèle du réseau GPON-FTTH	94
6.3	Paramètres experts du modèle initial d'un PON	100
6.4	Validation du modèle initial d'un PON	104
6.5	Comparaison entre système expert et modèle probabiliste de diagnostic	113
6.6	Discussion et conclusion	116
7	Ajustement automatique des paramètres du modèle GPON-FTTH : application de l'algorithme EM	118
7.1	Introduction	119
7.2	Contexte applicatif de l'algorithme EM	119
7.3	L'EM avec les données réelles d'apprentissage	120
7.4	Valeur ajoutée de l'EM pour le diagnostic du réseau d'accès GPON-FTTH	123
7.5	Discussion et conclusion	127
8	Extension du modèle GPON-FTTH : prise en compte du réseau de distribution optique	128
8.1	Introduction	129
8.2	Diversité des ingénieries de l'ODN (Optical Distribution Network)	129
8.3	Modélisation automatique de l'ODN	130
8.4	Modèle complet du réseau GPON-FTTH	133
8.5	Valeur ajoutée du modèle de l'ODN pour le diagnostic GPON-FTTH	134
8.6	Discussion et conclusion	137
9	Conclusion	139
9.1	Rappel des objectifs de la thèse	140
9.2	Synthèse des contributions	140
9.3	Perspectives envisagées	142
	Actions de valorisation	143
	Références bibliographiques	144

Liste des figures

2.1	Organisation des systèmes experts à base de règles CRONK et collab. [1998].	13
2.2	Un exemple de l'algorithme "forward chaining" PHAM [1988].	14
2.3	Modèle en couche des dépendances entre services et fonctions d'un réseau MALGORZATA et ADARSHPAL [2001].	18
2.4	Modèle hiérarchique basé sur l'architecture TMN KÄTKER et GEIHS [1997].	20
2.5	Système de diagnostic apprenant.	23
2.6	Cycle de vie d'un système de raisonnement à base de cas.	24
2.7	Distribution des connaissances dans un système CBR RICHTER [1995].	25
2.8	Schéma synoptique de deux neurones connectés SHIFFMAN [2012].	27
2.9	Structure fonctionnelle d'un neurone artificiel.	28
2.10	Propagation d'une perte de signal sur le port P du module STM-1 du réseau SDH AGHASARYAN et collab. [1998].	30
2.11	Réseau de Petri modélisant la propagation d'une rupture du signal sur un port STM-1 dans le réseau SDH AGHASARYAN et collab. [1998].	31
2.12	Les composants du graphe de dépendances causales JINGXIAN et collab. [2010] JINGXIAN et collab. [2011].	32
2.13	Un simple réseau bayésien RUSSEL et NORVIG [2003].	36
2.14	Segment d'un réseau bayésien en arbre.	37
3.1	Un exemple simple d'un réseau bayésien	46
3.2	Fragment d'un réseau bayésien en arbre.	48
3.3	Calcul local de la croyance d'un noeud dans un réseau bayésien en arbre.	50
3.4	Mécanisme de propagation d'évidences dans un réseau bayésien en arbre.	51
3.5	Fragment d'un réseau bayésien avec une topologie en polyarbre	53
3.6	Message $\lambda_X(u_i)$ envoyé par le noeud X à un parent U_i	53
3.7	Calcul de la croyance d'un noeud dans un réseau bayésien en polyarbre.	54
3.8	Illustration des étapes de construction d'un arbre de jonction.	58
3.9	Arbre de jonction du réseau bayésien de la Figure 3.1	59
4.1	Principe général de l'algorithme EM.	75
5.1	Modèle générique d'un réseau de télécommunication.	82
5.2	Un exemple illustratif du modèle générique.	83
5.3	Communication entre les couches 2 et la couche 3 pour le calcul des croyances des noeuds de la couche 2.	86
5.4	Processus de reconfiguration initié à la couche 2.	88
5.5	Machine à états de l'agent de reconfiguration de la couche 2.	89
5.6	Processus de reconfiguration initié à la couche 1.	90
5.7	Machine à états de l'agent de reconfiguration de la couche 1.	90

6.1	Architecture du réseau GPON-FTTH.	94
6.2	Modèle initial d'un PON du réseau d'accès GPON-FTTH.	96
7.1	Histogramme des log-vraisemblances des cas de l'échantillon de données. .	121
7.2	L'évolution de la log-vraisemblance des données mesurées sur un réseau GPON-FTTH opérationnel avec les itérations.	123
8.1	Schéma synoptique d'un coupleur de taux 1/4.	129
8.2	Schéma synoptique de l'ingénierie ODN [(2,), (2, 2), (16, 16, 16, 16)].	130
8.3	Modèle du réseau de distribution optique d'un PON du réseau GPON-FTTH.	131
8.4	Modèle de l'ODN pour l'ingénierie : [(2,), (2, 2), (16, 16, 16, 16)]	133

Liste des tableaux

2.1	Tableau récapitulatif des différentes méthodes de diagnostic.	42
3.1	Paramètres du réseau bayésien présenté à la Figure 3.1.	46
4.1	Un exemple d'échantillon de données incomplet.	75
4.2	Paramètres θ^0 du réseau bayésien 3.1.	76
6.1	Distribution de probabilités conditionnelles $\mathbb{P}(\text{LOF}_{us} \text{O}_1, \text{LCDG}_{us})$	101
6.2	Distribution de probabilités conditionnelles $\mathbb{P}(\text{LCDG}_{us} \text{BIP}_{us}, \text{DOW})$	102
6.3	Distribution de probabilités conditionnelles $\mathbb{P}(\text{LOS}_{us} \text{Rx}_{\text{OLT}}, \text{TIA})$	102
6.4	Distribution de probabilités conditionnelles $\mathbb{P}(\text{Rx}_{\text{OLT}} \bullet, \text{V}_0, \bullet, \text{Tx}_{\text{ONT}})$	103
6.5	Distribution de probabilités conditionnelles $\mathbb{P}(\text{Rx}_{\text{OLT}} \bullet, \bullet, \text{BR}, \text{Tx}_{\text{ONT}})$	103
6.6	Distribution de probabilités conditionnelles $\mathbb{P}(\text{Rx}_{\text{OLT}} \text{OK}, \text{V}_1, \text{BR}, \text{Tx}_{\text{ONT}})$	103
6.7	Distribution de probabilités conditionnelles $\mathbb{P}(\text{Rx}_{\text{OLT}} \text{AT}, \text{V}_1, \text{AT}, \text{Tx}_{\text{ONT}})$	104
6.8	État nominal de l'arbre PON.	105
6.9	Rupture soudaine de la communication entre l'OLT et l'ONT ₁	106
6.10	Rupture progressive de la communication entre l'OLT et l'ONT ₁	106
6.11	Dégradation complètement observée du signal entre l'OLT et l'ONT ₁	107
6.12	Dégradation partiellement observée du signal entre l'OLT et l'ONT ₁	107
6.13	Dégradation du signal entre l'OLT et chacun des deux ONTs.	108
6.14	Perte de la communication sur le canal descendant entre l'OLT et l'ONT ₁	108
6.15	Dégradation du signal entre l'OLT et chacun des ONTs connectés sur un PON de grande dimension.	109
6.16	États inférés des variables manquantes du cas de la Table 6.15.	109
6.17	Dégradation du signal entre l'OLT et chacun des voisins de l'ONT ₁ sur un PON de grande dimension.	110
6.18	Dégradation du signal entre l'OLT et l'ONT ₁ connecté sur un PON de grande dimension.	110
6.19	Rupture soudaine de communication entre l'OLT et l'ONT ₁ sur un PON de grande dimension.	111
6.20	Rupture soudaine de communication entre l'OLT et chacun des ONTs connectés sur un PON de grande dimension.	112
6.21	Interruption différée de l'alimentation électrique d'un ONT.	112
6.22	Les états probables des variables manquantes pourtant utiles pour détecter une alimentation éteinte ou défectueuse de l'ONT ₁ dans la Table 6.21.	112
6.23	Interruption instantanée de l'alimentation électrique d'un ONT.	113
6.24	Dégradation du signal entre l'OLT et un ONT avec des paramètres nominaux.	113
6.25	Probabilités d'observer les alarmes DOW[1] et TIA[1] relatives à l'ONT ₁ et a posteriori aux évidences du cas de la Table 6.24.	113

6.26	Matrice de confusion entre le système expert à base de règles de diagnostic DELC et l'outil de diagnostic probabiliste PANDA.	114
7.1	Matrice de confusion à deux dimensions pour la comparaison des résultats de diagnostic des deux modèles.	124
7.2	Affaiblissement du signal sur le canal montant entre l'ONT ₁ et l'OLT.	124
7.3	Affaiblissement du signal entre l'OLT et chacun des ONTs voisins de l'ONT ₁	125
7.4	Transmission à très faible puissance de l'ONT ₁	125
7.5	Perte de trames de données entre l'OLT et l'ONT ₁	126
7.6	Rupture soudaine de la communication sur un PON à un seul ONT.	126
7.7	Rupture soudaine de la communication entre l'OLT et l'ONT ₁ et dégradation du signal entre l'OLT et chacun des ONTs voisins.	127
8.1	Affaiblissement du signal entre l'OLT et l'ONT ₁ et prise en compte de l'ODN.	135
8.2	Connexion de toutes les sorties d'un coupleur à des ONTs à faibles puissances.	135
8.3	Connexion des ONTs à faibles puissances à toutes les sorties de deux coupleurs précédés par un coupleur commun dans l'arborescence du réseau de distribution.	136
8.4	Connexion des ONTs à faibles puissances à toutes les sorties de deux coupleurs n'étant pas précédés par un coupleur commun dans l'arborescence de l'ODN.	136
8.5	Connexion des sorties d'un coupleur à certains ONTs à faibles niveaux de puissances et d'autres ONTs à puissances nominales.	137

Remerciements

Les travaux de thèse présentés dans ce manuscrit ont été réalisés dans le cadre d'une convention industrielle de formation par la recherche (CIFRE) entre Orange Labs (Lannion) et Telecom Bretagne (Brest). Je tiens tout d'abord à remercier ces laboratoires et plus particulièrement Messieurs Michel BEUVELOT, Stéphane GOSELIN et Madame Sandrine VATON de m'avoir donné l'opportunité de travailler avec eux et de m'avoir très chaleureusement accueilli au sein de leur équipe de recherche respective.

Je souhaite témoigner ma gratitude à ma directrice de thèse Madame Sandrine VATON, à mes encadrants Messieurs Jean-Luc COURANT, Stéphane GOSELIN, Michel BEUVELOT pour tout l'intérêt qu'ils ont porté à mes travaux, pour leurs conseils avisés, pour la rigueur qu'ils ont inculquée en moi durant ces trois dernières années. J'ai découvert grâce à eux le monde de l'intelligence artificielle et celui des méthodes probabilistes appliquées à la gestion de l'incertitude dans les systèmes intelligents. Leurs pertinentes remarques et commentaires ont permis d'améliorer la clarté de la présentation des idées véhiculées dans ce manuscrit.

J'adresse mes remerciements à Madame Hind CASTEL et Monsieur Olivier BRUN pour l'intérêt qu'ils ont porté à ma thèse en acceptant d'être les rapporteurs de ce mémoire, à Monsieur Éric FABRE pour ses précieux conseils lors de ma soutenance de mi-parcours. Je remercie aussi Messieurs Bernard COUSIN, Yacine GHAMRI DOUDANE pour l'honneur qu'ils m'ont fait en acceptant de faire partie de mon jury.

Mes remerciements vont aussi naturellement à l'ensemble des membres de l'équipe ADN d' Orange Labs à Lannion et particulièrement à Messieurs Alexandre FERRIEUX, Manuel RÉGENT, Renaud LE MEUR, et Madame Anne-Marie BLANCHARD avec qui j'ai travaillé et échangé sur de nombreux sujets techniques de l'industrie des télécommunications. Je remercie aussi des personnes chaleureuses que j'ai rencontrées à Orange Labs comme Messieurs Alain RINGNET, Philippe MERCIER et Daniel VENMANI avec qui j'ai passé de très bons moments et pour leurs sages conseils d'adultes dans la vie quotidienne.

Je tiens aussi à remercier d'autres doctorants et apprentis d'Orange Labs avec qui nous avons sympathisé, multiplié des activités sportives et s'entraider au point de devenir de bons amis. Je pense notamment à Messieurs Yazid KACED, Fabien DOUCET, Audrey SEDJRO, Frigui NEJM, Simon RICORDEAU et bien d'autres.

Enfin, c'est avec beaucoup de nostalgie que je remercie de tout cœur mes parents Monsieur MOUAFO Jean et Madame MAKAMCHE Émilienne pour tous les efforts qu'ils ont dû faire pour me donner l'opportunité d'étudier en France. À tous mes frères et sœurs et à ma famille en général, je dis un grand merci pour leurs encouragements et le soutien moral que j'ai puisé en eux. Je ne peux que leur dédier ce modeste labeur.

Résumé

Effectuer le diagnostic automatique des réseaux de télécommunications requiert la compréhension du phénomène de propagation de pannes et d'alarmes sur ces réseaux. Cette compréhension permet d'acquérir des connaissances pertinentes nécessaires pour résoudre automatiquement le problème inverse de la propagation de pannes. Deux méthodes principales peuvent être utilisées pour comprendre la propagation de pannes et d'alarmes afin de deviner ou approximer autant que possible les causes racines des anomalies observées sous forme d'alarmes ou de paramètres hors du standard de fonctionnement nominal d'un réseau. Les systèmes experts qui formulent des lois ou des règles qui décrivent le phénomène. Les méthodes artificielles ou inspirées des sciences de l'artificiel pour lesquelles un phénomène n'est compris que s'il peut être reproduit par modélisation. Cette thèse propose une méthode générique probabiliste qui facilite la modélisation de la propagation de pannes sur les réseaux de télécommunications de grande dimension. Cette méthode basée sur le formalisme des réseaux bayésiens est appliquée pour développer un modèle de la propagation de pannes et d'alarmes sur le réseau d'accès FTTH (Fiber To The Home) de type GPON (Gigabit Passive Optical Network). Les connaissances métiers dites "expertes" du réseau GPON-FTTH sont utilisées pour construire la structure et déterminer approximativement les paramètres du modèle ainsi appelé modèle "expert" du réseau GPON-FTTH. Ce modèle est confronté à la réalité en effectuant le diagnostic des anomalies observées sur le réseau GPON-FTTH commercial de l'opérateur de télécommunications Orange. Les résultats de diagnostic sont satisfaisants puisqu'ils sont cohérents avec l'expertise métier du réseau GPON-FTTH. Nous montrons comment et pourquoi les résultats de diagnostic du modèle probabiliste sont plus cohérents avec l'expertise métier, et plus raisonnables sur un échantillon de cas de diagnostic représentatifs que le système expert à base de règles actuellement en exploitation dans le système de gestion du réseau GPON-FTTH d'Orange.

Avec l'objectif d'améliorer davantage les performances de localisation de pannes du modèle probabiliste, nous étudions et appliquons l'algorithme EM (Expectation Maximization) afin d'ajuster automatiquement les paramètres "experts" du modèle, à partir des données réelles générées par les équipements du réseau GPON-FTTH commercial d'Orange. Nous montrons que le modèle probabiliste avec les paramètres optimisés améliore raisonnablement les performances de diagnostic du modèle "expert" du réseau GPON-FTTH.

Abstract

Carrying out self-diagnosis of telecommunication networks requires an understanding of the phenomenon of fault propagation on these networks. This understanding makes it possible to acquire relevant knowledge in order to automatically solve the problem of reverse fault propagation. Two main types of methods can be used to understand fault propagation in order to guess or approximate as much as possible the root causes of observed alarms. Expert systems formulate laws or rules that best describe the phenomenon. Artificial intelligence methods consider that a phenomenon is understood if it can be reproduced by modeling. We propose in this thesis, a generic probabilistic modeling method which facilitates fault propagation modeling on large-scale telecommunication networks. A Bayesian network (BN) model of fault propagation on GPON-FTTH (Gigabit capable Passive Optical Network-Fiber To The Home) access network is designed according to the generic model. GPON-FTTH network skills are used to build structure and approximatively determine parameters of the BN model so-called expert BN model of the GPON-FTTH network. This BN model is confronted with reality by carrying out self-diagnosis of real malfunctions encountered on the commercial GPON-FTTH network of Orange telecommunication operator. Obtained self-diagnosis results are very satisfying and we show how and why the BN model improves self-diagnosis of this network compared to the rule-based expert system currently operated by Orange.

With the main goal to improve diagnostic performances of the BN model, we study and apply EM (Expectation Maximization) algorithm in order to automatically fine-tune parameters of the BN model from tremendous amount of real data generated by the commercial GPON-FTTH network of Orange. We show that the new BN model with optimized parameters reasonably improves self-diagnosis previously carried out by the expert Bayesian network model of the GPON-FTTH access network.

Chapitre 1

Introduction

« The actual science of logic is conversant at present only with things either certain, impossible or entirely doubtful... Therefore the true logic for this world is the calculus of probabilities, which takes account of the magnitude of the probability which is, or ought to be, in a reasonable man's mind. »

James Clerk Maxwell, 1850

Sommaire

1.1 Énoncé du problème	6
1.2 Contribution	7
1.2.1 Le programme de recherche NOEMI d'Orange Labs	7
1.2.2 Les contributions de la thèse	7
1.3 Structure de la dissertation	8

1.1 Énoncé du problème

Les opérateurs de télécommunication font de plus en plus d'efforts pour fournir des services de très bonne qualité à leurs abonnés. En outre, les réseaux de télécommunication doivent être fiables et robustes pour garantir également la haute disponibilité de ces services. La gestion des réseaux est de ce fait devenue un problème central pour les opérateurs de télécommunication qui développent des travaux de recherches afin d'automatiser autant que possible les opérations complexes de gestion et d'administration des réseaux, telles que la gestion de pannes. Le diagnostic de pannes est un aspect central de la gestion et de l'administration d'un réseau. Une panne se définit comme la cause racine d'une ou de plusieurs anomalies survenues sur un réseau et observées sous forme d'alarmes ou de paramètres hors du standard de fonctionnement nominal de ce réseau. Les alarmes ou symptômes sont des manifestations externes des pannes [JAKOBSON et WEISSMANI \[1993\]](#). Le but principal du diagnostic est de retrouver aussi rapidement que possible les causes racines des anomalies observées qui interrompent ou dégradent la qualité de service fournie aux abonnés.

Traditionnellement, le diagnostic de pannes sur un réseau était effectué manuellement par des experts avérés de ce réseau. Cependant, le développement des réseaux de télécommunication ces deux dernières décennies a considérablement augmenté leurs dimensions et complexifié leurs architectures. Les réseaux sont devenus des systèmes distribués et dynamiques de très grande dimension. Une panne survenue sur un réseau se propage et engendre d'autres pannes et alarmes qui à leur tour en engendrent d'autres. Ce phénomène est appelé propagation de pannes. Il s'agit d'un ensemble de relations de causes à effets entre les causes racines, les causes intermédiaires et les alarmes observées. La conséquence de la propagation de pannes et d'alarmes est qu'une seule et unique cause racine peut engendrer une cascade complexe et distribuée de plusieurs autres pannes et de leurs alarmes correspondantes [HOUNKONNOU \[2013\]](#). Ceci est d'autant plus vrai lorsque plusieurs pannes se propagent simultanément au travers de l'architecture distribuée d'un réseau de télécommunication de très grande dimension. La propagation de pannes établit une relation f entre une configuration p de pannes survenues sur le réseau à un instant $t - \delta t$ et la configuration a d'alarmes observées à l'instant t telle que $a = f(p)$ ou alors $p = f^{-1}(a)$.

Le diagnostic de pannes est devenu trop complexe pour les experts humains qui ne peuvent formuler et traiter que très peu d'hypothèses dans leurs raisonnements. En effet, les humains ont besoin d'une solide expérience pour bien comprendre le phénomène de propagation de pannes sur un réseau de grande dimension. La propagation de pannes est un phénomène très complexe dû à la nature dynamique, distribuée et non déterministe d'un réseau. Une seule panne peut déclencher plusieurs alarmes distinctes et une alarme peut être déclenchée par plusieurs causes racines distinctes. Comprendre le phénomène de propagation de pannes afin d'automatiser le diagnostic de pannes est un problème central et critique pour les opérateurs de télécommunication. En effet, la compréhension de ce phénomène permet d'acquérir des connaissances pertinentes sur le comportement du réseau et nécessaires à la résolution du problème inverse de la propagation de pannes. Diagnostiquer une anomalie observée sur un réseau revient donc à deviner ou approximer autant que possible le comportement caché f^{-1} de la propagation inverse de pannes sur ce réseau. Il s'agit de partir d'une configuration a d'alarmes observées à un instant t , de retourner dans un passé assez récent $t - \delta t$ pour trouver leurs explications causales

$p = f^{-1}(a)$. Ainsi le diagnostic consiste essentiellement à déterminer ou approximer la fonction cachée f^{-1} , c'est-à-dire trouver une application, une méthode ou un modèle h telle que les hypothèses de pannes $h(a)$, formulées sur la base de ce modèle, expliquent les anomalies observées a . Autrement dit, les hypothèses de pannes $h(a)$ sont valides si et seulement si $h(a) \simeq f^{-1}(a) = p$.

1.2 Contribution

Nous présentons dans cette section le projet de recherche dans lequel s'inscrivent les travaux menés dans cette thèse ainsi que les contributions de la thèse.

1.2.1 Le programme de recherche NOEMI d'Orange Labs

Le programme de recherche NOEMI (Network OpEration, administration, Maintenance and Integration) d'Orange Labs poursuit un objectif principal : développer des Opérations d'Administration et de Maintenance (OAM) pour la gestion flexible et automatisée des équipements et des périphériques réseau, afin de faciliter l'exploitation et la valorisation des réseaux. Cet objectif nécessite notamment le développement de nouvelles approches de déploiement et d'intégration des réseaux ainsi que les fonctionnalités associées, comprenant les aspects relatifs à la convergence fixe-mobile et à l'organisation automatisée des réseaux. Le programme NOEMI se subdivise en cinq projets de recherche parmi lesquels le projet FABOAM (Fixed Access, Backhaul and core network OAM) qui s'intéresse au développement des opérations d'administration et de maintenance des réseaux d'accès fixes, des réseaux de "backhaul" ou réseau de collecte et des coeurs de réseaux. Les travaux de cette thèse se focalisent sur le développement des opérations d'administration et de maintenance des réseaux d'accès fixes bien qu'ils soient suffisamment génériques pour être appliqués à d'autres types de réseaux.

1.2.2 Les contributions de la thèse

Nous énumérons dans cette sous-section les contributions de cette thèse dans le domaine du diagnostic des réseaux d'accès fixes s'inscrivant dans le projet de recherche FABOAM d'Orange Labs. De façon très synthétique, cette thèse propose une méthodologie facilitant la modélisation des réseaux de télécommunication pour des raisons de diagnostic, et applique cette méthode pour le diagnostic du réseau d'accès optique GPON (Gigabit Passive Optical Network) de type FTTH (Fiber To The Home). Les contributions détaillées sont les suivantes :

- Un modèle générique basé sur le formalisme des réseaux bayésiens, qui intègre les propriétés de modularité et d'extensibilité facilitant ainsi la modélisation d'un réseau de télécommunication et la reconfiguration automatique du modèle.
- L'application de la méthode générique pour construire un modèle de la topologie et de la propagation de pannes et d'alarmes sur le réseau d'accès optique GPON (Gigabit Passive Optical Network) de type FTTH (Fiber To The Home). Le modèle dit "modèle expert" puisque construit en s'appuyant sur les connaissances métiers (dites "expertes") du réseau GPON-FTTH, est composé de deux blocs fonctionnels : le réseau de transport et le réseau de connexion ou branchement des abonnés. Le réseau de distribution optique (ODN) qui constitue le troisième bloc fonctionnel du

réseau GPON-FTTH sera pris en considération un peu plus tard dans cette thèse.

- Le développement d'un outil de diagnostic appelé PANDA (Probabilistic tool for Access Network self-Diagnosis). Il s'agit d'une implémentation en Python du modèle probabiliste du réseau GPON-FTTH et de l'inférence sur ce modèle. L'outil PANDA est expérimenté pour le diagnostic de pannes sur le réseau GPON-FTTH de l'opérateur Orange France. Les performances de localisation de pannes sont comparées avec celles de l'outil DELC (Diagnostic Expert de la Ligne Client), un système expert à base de règles actuellement en exploitation dans le système de gestion du réseau GPON-FTTH d'Orange.
- L'ajustement automatique des distributions de probabilités conditionnelles du modèle "expert" par une application de l'algorithme EM (Expectation Maximization) sur un échantillon de données réelles générées par les équipements du réseau d'accès GPON-FTTH opérationnel d'Orange. Le modèle "expert" avec les paramètres ajustés par l'EM améliore les performances de diagnostic précédemment effectué avec le modèle "expert" initial.
- L'extension du modèle "expert" initial pour prendre en compte le réseau de distribution optique (ODN). Le modèle du bloc fonctionnel ODN proposé s'adapte à toutes les ingénieries de distribution optique. L'ajout du modèle de l'ODN dans le modèle "expert" initial permet d'obtenir un modèle complet pour le diagnostic de pannes sur un réseau GPON-FTTH.

1.3 Structure de la dissertation

Ce manuscrit de thèse est organisé en neuf chapitres :

- **Chapitre 1** : Il s'agit du chapitre courant qui introduit ce document tout en définissant le diagnostic dans les réseaux de télécommunication comme la résolution exacte ou approchée du problème inverse de la propagation de pannes et d'alarmes sur ces réseaux.
- **Chapitre 2** : l'état de l'art dans le domaine du diagnostic des réseaux est présenté dans ce chapitre. Deux approches principales sont exposées : (1) les systèmes experts à base de règles qui formulent des règles ou des lois pour décrire et comprendre le phénomène de propagation de pannes, (2) les approches inspirées des sciences de l'artificiel qui considèrent qu'un phénomène n'est compris que lorsqu'il peut être reproduit ou simulé. Ces approches prennent de la distance par rapport au phénomène de propagation de pannes et essaient de l'approcher dans son comportement. Dans cette dernière catégorie se rangent les méthodes de diagnostic à base de modèle qui développent des raisonnements sur un modèle ou une représentation explicite de la structure et du fonctionnement du réseau, les méthodes qui essaient d'apprendre artificiellement le comportement du réseau sans le modéliser. L'apprentissage se fait dans ce cas à partir des données générées par les équipements du réseau à diagnostiquer. Ces études nous ont permis de choisir objectivement le formalisme basé sur les réseaux bayésiens autour duquel nous

avons développé une méthode générique de modélisation d'un réseau de télécommunication pour des raisons de diagnostic.

- **Chapitre 3 :** Les algorithmes de raisonnement ou d'inférence probabiliste exacte sur un réseau bayésien sont étudiés dans ce chapitre. Les réseaux bayésiens ayant une topologie en arbre ou réseaux non connectés, en polyarbre ou réseaux simplement connectés et les réseaux avec des boucles acycliques ou réseaux multi connectés sont tous considérés.
- **Chapitre 4 :** Nous étudions l'apprentissage artificiel des paramètres d'un réseau bayésien grâce à la méthode du maximum de vraisemblance (MLE) d'un échantillon de données complet ou incomplet. Les bases de l'estimation au sens du maximum de vraisemblance sont tout d'abord rappelées dans le cas d'un échantillon de données complet avant d'étudier dans les détails l'algorithme EM (Expectation Maximization) dans le cas d'un échantillon incomplet. Ce dernier cas est d'ailleurs celui qui nous intéresse particulièrement puisque les données générées par les équipements d'un réseau de télécommunication ne sont presque jamais complètes, c'est-à-dire qu'il existe presque toujours des variables manquantes ou cachées.
- **Chapitre 5 :** Nous présentons ici une méthode générique qui facilite la modélisation de la topologie et du comportement (propagation de pannes) d'un réseau de télécommunication. C'est un modèle en trois couches, basé sur les réseaux bayésiens, qui intègre les propriétés de modularité et d'extensibilité lui permettant de se reconfigurer et de s'adapter facilement aux éventuels changements sur la topologie et le comportement du réseau de télécommunication.
- **Chapitre 6 :** La méthode générique est appliquée pour construire un modèle de la topologie et de la propagation de pannes et d'alarmes sur le réseau GPON-FTTH. Ce modèle dit "modèle expert" puisque construit en s'appuyant essentiellement sur les connaissances métiers du réseau GPON-FTTH est utilisé pour faire le diagnostic de pannes sur ce réseau d'accès fixe. Ce modèle est doublement validé. Une première validation est faite sur la plateforme expérimentale du réseau GPON-FTTH d'Orange Labs. Une seconde validation est faite sur le réseau GPON-FTTH opérationnel de l'opérateur de télécommunication Orange. Les résultats de cas réels de diagnostic sont comparés avec ceux du système expert de diagnostic à base de règles en exploitation chez Orange.
- **Chapitre 7 :** Ce chapitre présente une implémentation en Python de l'algorithme EM pour ajuster automatiquement les paramètres du modèle "expert" du réseau GPON-FTTH. Cette implémentation utilise les données réelles générées par les équipements du réseau GPON-FTTH de l'opérateur de télécommunication Orange. Nous discutons des difficultés rencontrées lors de la manipulation des données réelles par l'algorithme EM et des solutions proposées. La valeur ajoutée de l'EM est évaluée en comparant les précédents résultats de diagnostic du modèle "expert" aux résultats de diagnostic obtenus avec le modèle du réseau GPON-FTTH dont les paramètres ont été ajustés automatiquement par l'algorithme EM.
- **Chapitre 8 :** Le modèle "expert" initial du réseau GPON-FTTH est composé de deux

blocs fonctionnels. Le modèle du réseau de transport en amont et celui du réseau de connexion des abonnés en aval. Dans ce chapitre nous proposons une extension du modèle "expert" initial pour prendre en compte la structure du réseau de distribution (ODN). Nous proposons un modèle de l'ODN qui s'adapte à toutes les ingénieries de distribution optique. L'ajout du modèle de l'ODN dans le modèle "expert" initial nous donne un modèle complet de diagnostic du réseau GPON-FTTH.

- **Chapitre 9** : Ce chapitre conclut les travaux menés dans le cadre de cette thèse. Plusieurs travaux futurs y sont également envisagés.

Chapitre 2

État de l'art dans le domaine du diagnostic des réseaux de télécommunication

« The theory of probabilities is at bottom nothing but common sense reduced to calculus; it enables us to appreciate with exactness that which accurate minds feel with a sort of instinct for which oftimes they are unable to account. »

Pierre Simon Laplace, 1819

Sommaire

2.1 Introduction	12
2.2 Système expert à base de règles	12
2.3 Système expert à base de modèles	17
2.4 Le diagnostic comme système d'apprentissage artificiel	21
2.4.1 Science naturelle et science de l'artificiel	21
2.4.2 Apprentissage artificiel par induction	22
2.4.3 Raisonnement à base de cas	23
2.4.4 Réseaux de neurones	27
2.5 Réseau de Petri	29
2.6 Graphe de dépendances	31
2.7 Réseaux bayésiens	35
2.8 Discussions et conclusion	41

2.1 Introduction

Ce chapitre présente l'état de l'art dans le domaine du diagnostic des réseaux de télécommunication. Deux approches principales sont exposées. Les systèmes experts à base de règles qui formulent des règles ou des lois pour décrire et comprendre le phénomène de propagation de pannes et d'alarmes sur un réseau de télécommunication. Les approches inspirées des sciences de l'artificiel qui considèrent qu'un phénomène n'est compris que lorsqu'il peut être reproduit ou simulé. Ces dernières approches prennent de la distance par rapport au phénomène de propagation de pannes et essaient de l'approcher dans son comportement. Dans cette catégorie se rangent les méthodes de diagnostic à base de modèle qui développent des raisonnements sur une représentation explicite de la structure et du fonctionnement du réseau et les méthodes qui essaient d'apprendre artificiellement le comportement du réseau sans le modéliser. L'apprentissage se fait dans ce cas à partir des données générées par les équipements du réseau à diagnostiquer, et consiste essentiellement à résoudre par interpolation ou induction le problème inverse de la propagation de pannes et d'alarmes sur un réseau de télécommunication.

2.2 Système expert à base de règles

Un système expert de diagnostic est une suite d'applications informatiques qui intègrent une grande base de connaissances ou de raisonnements d'experts sur des tâches de diagnostic bien précises et qui infèrent de façon automatique les causes racines des anomalies observées. Pour constituer une base de raisonnements valides, les connaissances d'experts en matière de diagnostic peuvent être formalisées sous forme de règles, d'arbre de décisions, de logique propositionnelle, etc. Une évaluation de différents schémas de représentation des connaissances est faite dans [PAU \[1986\]](#). La complexité des systèmes experts se situe principalement au niveau de la représentation formelle des connaissances spécialisées [HOUNKONNOU \[2013\]](#). Un système expert se distingue d'un logiciel classique. En effet, un logiciel classique est développé autour d'un ensemble de procédures algorithmiques. La résolution d'un problème suit une séquence d'étapes bien définie par le programmeur qui doit de ce fait connaître toutes les combinaisons possibles des données d'entrée du logiciel et les traitements associés à ces états. Un système expert quant à lui n'est pas construit autour d'un ensemble de procédures prédéfini. En effet, un système expert peut intégrer la capacité à déterminer lui même les traitements adaptés à un état donné des paramètres d'entrée, c'est-à-dire une séquence d'étapes non prédéfinie par le programmeur pour cet état. Cette différence entre un logiciel classique et un système expert est essentiellement due à la méthode d'organisation et d'utilisation des connaissances spécialisées.

Un programme traditionnel est organisé en deux niveaux : les données et le contrôle [CRONK et collab. \[1998\]](#), c'est-à-dire le code qui résout le problème. La reconnaissance des tâches et la méthode de résolution du problème sont explicitement codées par le programmeur. Il est donc très difficile d'ajouter une nouvelle connaissance sur une tâche donnée ou de changer la méthode de résolution d'un problème complexe. Un système expert quant à lui effectue une séparation entre la reconnaissance des tâches et la résolution du problème. Cette séparation lui confère une architecture logicielle en trois niveaux : les données ou faits en mémoire (working memory), les connaissances sur les tâches (task knowledge ou knowledge base) et le moteur d'inférence (control) [CRONK et collab. \[1998\]](#). Une inférence est une conclusion déduite sur la base de faits ou d'hypothèses. Cette archi-

tecture réduit considérablement les difficultés relatives à la construction et à la modification des programmes informatiques complexes. La Figure 2.1 présente cette architecture.

Au niveau "control", le moteur d'inférence est un programme qui détermine comment appliquer les connaissances contenues dans la "knowledge base" aux faits et hypothèses courants décrits dans la "working memory" afin d'inférer de nouvelles données qui pourront être utilisées pour de prochaines inférences. Une donnée est un ensemble d'attributs et de valeurs associées qui décrivent explicitement un fait. Pour des faits correspondant à une situation donnée, le moteur d'inférence détermine dans un premier temps les règles potentielles applicables. Dans un second temps, le moteur d'inférence détermine parmi ces règles candidates la règle la plus appropriée à la situation courante. Une règle experte est exprimée sous la forme IF *< condition >* THEN *< action >*.

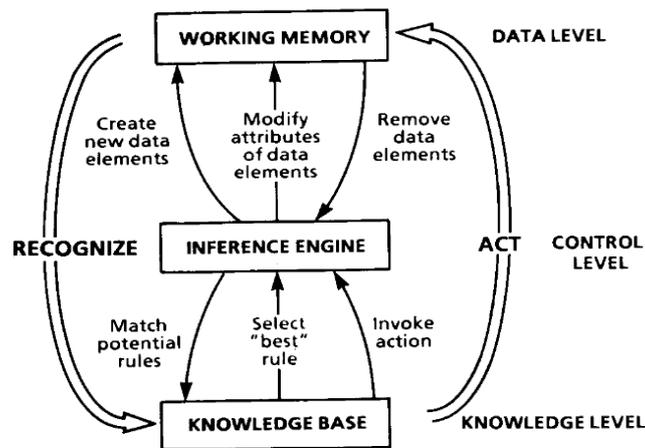


FIGURE 2.1 – Organisation des systèmes experts à base de règles [CRONK et collab. \[1998\]](#).

La condition peut être évaluée directement si les données ou les hypothèses requises sont disponibles. Certains systèmes experts automatisent des tests sur le domaine expertisé tel qu'un réseau de télécommunication par exemple, et les résultats servent à enrichir les faits et donc mieux évaluer les conditions. L'action peut être une décision définitive ou la création de nouveaux attributs de données, la modification ou la suppression de certaines données dans la mémoire de travail (working memory). Le mécanisme de contrôle implémenté par le moteur d'inférence est un cycle de reconnaissance de faits et d'activation de règles. Ce cycle exécute les trois étapes suivantes [CRONK et collab. \[1998\]](#) :

- Le "matching" ou la recherche de toutes les règles applicables au contenu courant de la mémoire de travail (working memory). Cet ensemble de règles candidates est appelé "conflict set".
- La sélection ou la résolution du conflit détermine parmi les règles concurrentes, laquelle est la plus appropriée pour être invoquée à ce moment, c'est-à-dire la règle qui correspond au mieux aux données dans la mémoire de travail.
- L'invocation ou l'exécution de la règle précédemment choisie qui consiste à appliquer les actions spécifiées par cette règle. Ces actions changent typiquement les données dans la "working memory". Dans certains systèmes experts, des règles elles-mêmes sont ajoutées dans la "knowledge base", supprimées ou modifiées.

Le cycle est répété jusqu'à ce qu'aucune règle correspondant au contenu de la mémoire de travail ne soit trouvée (ou une interruption externe du système). Par conséquent, le mécanisme de contrôle est la répétition de l'évaluation des conditions des règles basée

sur des données changeantes et non pas sur une structure statique du programme. Ce mécanisme qui n'est pas utilisé dans les logiciels traditionnels est appelé "data-driven control" ou "forward-chaining".

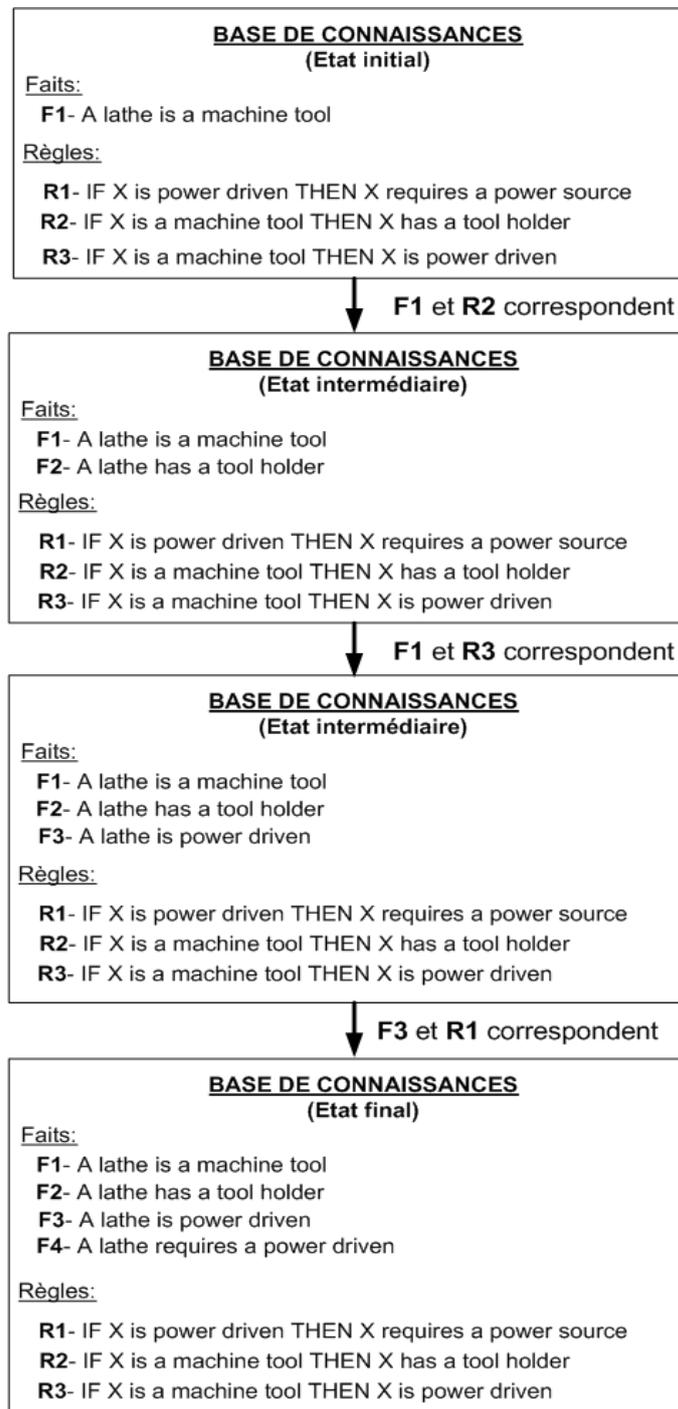


FIGURE 2.2 – Un exemple de l'algorithme "forward chaining" PHAM [1988].

Le raisonnement d'un système expert repose essentiellement sur la procédure d'inférence qui peut varier d'un système à l'autre. Avec l'algorithme de chaînage avant (forward chaining), le système essaye de faire correspondre les faits disponibles dans la mémoire de travail, avec la portion IF des règles dans la base des connaissances (knowledge base) PHAM [1988]. Lorsque de telles règles sont trouvées (l'ensemble des règles concurrentes), l'une d'entre elles est sélectionnée sur la base d'un critère discriminant approprié, et exé-

cutée. L'exécution génère de nouveaux attributs de données qui enrichissent les faits en mémoire dont le contenu sera utilisé à la prochaine itération pour de nouveau déterminer un ensemble de règles concurrentes et sélectionner celle d'entre elles qui sera exécutée. Les itérations se terminent lorsqu'aucune règle ne peut plus être exécutée, c'est-à-dire qu'aucune règle ne correspond au contenu courant de la mémoire ou lorsqu'il est impossible de discriminer entre les règles concurrentes. La Figure 2.2 présente un exemple illustratif de l'algorithme "forward chaining" (data-driven inferencing). Dans l'exemple présenté par la Figure 2.2, chacune des portions IF des règles R2 et R3 correspond au fait F1 en instanciant la variable X à "lathe". Le critère discriminant adopté dans cet exemple est "first come, first served". La règle R2 est donc sélectionnée et exécutée puisqu'elle est rencontrée avant R3 dans la base de connaissances. Un autre critère discriminant tel que la "priorité" ou la "récence" aurait pu être appliqué pour résoudre le conflit entre les règles concurrentes.

Notons aussi qu'il existe d'autres procédures d'inférence comme alternative au chaînage avant. On peut noter par exemple le chaînage arrière (backward chaining ou goal-driven inferencing) qui consiste à prouver une assertion appelée "objectif" PHAM [1988]. Si l'assertion ne peut être directement prouvée par les faits disponibles dans la base des connaissances, le système examine les règles pour lesquelles la portion THEN contient l'assertion. En effet, pour chacune de ces règles, le système vérifie s'il existe un fait dans la base correspondant à la portion IF. Si un tel fait est disponible, alors la règle est exécutée. Dans le cas contraire, le fait dérivant de la portion IF de la règle est considéré comme une autre assertion (appelée "sous-objectif") à prouver. Le processus se poursuit récursivement jusqu'à ce que tous les faits requis pour prouver l'assertion initiale sont trouvés, ou qu'une assertion correspondant à un "sous-objectif" ne peut être prouvée. Dans ce dernier cas, l'assertion initiale est elle-même désapprouvée.

L'efficacité de la procédure d'inférence d'un système expert est étroitement liée à la pertinence des connaissances acquises et à leur représentation. Les difficultés relatives à la conception des systèmes experts sont justement l'acquisition et la représentation des connaissances. En effet, il est difficile de formaliser le problème à résoudre lors de la conception d'un système expert DAVID et KRIVINE [1987]. Pourtant il faut que le problème soit bien formalisé afin de faciliter l'extraction des connaissances pertinentes de l'énorme quantité d'informations ou de données collectées durant la période d'expertise. L'extraction des connaissances peut s'avérer très complexe si les informations et les données collectées contiennent beaucoup d'incohérences ou de redondances qu'il faut pouvoir identifier et supprimer, mais aussi beaucoup d'insuffisances ou d'absences d'informations qu'il faut pouvoir compléter. Enfin, nous noterons comme conséquence des difficultés précitées qu'il est très difficile de valider un système expert. Néanmoins, malgré ces difficultés, des systèmes experts ont été conçus dans le passé avec plus ou moins de succès.

DENDRAL est le tout premier système expert développé en 1965, par les informaticiens Edward Feigenbaum, Bruce Buchanan, le médecin Joshua Lederberg et le chimiste Carl Djerassi, à l'université de Stanford, pour permettre à la NASA (National Aeronautics and Space Administration) d'effectuer des analyses chimiques du sol de la planète MARS durant les missions spatiales. L'objectif principal de DENDRAL était d'aider les chimistes organiques à l'identification des molécules organiques inconnues, par analyse de leurs spectres de masse et en utilisant les connaissances en chimie.

Le programme DENDRAL est constitué de deux sous-programmes : Heuristic Dendral et Meta Dendral.

- Heuristic Dendral est un programme qui utilise des spectres de masse ou d'autres données expérimentales conjointement avec une base de connaissances en chimie, pour produire un ensemble de structures chimiques possibles. Le spectre de masse d'un composé est produit par un spectromètre de masse, et est utilisé pour déterminer son poids moléculaire, c'est-à-dire la somme des masses de ses constituants atomiques. Par exemple, le composé eau (H_2O), a une masse moléculaire de 18 puisque l'hydrogène a une masse de 1,01 et de l'oxygène 16,00, et son spectre de masse présente un pic à 18 unités. Heuristic Dendral utilise cette masse d'entrée et la connaissance du tableau périodique des éléments chimiques ainsi que des règles de valence, pour déterminer les combinaisons possibles des constituants atomiques dont la somme des masses atomiques serait 18. Lorsque la masse moléculaire augmente et que les molécules deviennent plus complexes, le nombre de composés possibles augmente de manière drastique. Ainsi, un programme qui est en mesure de réduire ce nombre de solutions candidates au moyen d'un processus de formation d'hypothèses est essentiel, d'où le rôle de Meta Dendral.
- Meta Dendral est un système d'acquisition de connaissances qui reçoit en entrée, l'ensemble des structures chimiques possibles (produites par Heuristic Dendral) et les spectres de masse correspondants, et propose un ensemble d'hypothèses pour expliquer la corrélation entre certaines des structures proposées et le spectre de masse de la molécule inconnue à identifier. Ces hypothèses sont ensuite renvoyées à Heuristic Dendral pour tester leur applicabilité. Ainsi, Heuristic Dendral est un système de performance et Meta Dendral est un système d'apprentissage.

Lor et collab. [LOR et KAR-WING \[1993\]](#) ont développé un système de diagnostic des réseaux de multiplexeurs. Les connaissances expertes de diagnostic sont classifiées en deux catégories : les connaissances expertes génériques et les connaissances expertes spécialisées sur des tâches de diagnostic bien précises. Le système expert utilise une base de données contenant des informations statiques et des informations dynamiques nécessaires durant le processus de diagnostic. Ces informations concernent les relations entre les entités logiques (groupes de canaux) et les entités physiques (équipements et liens) telles que les informations de routage, les attributs des entités physiques et les relations d'incidences entre les noeuds du réseau et les liens, etc. Le diagnostic est un processus interactif entre l'expertise générique et l'expertise spécialisée. La survenance d'une panne spécifique invoque une règle spécialisée, qui à son tour, invoque une règle générique. Les règles génériques produisent un diagnostic général et un plan de réparation pour cette panne. Les règles spécialisées sont invoquées pour appliquer ce plan. Plusieurs échanges entre la base de connaissances expertes génériques et celle des connaissances expertes spécialisées sont nécessaires dans une session de diagnostic d'une anomalie survenue sur le réseau.

DELC (Diagnostic Expert de la Ligne Client) est un système expert à base règles développé par Orange Labs France pour le diagnostic automatique des réseaux d'accès cuivre xDSL (Digital Subscriber Line) et optique GPON (Gigabit passive Optical Network) de type FTTH (Fiber To The Home). Le diagnostic d'une ligne se fait en deux étapes. On commence d'abord par collecter les données disponibles de cette ligne, telles que les niveaux de puissances transmises et reçues par les équipements de cette ligne, les tensions d'alimentation, les courants de polarisation, les températures des équipements de cette ligne,

les compteurs d'erreurs de transmission, les alarmes observées, etc. Chaque donnée est stockée dans une KPI (Key Performance Identifier). Ensuite des règles expertes prédéfinies utilisent ces KPIs pour produire une indication ou une décision finale de diagnostic appelée conclusion. Une règle n'est invoquée et exécutée par le moteur d'inférence que si toutes les KPIs utilisées par cette règle, c'est à dire les KPIs présentés dans la portion IF de la règle, ont été renseignées lors de la phase de collecte des données de la ligne. La reconnaissance des faits (combinaisons de valeurs de plusieurs KPIs) et le déclenchement des règles sont assurés par le programme Drools développé par la communauté JBoss, qui fournit un moteur d'inférence. L'outil DELC est actuellement en exploitation dans le système de gestion du réseau GPON-FTTH de l'opérateur Orange France. L'objectif opérationnel de cette thèse pour Orange, est de développer une nouvelle approche de diagnostic qui doit non seulement améliorer les performances de localisation de pannes de DELC mais aussi requérir beaucoup moins de maintenance en plus d'être capable de s'adapter facilement aux évolutions des ingénieries du réseau GPON-FTTH.

Un système expert est un outil capable de reproduire les mécanismes cognitifs d'un expert humain, dans un domaine particulier. Il s'agit de l'une des voies tentant d'aboutir à l'intelligence artificielle. Plus précisément, un système expert est un logiciel capable de répondre à des questions, en effectuant un raisonnement à partir de faits et de règles connus dans le cas d'un système expert à base de règles. Lorsque la dimension du domaine étudié est très grande, l'acquisition des connaissances nécessaires pour constituer les faits et formuler des règles peut s'avérer très difficile et fastidieuse. En effet, il faut pouvoir agréger les connaissances de plusieurs experts, exploiter de nombreux documents et parfois même effectuer des tests expérimentaux pour lever certaines ambiguïtés. Plusieurs cycles entre l'acquisition de connaissances, leur représentation en un langage descriptif comme des règles, l'implémentation du système et des tests peuvent être nécessaires pour obtenir un système correct [LEWIS \[1993\]](#). L'avantage principal d'un système expert est qu'il est très performant pour résoudre les problèmes rencontrés durant la période d'expertise pour lesquels des règles ont été formulées. Cependant, pour un domaine de grande dimension, le nombre de règles augmente drastiquement et il devient très difficile d'assurer leur maintenance. En effet, il faut pouvoir poursuivre l'expertise du domaine étudié, formuler de nouvelles règles et les corrélérer manuellement avec toutes les règles existantes. Cet inconvénient des systèmes experts à base de règles est connu sous l'anglicisme "knowledge acquisition bottleneck". Un système expert est donc très adapté pour des domaines très peu changeants. Par contre, si le domaine est très dynamique, certaines règles expertes peuvent devenir très rapidement obsolètes et fragiliser le système qui deviendra incapable de résoudre certains problèmes. Ces faiblesses des systèmes experts à base de règles ont conduit au développement d'une nouvelle approche de représentation des connaissances expertes.

2.3 Système expert à base de modèles

Cette seconde classe de systèmes experts n'utilise plus une base de raisonnements spécialisés sur des tâches précises de diagnostic, mais des connaissances sur le comportement ou le fonctionnement du réseau de télécommunications. Ces connaissances sont utilisées pour construire une représentation structurée et explicite du fonctionnement du réseau. Il s'agit d'une description des équipements, de la topologie, de l'architecture et surtout du processus de propagation d'évènements et de la corrélation d'évènements [JAKOBSON et WEISSMAN \[1995\]](#). On prend en compte la capacité d'une panne sur une entité

du réseau à changer l'état d'autres entités [MALGORZATA et ADARSHPAL \[2001\]](#). La structure du réseau et son comportement sont donc modélisés [GARDNER et HARLE \[1997\]](#). La complexité de développement d'un système de diagnostic à base de modèle est due au fait qu'un réseau de télécommunications de grande échelle est très souvent hétérogène avec un grand nombre d'équipements de types et de fabricants différents. Les composants logiciels sont aussi de différents types (protocoles, services, applications, etc.). La topologie peut changer et certains protocoles peuvent être dynamiques [HOUNKONNOU \[2013\]](#). Les dépendances dynamiques entre équipements ou composants logiciels rendent également dynamique la propagation de pannes.

La construction du modèle n'est que la première étape du développement d'une approche de diagnostic d'un réseau à base d'un modèle de ce réseau. La seconde étape consiste à développer ou à appliquer un algorithme de raisonnement sur le modèle. L'algorithme de raisonnement commence par les entités ayant déclenché des alarmes, et explore les relations entre les entités du réseau formalisées par le modèle [MALGORZATA et ADARSHPAL \[2004\]](#). L'algorithme est ainsi capable de déterminer quelles alarmes sont corrélées et localiser ainsi les entités incriminées du réseau.

De nombreuses techniques à base de modèle proposées dans la littérature utilisent une représentation orientée objet du réseau à diagnostiquer [HOUCK et collab. \[1995\]](#). Jordaan et Paterok [JORDAAN et PATEROKL \[1993\]](#) ont présenté un algorithme de corrélation d'évènements basé sur l'exploration d'un modèle générique orienté objet du réseau. Ce modèle générique est lui-même basé sur le "framework" OSI (Open Systems Interconnection) de gestion des réseaux (OSI Management Framework). L'exploration commence par l'objet sur lequel une anomalie est observée. Tous les objets qui lui sont reliés sont inspectés pour localiser la cause racine. L'inspection des objets est faite dans un ordre arbitraire. Houck et collab. [HOUCK et collab. \[1995\]](#) ont également proposé un système de corrélation d'alarmes basé sur un modèle orienté objet de la couche physique des réseaux de télécommunication.

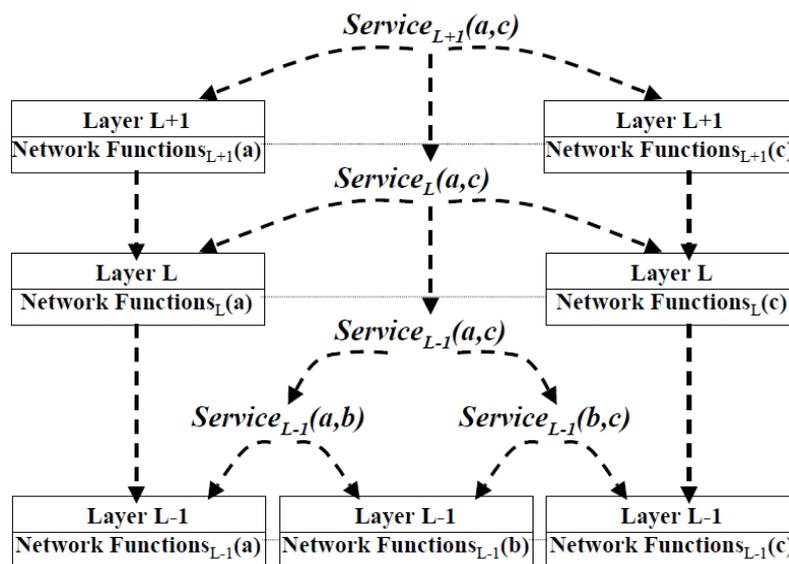


FIGURE 2.3 – Modèle en couche des dépendances entre services et fonctions d'un réseau [MALGORZATA et ADARSHPAL \[2001\]](#).

Une approche très intéressante proposée dans [GOPAL \[2000\]](#) s'inspire du modèle OSI

(Open Systems Interconnection) pour développer un modèle en couche des dépendances entre les protocoles, services et fonctions d'un réseau. Dans [GOPAL \[2000\]](#), un service fourni par un protocole de la couche L , entre les noeuds \mathbf{a} et \mathbf{b} ($Service_L(a, b)$), est implémenté par les fonctions de la couche L sur les entités \mathbf{a} et \mathbf{b} ($NetworkFunctions_L(a)$ et $NetworkFunctions_L(b)$), et les protocoles de la couche L au travers desquelles les noeuds \mathbf{a} et \mathbf{b} communiquent. Les protocoles de la couche L s'exécutant entre les noeuds \mathbf{a} et \mathbf{b} utilisent les fonctions de la couche $L - 1$ sur les noeuds \mathbf{a} et \mathbf{b} . Les fonctions de la couche L sur le noeud \mathbf{a} dépendent des fonctions de la couche $L - 1$ sur le noeud \mathbf{a} . Le modèle est donc un graphe de dépendances récursives entre services, protocoles et fonctions. Dans [MALGORZATA et ADARSHPAL \[2001\]](#), ce modèle est simplifié en supposant que les protocoles sont correctement implémentés. Les noeuds du graphe relatifs aux protocoles sont donc éliminés. La Figure 2.3 présente le modèle en couche simplifié dans lequel $Service_L(a, c)$ dépend directement de $Service_{L-1}(a, c)$.

Dans un tel modèle en plusieurs couches, il est possible d'appliquer un raisonnement de couche en couche. Dans un premier temps, une recherche horizontale est faite dans la couche dans laquelle une anomalie a été observée [KÄTKER et PATEROK \[1997\]](#). Lorsque le composant défectueux de cette couche est localisé, une recherche verticale déplace le processus de localisation de pannes à la couche adjacente inférieure. Dans cette couche inférieure, une recherche horizontale est initiée et ainsi de suite. Dans NetFACT [HOUCK et collab. \[1995\]](#), le diagnostic est effectué en deux phases. Premièrement, lors de la recherche horizontale, des votes sont assignés aux composants potentiellement incriminés en se basant sur le nombre de symptômes relatifs à ces composants. La cause racine est identifiée dans la seconde phase de recherche, qui détermine si le composant qui a reçu le plus grand nombre de votes dans la première étape était la cause racine ou si l'anomalie observée sur ce composant est une conséquence de la panne d'un autre composant dont il dépend.

Lorsque l'on réalise un modèle, il est important de sélectionner un niveau d'abstraction approprié. En effet, d'une part, le niveau d'abstraction doit être suffisamment élevé afin de limiter la complexité du système de gestion des pannes. D'autre part, l'isolation des pannes est un processus qui requiert des connaissances et des informations détaillées relatives aux ressources du réseau ainsi qu'à leurs comportements. Pour faire face à ces exigences contradictoires, un modèle générique hiérarchique est proposé dans [KÄTKER et GEIHS \[1997\]](#). Ce modèle générique est présenté par la Figure 2.4 et est inspiré de l'architecture logique en quatre couches du standard TMN (Telecommunication Management Network) de la recommandation M.3010 de l'ITU (International telecommunication Union). Ce "framework" définit quatre couches logiques pour la gestion des réseaux : (1) la couche "Business" qui inclut les fonctions relatives aux aspects commerciaux tels que la facturation, l'analyse des tendances boursières et d'autres rapports financiers, (2) la couche "Service" définit et administre les services, (3) la couche "Network" distribue les ressources réseaux et effectue les tâches de configuration, de contrôle et de supervision du réseau, (4) la couche "Element" manipule individuellement les éléments du réseau incluant la gestion des alarmes des composants matériels et logiciels.

Chaque couche dans l'architecture TMN dispose d'un système de gestion de pannes spécifique aux exigences des composants et des fonctions de la couche [KÄTKER et GEIHS \[1997\]](#). Dans la couche "Element" par exemple, les outils de diagnostic des composants matériels spécifiques fournis par les fabricants peuvent être appliqués. Le modèle de dé-

pendances de services (Abstract Service Dependency Model), qui constitue la couche supérieure de l'architecture (voir la Figure 2.4), fournit une vue intégrée et abstraite du système de diagnostic dans sa totalité. Ce modèle abstrait est l'intégrateur et le coordinateur des modèles de gestion des pannes spécifiques aux différentes couches inférieures qui sont supposées indépendantes les unes des autres. Les symptômes agrégés sont transférés des systèmes de gestion des pannes spécifiques aux couches inférieures vers le système de gestion de pannes intégré. Le système intégrateur et coordinateur est capable de corréler les symptômes agrégés des différentes couches inférieures d'une part, et de déclencher le diagnostic dans une couche inférieure spécifique. Par exemple, si la couche "Network" reporte une anomalie de connectivité due à un dysfonctionnement sur un switch, la couche supérieure sollicite le modèle de diagnostic de la couche "Element" pour diagnostiquer ce switch dans les détails.

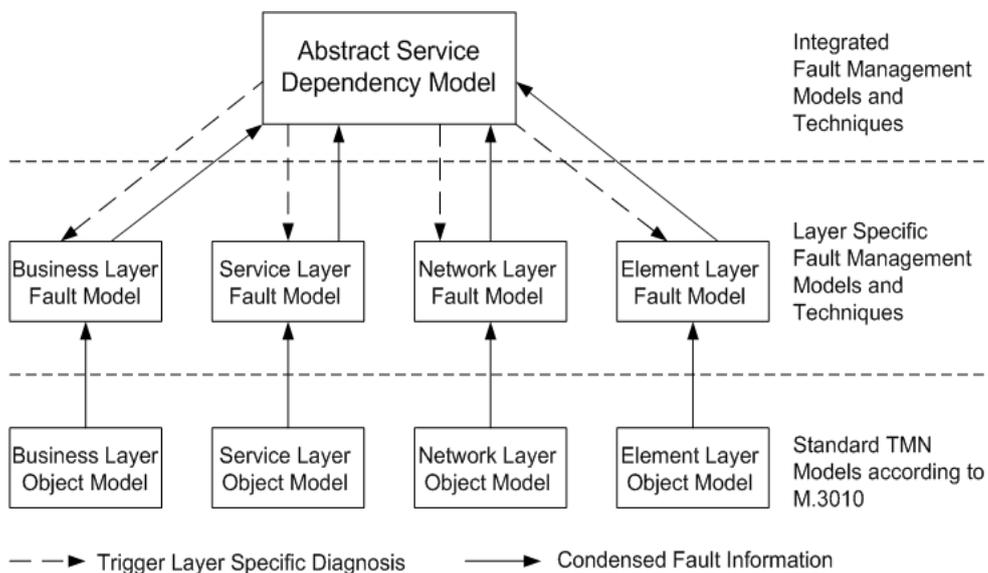


FIGURE 2.4 – Modèle hiérarchique basé sur l'architecture TMN KÄTKER et GEIHS [1997].

Dans FROHLICH et collab. [1997], est introduite une méthode à base d'un modèle pour la corrélation d'alarmes dans les réseaux de téléphones cellulaires. Ce modèle appelé SD (System Description) consiste en un ensemble d'axiomes caractérisant le fonctionnement des composants du réseau. La topologie du réseau est modélisée par un ensemble de faits. Un ensemble de formules logiques basées sur des prédicats est utilisé pour spécifier les alarmes et leurs types. Un autre ensemble de formules logiques spécifie le comportement des alarmes ainsi que leur propagation. La corrélation des alarmes repose sur le langage de calcul de prédicats ainsi défini qui permet de définir de nouveaux prédicats grâce aux formules logiques. Nous noterons aussi que cette méthode sépare les connaissances relatives à la topologie du réseau de celles relatives à son fonctionnement, facilitant ainsi les changements sur la topologie du réseau.

D'autres méthodes de diagnostic à base de modèle construisent le modèle de façon semi-automatique. Ce concept est appelé "self-modeling". Dans HOUNKONNOU [2013], une approche de modélisation semi-automatique est développée, basée sur des motifs ou "patterns" qui décrivent de façon générique les dépendances entre les ressources utilisées par les services du réseau IMS (IP Multimedia Subsystem). Un motif est un graphe de dépendances acyclique probabiliste (réseau bayésien), manuellement construit, mais

automatiquement instancié lorsqu'une panne survient dans un service IMS quelconque. Une approche similaire de modélisation semi-automatique est proposée dans [SANCHEZ et collab. \[2015\]](#), appliquée au contexte des réseaux SDN (SoftWare Defined Network) et NFV (Network Functions Virtualization). Elle est basée sur des motifs qui modélisent les composants d'un réseau SDN. Un algorithme interprète la topologie courante donnée par le contrôleur SDN et un motif, basé sur un réseau bayésien, est instancié automatiquement avec d'éventuelles dépendances en plus selon la topologie.

L'approche à base de modèle est facile à déployer et à modifier, et est appropriée pour un réseau à grande échelle si les informations relatives aux ressources du réseau sont disponibles [MIYAZAWA et NISHIMURA \[2011\]](#). Cette méthode peut diagnostiquer un plus grand nombre de pannes que les méthodes expertes à base de règles, elle peut diagnostiquer les problèmes qui sortent de son expertise même si ses performances se dégradent dans ces cas-là. Les explications causales des anomalies sont intelligibles puisque chaque étape du diagnostic peut être comprise. Le modèle structurel et fonctionnel du réseau peut être conçu de façon modulaire ou incrémentale facilitant ainsi sa mise à jour lorsque de nouvelles connaissances sur le fonctionnement du réseau sont acquises. Par contre, il est assez difficile de construire un modèle proche de la réalité structurelle et fonctionnelle du réseau tout en gardant un niveau d'abstraction suffisamment élevé pour rendre le modèle indépendant des différentes techniques d'ingénierie mises en oeuvre dans un réseau de télécommunications de grande échelle. Le modèle obtenu n'est qu'une image figée du réseau et peut donc devenir rapidement obsolète si le réseau est très dynamique. Pour pallier les insuffisances des systèmes experts à base de règles et des systèmes experts à base de modèles, de nouvelles méthodes de diagnostic inspirées de l'apprentissage artificiel ont été proposées par la communauté de l'intelligence artificielle.

2.4 Le diagnostic comme système d'apprentissage artificiel

Effectuer le diagnostic d'un réseau de télécommunication requiert la compréhension du phénomène de propagation de pannes et d'alarmes dans ce réseau. Cette compréhension permet d'acquérir des connaissances pertinentes afin de résoudre automatiquement le problème inverse de la propagation de pannes et d'alarmes.

2.4.1 Science naturelle et science de l'artificiel

De façon très générale, deux principales approches peuvent être utilisées pour comprendre un phénomène. Les méthodes dites naturelles ou inspirées des sciences naturelles qui formulent des lois sous-adjacentes ou des règles, de préférence simples, qui décrivent le phénomène [CORNÉJOLS et MICLET \[2013\]](#). Les méthodes artificielles ou inspirées des sciences de l'artificiel pour lesquelles la compréhension d'un phénomène doit passer par la construction de modèles capables de reproduire ou de simuler le phénomène. Ainsi, selon le point de vue des sciences de l'artificiel, comprendre implique la capacité de fabriquer pour reproduire [CORNÉJOLS et MICLET \[2013\]](#). Connaître dans cette optique, c'est concevoir un modèle opératoire du monde pour le soumettre à des manipulations réglées. Connaître c'est donc prendre de la distance par rapport à un phénomène et se donner les moyens de l'approcher dans son comportement.

Un système expert à base de règles est une méthode dite naturelle puisque la connaissance du domaine expertisé est un ensemble de règles qui décrivent le domaine en ques-

tion. La fragilité ou la moindre flexibilité des systèmes experts à base de règles ou leur incapacité à réagir aux anomalies inconnues de leur base de connaissances est une conséquence de leur raisonnement au cas par cas, c'est-à-dire de leur capacité de généralisation très limitée, voire même inexistante, dans certains cas. Les approches de diagnostic à base de modèle quant à elles sont artificielles puisqu'elles essaient de reproduire le phénomène de propagation de pannes et d'alarmes en le modélisant. De ce fait, la portée du raisonnement sur le modèle est nettement plus étendue et permet de couvrir plus de cas de diagnostic que le système expert à base de règles. Cependant, les performances du modèle peuvent se dégrader pour certaines situations s'écartant assez considérablement de l'expertise encodée dans le modèle.

En science de l'artificiel, la capacité de généraliser, c'est-à-dire de tirer des lois de portée générale en partant de l'observation des cas particuliers, s'appelle l'induction [CORNÉJOLS et MICLET \[2013\]](#). Pour diagnostiquer un très grand nombre de signatures d'anomalies qui pourraient survenir sur un réseau de télécommunications, le système de diagnostic doit être un système apprenant, c'est-à-dire doté des capacités d'induction qui lui permettront d'utiliser sa base de connaissances pour trouver les causes racines de nouvelles anomalies qui lui étaient jusqu'alors inconnues. Cette nouvelle approche basée sur le concept d'apprentissage artificiel est notamment utilisée par une autre génération de systèmes experts. La troisième génération de systèmes experts utilise le raisonnement à base de cas « case-based reasoning » et intègre la capacité d'apprendre naturellement avec l'expérience et à éviter une maintenance excessive [LEWIS \[1993\]](#).

2.4.2 Apprentissage artificiel par induction

L'induction se définit comme le processus par lequel on tire des lois de portée générale en partant de l'observation des cas particuliers [CORNÉJOLS et MICLET \[2013\]](#). Le système apprenant reçoit les données de l'univers dans lequel il est placé. Dans le cas de l'induction supervisée, chacune de ces données prend la forme d'un couple dans lequel on distingue d'une part la description d'une situation encore appelée observation (par exemple des alarmes observées sur un réseau ainsi que les paramètres scalaires mesurés sur ce réseau à un instant t quelconque) et d'autre part une réponse, que l'on appelle souvent étiquette ou sortie désirée (par exemple la panne ou cause racine des alarmes observées, survenue à un instant $t - \delta t$), qui est généralement fourni par un oracle. Une donnée d_i est donc un couple (observation, étiquette ou panne) que nous pouvons noter (o_i, p_i) .

L'induction peut être considérée comme la détermination ou l'approximation d'une fonction. Considérons par exemple que f est une fonction qui définit le comportement du phénomène de propagation de pannes et d'alarmes dans un réseau, c'est-à-dire que $\forall o_i, f(p_i) = o_i$. Le but d'un système de diagnostic apprenant serait dans ce cas de déterminer ou d'approximer autant que possible la fonction f^{-1} qui définit le comportement caché de propagation inverse de pannes et d'alarmes sur le réseau. L'induction supervisée consiste alors à déterminer à partir des données étiquetées, une fonction h telle que les hypothèses de pannes $h(o_i) \approx f^{-1}(o_i) = p_i$ (voir la Figure 2.5).

Dans le cas de l'induction non supervisée, les données ne sont pas étiquetées. Le système de diagnostic apprenant peut dans ce cas identifier des motifs ("patterns"), des régularités dans les observations qui lui permettent de corrélérer ces observations pour re-

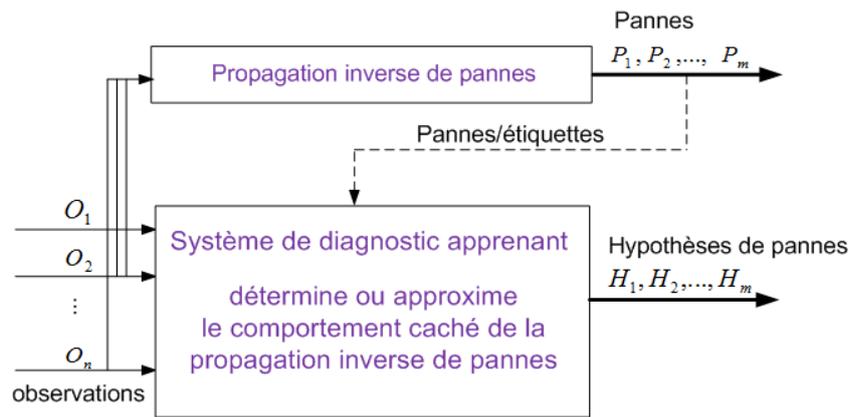


FIGURE 2.5 – Système de diagnostic apprenant.

trouver leurs causes racines. Un système apprenant sans supervision peut découvrir de nouvelles étiquettes inconnues par les experts du domaine étudié. Par exemple, un système de diagnostic apprenant sans supervision peut découvrir de nouvelles signatures de pannes, ce qui est très intéressant compte tenu de la grande diversité de signatures de pannes dans un réseau de télécommunications de grande dimension fournissant une panoplie de services différents. Le concept d'induction a été appliqué par les méthodes de diagnostic raisonnant à base de cas ("case based reasoning"), la troisième génération des systèmes experts.

2.4.3 Raisonnement à base de cas

L'idée principale du raisonnement à base de cas ou "case-based reasoning" (CBR), est de récupérer, adapter et exécuter les solutions des précédents problèmes pour évaluer les problèmes actuels [PENIDO et collab. \[1999\]](#). Les solutions de diagnostic passées sont stockées sous forme de cas dans une base de connaissances. Les cas contiennent les caractéristiques les plus pertinentes des solutions de diagnostics passés, ils sont adaptés et utilisés pour résoudre les nouveaux problèmes. L'expérience acquise en diagnostiquant ces nouveaux problèmes constitue de nouveaux cas stockés pour une prochaine utilisation. Un système de diagnostic raisonnant à base de cas est donc un système apprenant. Ce système intègre la capacité d'apprendre non seulement de ses précédentes solutions correctes de diagnostic, mais aussi de ses échecs. En effet, lorsqu'une tentative de diagnostic d'une situation échoue, le système identifie et journalise la raison de cet échec afin de s'en souvenir lors de futurs diagnostics de situations similaires [GARDNER et HARLE \[1996\]](#).

Un système CBR dispose d'une base de cas. Chaque cas possède une description détaillée du problème et une solution. Pour utiliser ces informations, un moteur est aussi nécessaire. Celui-ci va retrouver les cas similaires au nouveau problème posé. Après analyse, le moteur fournit une solution adaptée qui doit être validée. Enfin, le moteur ajoute le problème et sa solution dans la base de cas. La Figure 2.6 résume ces quatre étapes du cycle de vie d'un système CBR. De ces étapes se dégagent trois problèmes majeurs : (1) la représentation des cas, (2) la recherche des cas, (3) la création de la fonction d'adaptation. Pour développer un système de raisonnement fiable et efficace à base de cas, il est donc nécessaire de trouver une solution efficace à chacun de ces problèmes. La révision et l'apprentissage sont deux autres problèmes qui découlent des trois premiers.

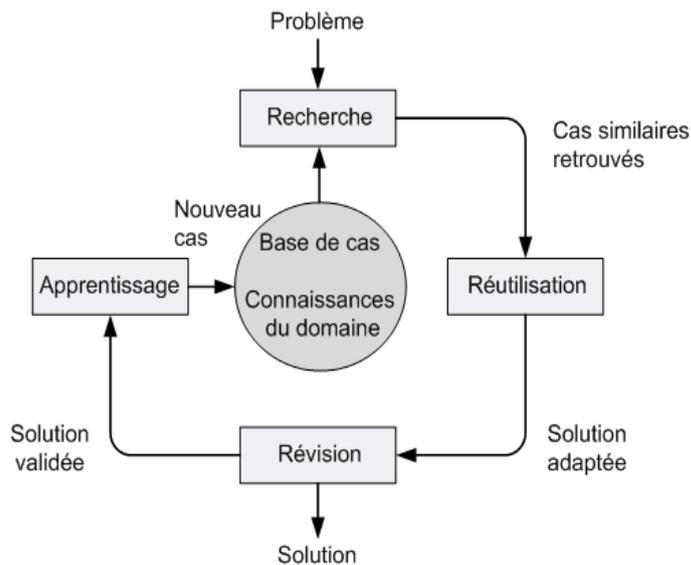


FIGURE 2.6 – Cycle de vie d'un système de raisonnement à base de cas.

La représentation des cas détermine l'efficacité et la rapidité de la recherche des cas dans la base. Il faut donc bien sélectionner les informations pertinentes du cas à représenter et le formalisme de cette représentation. De façon générale, les caractéristiques contiennent les informations relatives à la description du problème à résoudre, la solution du problème et les étapes de sa résolution, la fiabilité de cette solution et l'explication des échecs éventuels. Généralement, un cas est représenté comme un dictionnaire, c'est-à-dire un ensemble de couples attribut-valeurs. Les attributs des caractéristiques les plus discriminantes sont utilisés comme des index lors de la recherche et l'ajout de cas. Les index doivent être suffisamment concrets et abstraits à la fois pour qu'ils concernent un maximum de cas et qu'ils soient réutilisables dans les raisonnements futurs. Ils doivent aussi permettre de déduire rapidement les cas.

Avant la recherche des cas similaires, il est nécessaire d'étudier le problème posé. Il faut identifier ses caractéristiques, mais aussi son contexte si cela est possible. Si certaines informations sont manquantes, il est possible de négliger certaines caractéristiques ou d'interroger l'utilisateur. C'est au cours de cette étape, préambule à la recherche à proprement parler, que le système CBR doit essayer de déterminer et corriger les données bruitées ou incohérentes. Pour cela, on peut faire appel à des outils de "datacleaning". Il est aussi possible d'essayer de déduire des caractéristiques à partir d'un modèle d'apprentissage par induction. En ce qui concerne la réutilisation de cas, les situations les plus simples ou idéales sont celles pour lesquelles le cas similaire retrouvé dans la base est quasiment identique au problème posé, dans ce cas on peut réutiliser directement la solution existante sans adaptation. Le plus souvent, et notamment dans un système CBR complexe, tel que celui du diagnostic d'un réseau de télécommunication avec une très grande diversité de signatures d'anomalies, une adaptation des solutions préexistantes est presque toujours requise. L'adaptation consiste à construire une nouvelle solution à partir des caractéristiques du problème courant et des cas similaires trouvés. Cette phase met l'accent sur les différences entre les cas trouvés et le problème, et sur l'information utile à transférer à la nouvelle solution. Cette phase nécessite des capacités d'apprentissage par induction du système CBR.

La solution au problème posé est ensuite évaluée pour déterminer sa fiabilité. Si la solution n'est pas fiable, c'est-à-dire que son évaluation n'est pas concluante, le système CBR doit réviser si possible son analyse du problème posé ou tout au moins identifier et journaliser les raisons de l'échec. La révision ou réparation peut être vue comme une autre fonction d'adaptation. La seule différence est que dans la réparation, on travaille à partir d'une solution incorrecte, mais adaptée au problème au lieu de solutions correctes inadaptées. Le système CBR s'appuie sur les explications de l'échec pour réaliser les modifications. Lorsque l'évaluation de la solution est concluante, c'est-à-dire que la solution est fiable, le système CBR doit retenir cette nouvelle expérience. Il faut donc déterminer quelles informations doivent être sauvegardées et sous quelle forme, et comment indexer ce nouveau cas dans la base.

Plusieurs projets et logiciels utilisant le raisonnement à base de cas ont été développés dans le passé. CHEF [HAMMOND \[1989\]](#) est un système CBR qui réalise des recettes de cuisine. L'utilisateur indique au programme les aliments dont il dispose et CHEF cherche à élaborer une recette à partir de ces ingrédients. CHEF possède une grande base de données de recettes de cuisines valides et cherche à créer par adaptation à ces recettes pré-existantes, une nouvelle recette contenant les ingrédients choisis. Lorsque les contraintes imposées par l'utilisateur sont trop fortes et qu'aucune solution n'a été trouvée, CHEF donne une explication indiquant les raisons de l'échec.

Les projets européens INRECA (INduction and REasoning from CAses) [RALPH \[2001\]](#) de 1992 à 1995 et INRECA II de 1996 à 1999 ont apporté plusieurs contributions dans le domaine de la représentation des connaissances des systèmes CBR. Le concept de "Knowledge Container" a été développé dans [RICHTER \[1995\]](#). Quatre conteneurs sont utilisés pour stocker les connaissances (voir la Figure 2.7). En principe, chaque conteneur est capable de stocker toutes les connaissances disponibles, mais cela ne signifie pas que cette approche soit souhaitable. Les trois premiers conteneurs contiennent des connaissances compilées (par "compilation", on entend le temps de développement avant la résolution du problème courant, et "compilation" est pris dans un sens très général, y compris les activités de codage), tandis que la base de cas contient les connaissances spécifiques aux cas qui sont interprétées au moment de l'exécution, c'est à dire lors de la résolution du problème courant. L'atout principal de ce schéma de représentation est la souplesse nécessaire pour décider pragmatiquement quel récipient contient quelles connaissances et donc de choisir le degré approprié de compilation. Une stratégie générale pour le développement de systèmes CBR est de compiler le moins de connaissances possible.

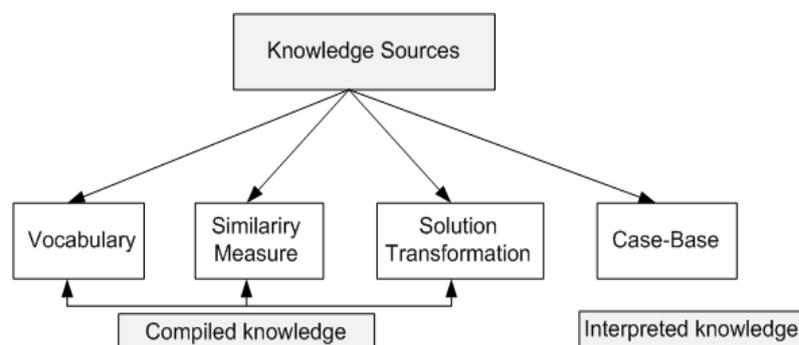


FIGURE 2.7 – Distribution des connaissances dans un système CBR [RICHTER \[1995\]](#) .

INRECA était l'un des premiers projets de systèmes CBR ayant développé une repré-

sentation strictement orientée objet des cas [BERGMANN et STAHL \[1998\]](#). Ces représentations sont particulièrement appropriées pour les domaines complexes dans lesquels des cas avec des structures différentes se produisent. Les cas sont représentés sous forme de collections d'objets, dont chacun est décrit par un ensemble d'attributs-valeurs. La structure d'un objet est décrite par une classe d'objets qui définit l'ensemble d'attributs conjointement avec un type (ensemble de valeurs possibles) pour chaque attribut. Les classes d'objets sont disposées dans une hiérarchie de classes, c'est habituellement un arbre n-aires dans lequel les sous-classes héritent des attributs ainsi que de la définition de la classe parente.

Dans [LEWIS et DREA \[1993\]](#), le problème de l'adaptation des cas est étudié. Une technique appelée adaptation paramétrique est proposée. Elle se base sur l'existence dans un ticket d'incident ("trouble ticket"), d'une relation entre les variables qui décrivent un problème et les variables qui spécifient la solution correspondante. Le système CBR développé prend en considération les paramètres de ladite relation lors de la proposition d'une solution à un nouveau problème posé. Ces paramètres sont représentés avec des variables linguistiques (c'est-à-dire des variables qui prennent des valeurs linguistiques et non numériques), et des fonctions d'appartenance des variables à des ensembles flous sont proposées telles que la valeur numérique des variables correspond à leurs degrés d'appartenance aux ensembles flous. Lewis a révisé dans [LEWIS \[1993\]](#), les techniques de représentation, de recherche, d'adaptation et d'intégration des connaissances dans une librairie de cas appelée CRITER, un système CBR basé sur des tickets d'incidences pour la gestion et le diagnostic de pannes d'un réseau. Nous rappelons qu'un système à tickets d'incidences ("trouble ticket system") est utilisé pour stocker les problèmes survenus sur le réseau, agissant comme une mémoire historique du réseau. Melchior et collab. ont aussi proposé dans [MELCHIOR et ROCKENBACH TAROUCO \[1999\]](#), un système CBR de diagnostic des pannes d'un réseau appelé DUMBO qui utilise les connaissances des cas de diagnostic stockés dans un système à tickets d'incidences pour proposer des solutions de diagnostic aux nouvelles anomalies survenues.

Une approche avec une architecture hybride entre un système expert à base de règles et un système CBR est proposée dans [ANDREAS \[2006\]](#) pour le diagnostic des dysfonctionnements des services fournis par un réseau. Les règles du système expert sont dérivées d'un modèle des services et de l'infrastructure sous-jacente. Le module CBR avec ses capacités d'apprentissage par induction utilise les cas de diagnostic résolus par le système expert à base de règles, pour proposer des solutions aux problèmes non résolus par les règles expertes. L'expérience ainsi acquise est utilisée pour améliorer le modèle de services et donc d'améliorer également les règles expertes.

Comme l'approche à base de modèle, le raisonnement à base de cas (CBR, case based reasoning) pallie également les deux inconvénients majeurs des systèmes experts à base de règles : la fragilité et la difficulté relative à l'acquisition des connaissances qui conduit à une maintenance difficile et fastidieuse. Les capacités d'un système CBR d'apprendre de son expérience acquise lui permettent de proposer des solutions à des problèmes nouveaux qui sortent de l'expertise contenue dans sa base des connaissances, facilitant ainsi la maintenance du système. D'autre part, l'unité de connaissance d'un système CBR est le cas et non pas la règle. Il est plus facile d'articuler, d'examiner et d'évaluer un cas qu'une règle [HOUNKONNOU \[2013\]](#). Un système CBR est également capable d'apprendre de ses propres erreurs/échecs et d'auto améliorer ses performances. Cependant, comme pour

les deux premières générations de systèmes experts, les systèmes CBR qui sont perçus comme la troisième génération de systèmes experts sont également trop façonnés pour le domaine applicatif investigué. L'efficacité temporelle d'un système CBR peut être problématique pour le diagnostic en temps réel qui nécessite de trouver à la volée des solutions aux problèmes qui surviennent. En effet, la phase de recherche et d'adaptation des cas peut être lourde surtout si le format de représentation des cas ne s'y prête pas. Il faut également faire attention à la phase d'évaluation des solutions de nouveaux problèmes. En effet, une mauvaise évaluation pourrait entraîner l'intégration de cas erronés dans la base des connaissances et faire ainsi dériver le système dans son ensemble. C'est d'ailleurs la raison pour laquelle cette phase est souvent externe au système CBR. La présence d'experts peut être nécessaire pour évaluer la fiabilité des nouveaux cas avant leur intégration.

2.4.4 Réseaux de neurones

Le cerveau humain peut être décrit comme un réseau de neurones biologiques, c'est-à-dire un ensemble de neurones interconnectés transmettant des modèles élaborés de signaux électriques. Les dendrites reçoivent les signaux d'entrée et, sur la base de ces entrées, un neurone produit un signal de sortie via un axone [SHIFFMAN \[2012\]](#). La Figure 2.8 présente un schéma synoptique de deux neurones biologiques connectés.

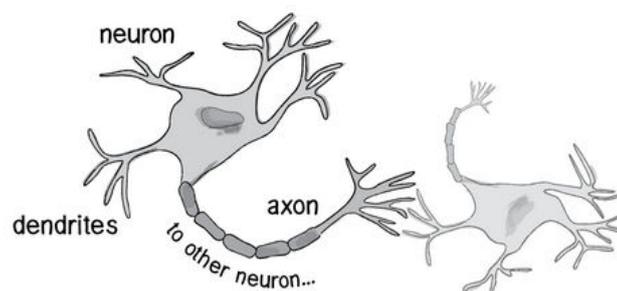


FIGURE 2.8 – Schéma synoptique de deux neurones connectés [SHIFFMAN \[2012\]](#).

Les informaticiens ont longtemps été inspirés par le cerveau humain. En 1943, Warren S. McCulloch, un neuro-scientifique, et Walter Pitts, un logicien, ont développé le premier modèle conceptuel d'un réseau de neurones artificiels (ANN). Dans leur article, «A logical calculus of the ideas imminent in nervous activity», ils décrivent le concept d'un neurone, une seule cellule vivante dans un réseau de cellules qui reçoit des entrées, traite ces entrées, et génère une sortie. Par analogie à un neurone biologique, un neurone artificiel est perçu comme un processeur autonome avec des canaux unidirectionnels pour la communication avec les autres neurones qui lui sont connectés. Un neurone artificiel à plusieurs canaux d'entrées fonctionnant comme des dendrites, et un seul canal de sortie fonctionnant comme un axone. Les points de connexions entre les neurones sont appelés synapses. L'opération typique d'un neurone artificiel est de calculer une somme pondérée des signaux d'entrée et de générer un signal de sortie si cette somme dépasse un certain seuil. La somme pondérée des signaux d'entrée est effectuée par la fonction de combinaison, qui multiplie le vecteur des entrées par une matrice de transformation. Le signal de sortie est généré par la fonction d'activation (ou fonction de seuillage, ou encore fonction de transfert) qui sert à introduire une non-linéarité dans le fonctionnement du neurone. La Figure 2.9 présente la structure fonctionnelle d'un neurone artificiel.

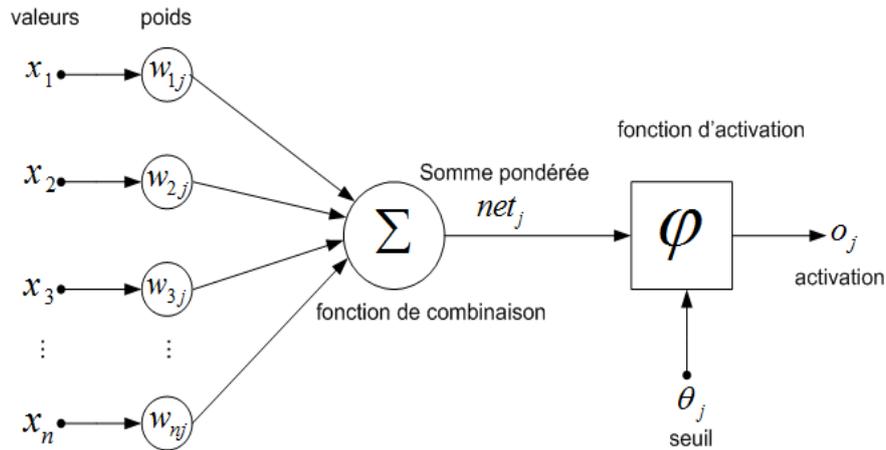


FIGURE 2.9 – Structure fonctionnelle d'un neurone artificiel.

Un ANN est en général composé d'une succession de couches dont chacune prend ses entrées sur les sorties de la précédente. Chaque couche i est composée de N_i neurones, prenant leurs entrées sur les N_{i-1} neurones de la couche précédente. A chaque synapse est associé un poids synaptique, de sorte que les sorties des neurones de la couche $i - 1$ sont multipliées par ces poids, puis additionnées par les neurones de niveau i , ce qui est équivalent à multiplier le vecteur d'entrée par une matrice de transformation. Mettre l'une derrière l'autre les différentes couches d'un réseau de neurones reviendrait à mettre en cascade plusieurs matrices de transformation et pourrait se ramener à une seule matrice, produit des autres, s'il n'y avait pas à chaque couche, la fonction de sortie qui introduit une non-linéarité à chaque étape. Ceci montre l'importance du choix judicieux d'une bonne fonction de sortie : un réseau de neurones dont les sorties seraient linéaires n'aurait aucun intérêt.

La structure simple d'un ANN, celle dite statique, ne présente pas de boucles de rétroaction et est le plus souvent utilisée dans un but d'apprentissage supervisé. Les structures plus complexes avec des boucles de rétroactions entre les neurones, sont celles des ANNs dites "cartes de Kohonen" ou cartes auto-organisatrices. Ces derniers sont en fait utilisés pour des algorithmes d'apprentissage non supervisé. Dans le contexte du diagnostic des réseaux, les ANNs sont très souvent utilisés pour faire de l'apprentissage supervisé puisqu'il est assez facile d'obtenir des données étiquetées, c'est à dire des correspondances entre des observations ou des mesures sur le réseau et les pannes associées. L'apprentissage dans ce cas consiste à utiliser les données étiquetées pour ajuster les poids des connexions entre les neurones. Des symptômes observés sur le réseau sont placés en entrée des neurones de la première couche du réseau de neurones et la sortie (pannes) produite est comparée avec l'étiquette des symptômes observés. Si le résultat de la comparaison n'est pas concluant, les poids des connexions entre les neurones sont ajustés pour minimiser la différence.

Reategui et collab. ont proposé dans [REATEGUI et collab. \[1997\]](#), une approche hybride qui intègre le raisonnement à base de cas (CBR) avec un réseau neuronal artificiel (ANN) pour le diagnostic médical. Lors de la résolution d'un nouveau problème posé, le réseau neuronal est utilisé pour faire des hypothèses et guider ainsi le module CBR dans la recherche de précédents cas similaires qui vérifient ces hypothèses. La connaissance acquise par le réseau de neurones est interprétée et associée à des descripteurs symbo-

liques de diagnostic, qui sont conservés et utilisés par le système pour déterminer si une réponse définitive est crédible, et de construire des explications pour le raisonnement effectué. Le modèle hybride ANN-CBR a été utilisé pour le développement d'un système de diagnostic des maladies cardiaques congénitales. Le système a été évalué en utilisant deux bases de données cardiologiques avec un total de 214 cas de coronaropathie. Le système hybride parvient à résoudre des problèmes qui ne peuvent pas être résolus par le réseau neuronal avec un bon niveau de précision. En outre, le système hybride propose des solutions aux problèmes communs des modèles CBR, tels que l'indexation et l'adaptation de cas, ainsi que pour des problèmes des réseaux de neurones, tels que l'interprétation des connaissances stockées dans un réseau de neurones et l'explication du raisonnement. Cette approche hybride a aussi été utilisée dans [YANG et collab. \[2003\]](#) pour le diagnostic de pannes dans les moteurs électriques.

Les réseaux de neurones et plus précisément les réseaux de neurones multicouches intègrent d'importantes capacités d'induction, c'est-à-dire qu'ils sont capables de tirer des généralisations plausibles sur la base d'observations limitées. En effet, un réseau de neurones assimile aisément sous une forme dense, des exemples éventuellement nombreux. Ensuite, grâce aux exemples appris, le réseau traite des exemples distincts encore non rencontrés, mais similaires. Ces capacités d'induction font des réseaux de neurones un outil très efficace pour la corrélation d'alarmes ainsi que d'autres applications telles que la reconnaissance de formes ou de motifs ("pattern matching"), la régression ou l'approximation de fonctions complexes, etc. Ils ont été utilisés dans [GARDNER et HARLE \[1997\]](#) [HERMANN \[2002\]](#) pour la localisation de pannes dans les réseaux. Dans leur version probabiliste, les réseaux de neurones font davantage face aux données manquantes, ambiguës ou même erronées. Cependant les réseaux de neurones demandent beaucoup d'entraînement ("training") pour être capables de généraliser en plus de n'être pas intelligibles pour leurs utilisateurs.

2.5 Réseau de Petri

Les réseaux de Petri sont couramment utilisés pour modéliser le comportement dynamique de systèmes discrets. Ils sont composés de deux types d'objets : les places et les transitions, l'ensemble des places permet de représenter les états du système, l'ensemble des transitions représente alors l'ensemble des événements dont l'occurrence provoque la modification de l'état du système [BRAMS \[1983\]](#). Les places sont des variables d'états du système à valeurs entières. De façon imagée, ces valeurs entières sont représentées par autant de marques ou jetons affectés à une place. L'état du système est alors associé à un marquage définissant pour chaque place le nombre de marques ou de jetons qui lui est affecté (ou bien qu'elle contient). À l'occurrence d'un événement correspond le franchissement d'une transition dépendant de la satisfaction de préconditions. Ces préconditions portent sur le nombre de marques contenues dans les places attachées à la transition associée à l'évènement. Ces places seront appelées places d'entrée de la transition. Le franchissement d'une transition a pour effet de modifier le marquage et donc l'état du système : on retire toutes les marques qui ont satisfait la précondition associée à la transition, on ajoute des marques dans des places appelées places de sortie de la transition. Dans [BRAMS \[1983\]](#), un réseau de Petri est défini par un quadruplet $R = (P, T, Pré, Post)$ où : P et T désignent respectivement l'ensemble fini des places et l'ensemble fini des transitions tel que $P \cap T = \emptyset$, $Pré$ et $Post$ sont des applications définies de $P \times T$ vers \mathbb{N} . Un réseau de Petri marqué est un couple $N = (R, M)$ formé du réseau de Petri R et d'une application

$M : P \rightarrow \mathbb{N}$ appelée marquage, tel que $\forall p \in P, M(p)$ est le nombre de marques ou d'états possibles de la place p . Une transition $t \in T$ est franchissable pour un marquage M si et seulement si $\forall p \in P, M(p) \geq \text{Pré}(p, t)$, c'est la précondition de franchissement de t . Si t est franchissable pour M (un état du système), le franchissement de t permet d'obtenir un nouveau marquage M' tel que $\forall p \in P, M'(p) = M(p) - \text{Pré}(p, t) + \text{Post}(p, t)$. Puisque $M(p) \geq \text{Pré}(p, t)$ alors $M'(p) \leq \text{Post}(p, t)$, c'est la postcondition pour qu'un marquage M' puisse résulter du franchissement de la transition t . L'utilisation d'un réseau de Petri est très souvent associée à son graphe des marquages. Un marquage désigne la distribution des jetons dans les places. Le graphe des marquages d'un réseau de Petri (R, M_0) de marquage initial M_0 est un graphe orienté dont les nœuds sont les marquages, et chaque arc relie un marquage à un autre qui est immédiatement accessible par une transition : si une transition t provoque un passage du système de l'état M_0 à l'état M_1 , un arc est tracé entre les marquages M_0 et M_1 et il est marqué avec t . Le graphe des marquages donne une vue simple de l'évolution d'un réseau de Petri, mais sa construction nécessite l'énumération exhaustive de tous les états du système modélisé.

Une utilisation des réseaux de Petri pour le diagnostic dans les systèmes distribués a été proposée dans AGHASARYAN et collab. [1998] avec une application concernant la gestion de pannes dans les réseaux de télécommunications. Cette approche de diagnostic AGHASARYAN et collab. [1998] : (1) utilise la capacité naturelle des réseaux de Petri à modéliser la concurrence dans les systèmes distribués, (2) n'utilise pas le graphe des marquages du réseau de Petri dans les algorithmes de diagnostic, ceci pour éviter l'explosion du nombre d'états due à l'énumération complète de tous les événements possibles tels que les alarmes par exemple, pouvant survenir dans le réseau de Petri, (3) mais s'appuie plutôt sur la sémantique d'ordre partiel, notamment à travers les techniques de dépliage ou de déploiement du réseau de Petri. En effet, le dépliage est un algorithme utilisant un ordre partiel entre les événements qui évite de tous les énumérer contrairement au graphe des marquages dont l'utilisation limiterait cette approche de diagnostic aux petits réseaux de télécommunication modélisés avec de petits réseaux Petri ayant un petit nombre d'évènements ou de transitions.

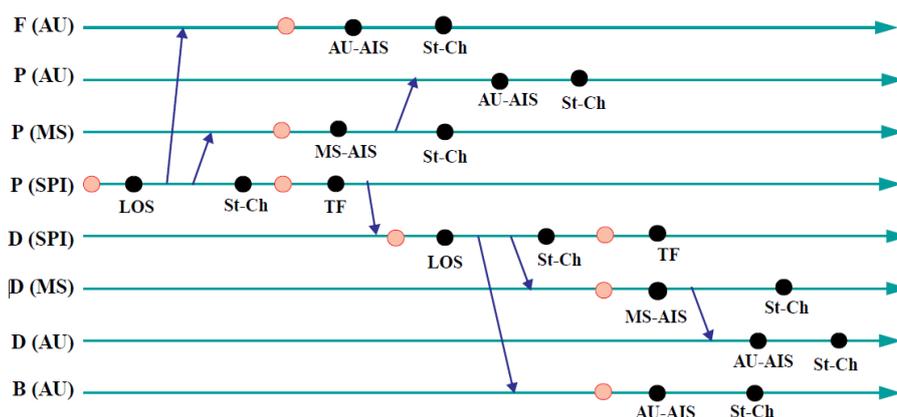


FIGURE 2.10 – Propagation d'une perte de signal sur le port P du module STM-1 du réseau SDH AGHASARYAN et collab. [1998].

Cette approche de diagnostic introduit la notion de motif d'alarmes ("alarm pattern") désignant un ensemble d'alarmes avec des dépendances ou concurrences entre elles, traduisant l'occurrence d'une panne. Un motif d'alarmes peut être vu comme un état transitoire entre deux pannes, c'est-à-dire que les pannes sont vues comme des états locaux

entre motifs d'alarmes. De ce fait, la propagation de pannes et d'alarmes dans un réseau de télécommunications est modélisée comme suit par un réseau de Petri [AGHASARYAN et collab. \[1998\]](#) : (1) les pannes sont représentées par les places, (2) le passage d'une panne à une panne secondaire est représenté par une transition à laquelle est associée le motif d'alarmes correspondant. Ce travail effectué en collaboration entre l'IRISA et le CNET (le centre de recherche de France-Telecom actuellement appelé Orange Labs) a été appliqué au réseau SDH (Synchronous Digital Hierarchy). La Figure 2.10 présente la propagation de pannes et d'alarmes dues à la survenance d'une perte de signal (LOS) sur un port P du module (Synchronous Transport Module) STM-1 d'un réseau SDH, on y observe notamment un ordre partiel entre ces alarmes. La Figure 2.11 présente le modèle de Petri associé. Les places sont représentées par des petits cercles et les transitions par de petits rectangles. Il faut remarquer que le réseau de Petri 2.11 respecte bien l'ordre partiel entre les alarmes présenté à la Figure 2.10.

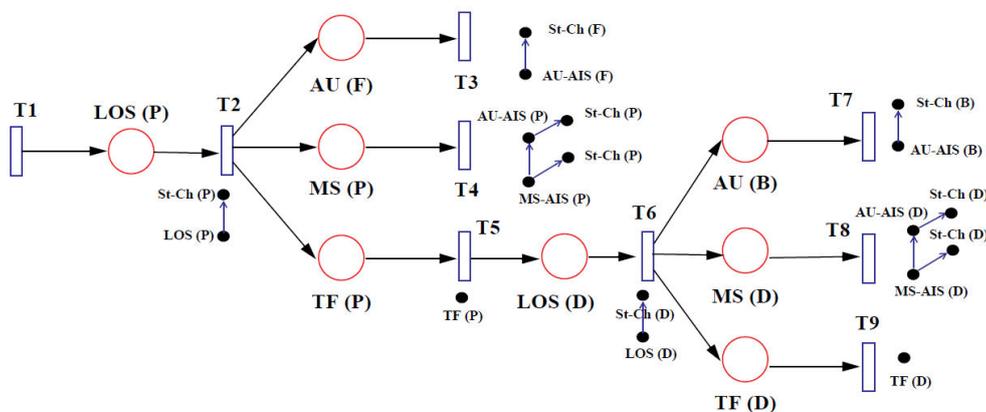


FIGURE 2.11 – Réseau de Petri modélisant la propagation d'une rupture du signal sur un port STM-1 dans le réseau SDH [AGHASARYAN et collab. \[1998\]](#).

La détection de pannes consiste à reconnaître un motif d'alarmes et le diagnostic est calculé en appliquant les techniques de dépliage des réseaux de Petri [BENVENISTE et collab. \[2003\]](#) [ESPARZA et collab. \[2002\]](#) [CABASINO et collab. \[2012\]](#), afin de tirer parti de la représentation par ordre partiel qui évite l'énumération des séquences d'évènements qui se produisent, notamment si l'analyse se fait par la construction d'un graphe de marquage. Notons qu'un réseau de Petri déplié est un autre réseau de Petri plus dense, mais dont la structure est plus simple, car acyclique, ce qui permet de partir d'un motif d'alarmes reconnu et remonter le réseau de Petri déplié pour retrouver la cause racine. Une version probabiliste de cette approche a aussi été développée dans [AGHASARYAN et collab. \[1998\]](#). L'intérêt de probabiliser le réseau de Petri est (1) de prendre en compte l'incertitude permettant de traiter les mesures aberrantes du réseau de télécommunication modélisé, les pertes d'alarmes et les alarmes ambiguës, (2) de discriminer entre plusieurs pannes possibles en estimant leurs vraisemblances par rapport aux alarmes observées, (3) d'apporter de la robustesse dans le processus de diagnostic qui se base sur une extension de l'algorithme de Viterbi pour calculer le chemin le plus vraisemblable dans le réseau de Petri probabilisé.

2.6 Graphe de dépendances

Dans la Section 2.3, nous avons présenté l'approche de diagnostic d'un réseau à base d'un modèle de ce réseau, le modèle étant une représentation structurelle et fonction-

nelle du réseau, construit à partir de l'expertise métier du réseau. Un graphe de dépendances est un formalisme graphique de représentation des connaissances expertes encodées dans le modèle du réseau. Un graphe de dépendances modélise plus précisément la propagation des pannes et d'alarmes sur le réseau. Les causes racines et les symptômes qu'elles génèrent sont représentés, mais aussi les chaînes complexes de dépendances entre ces causes racines, les causes intermédiaires et les symptômes [KÄTKER et PATEROK \[1997\]](#). Les connaissances représentées graphiquement proviennent des standards ou des normes du réseau modélisé, mais aussi de l'analyse par des experts, des résultats d'éventuels tests expérimentaux, des "logs" des équipements et surtout des "logs" du système de gestion du réseau qui contient des informations très pertinentes sur les états de fonctionnement et de dysfonctionnement du réseau. Puisque le graphe est construit par des experts humains, les dépendances entre les noeuds du graphe indiquent des relations de causes à effets entre eux. En effet, le phénomène de propagation de pannes et d'alarmes sur un réseau est un ensemble de relations de causes à effets, et le mode de la pensée privilégié et naturel des humains pour représenter de telles relations est un graphe de dépendances causales. Le raisonnement ou l'inférence sur un graphe de dépendances consiste à déterminer l'état de tous les noeuds pertinents, étant donnée l'observation de certains noeuds (généralement des alarmes ou des paramètres scalaires mesurés sur le réseau).

Un graphe de dépendances causales est un graphe orienté acyclique $G = (\mathbb{N}, \mathbb{S})$, où \mathbb{N} est l'ensemble des noeuds et $\mathbb{S} \subseteq \mathbb{N} \times \mathbb{N}$ est l'ensemble des arcs représentant les dépendances causales. Le terme acyclique signifie qu'il n'existe pas de boucles orientées entre les noeuds, c'est-à-dire que l'ensemble des descendants et l'ensemble des ascendants de chaque noeud sont disjoints. Plusieurs algorithmes de diagnostic des réseaux utilisant ce formalisme ont été proposés dans la littérature. Nous nous focalisons principalement sur les travaux exemplaires de [JINGXIAN et collab. \[2010\]](#) [JINGXIAN et collab. \[2011\]](#). Ces travaux décrivent un algorithme innovant pour le diagnostic des réseaux de télécommunications avec un modèle basé sur un graphe de dépendances causales. Dans [JINGXIAN et collab. \[2010\]](#) [JINGXIAN et collab. \[2011\]](#), la structure du graphe est basée sur cinq types de noeuds et un type d'arc (voir la Figure 2.12).

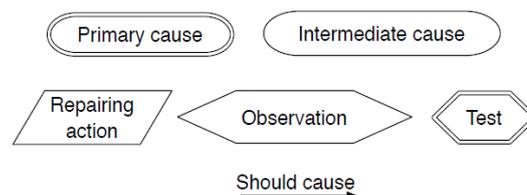


FIGURE 2.12 – Les composants du graphe de dépendances causales [JINGXIAN et collab. \[2010\]](#) [JINGXIAN et collab. \[2011\]](#).

Les causes racines ("primary causes") qui n'ont aucun prédécesseur, les causes intermédiaires ("intermediate causes") qui représentent des états partiels ou temporaires de la propagation de pannes, les observations qui sont les signatures visibles des pannes, les tests utilisés pour vérifier une hypothèse ou lever une incertitude, l'action de réparation ("repairing action") est un noeud spécial puisqu'il ne fait pas partie des relations de causes à effets, et l'arc labellisé "should cause" dénote une dépendance déterministe. Notons que les noeuds observables peuvent tout aussi bien être des noeuds intermédiaires succédés par d'autres noeuds observables. Le fait qu'ils soient considérés comme les "feuilles" du graphe et n'ont de ce fait aucun successeur signifie tout simplement que les conséquences

d'un noeud observable ne peuvent être que des noeuds observables. Les noeuds de type "repairing action" sont factices, ce sont juste des labels qui indiquent l'action de réparation à exécuter une fois que le diagnostic est déterminé.

Le processus de diagnostic consiste à collecter toutes les observations courantes, et déterminer sur la base de ces observations, les causes racines. Certains tests peuvent être déclenchés durant le processus de diagnostic pour discriminer entre certaines ambiguïtés. Les noeuds du graphe de dépendances sont des variables binaires. Un noeud peut être incriminé ("guilty") ou non ("innocent") JINGXIAN et collab. [2010] JINGXIAN et collab. [2011]. Pour une image instantanée du réseau, on positionne les noeuds correspondants aux symptômes observés à leur état "guilty", si ces symptômes sont négatifs. Les noeuds correspondants aux symptômes positifs observés sont positionnés à "innocent", et les états des autres noeuds du graphe sont inconnus. L'objectif de l'inférence sur le graphe de dépendances causales est de calculer la combinaison des états des noeuds non observés y compris les noeuds à la racine du graphe. Cinq règles consistantes sont définies pour guider le processus de diagnostic : (\mathbf{R}_1) si un noeud est incriminé, alors tous ses successeurs doivent être également incriminés, (\mathbf{R}_1') si un noeud différent d'une cause racine est incriminé, alors au moins un des ses prédécesseurs doit être incriminé, (\mathbf{R}_2) si un noeud est innocent, alors tous ses prédécesseurs doivent être innocents, (\mathbf{R}_2') si tous les prédécesseurs d'un noeud sont innocents, alors le noeud doit être innocent, (\mathbf{R}_3) l'état d'un noeud de test est incriminé ou innocent selon le résultat de l'exécution du test en question, c'est-à-dire qu'un test permet d'ajouter de nouvelles observations utiles au processus de diagnostic.

Plusieurs définitions introduites dans JINGXIAN et collab. [2010] JINGXIAN et collab. [2011] sont nécessaires pour formaliser le modèle et le raisonnement exploitant les règles consistantes pour effectuer un diagnostic. Pour chaque noeud $n \in \mathbb{N}$, on note $S(n)$ l'ensemble des successeurs du noeud n , $\hat{S}(n)$ l'ensemble de tous les descendants de n , $S^{-1}(n)$ l'ensemble des parents de n et $\hat{S}^{-1}(n)$ l'ensemble de tous les ascendants de n . De façon plus générale, si $N \subset \mathbb{N}$ est un sous-ensemble de noeuds du graphe, l'ensemble des descendants du sous-ensemble N est $\hat{S}(N) = \cup_{n \in N} \hat{S}(n)$ et l'ensemble des ascendants est $\hat{S}^{-1}(N) = \cup_{n \in N} \hat{S}^{-1}(n)$. L'hypothèse de graphe causal acyclique est formalisée par la propriété $\forall n \in \mathbb{N}, n \notin \hat{S}(n)$. L'ensemble \mathbb{P} des pannes considérées dans le graphe et l'ensemble \mathbb{O} des observations se définissent par $\mathbb{P} = \{n \in \mathbb{N} | S^{-1}(n) = \emptyset\}$ et $\mathbb{O} = \{n \in \mathbb{N} | S(n) = \emptyset\}$. Une observation $a \in \mathbb{O}$ est un symptôme (une alarme par exemple) d'une panne $p \in \mathbb{P}$, si et seulement si il existe un chemin causal de p à a , c'est-à-dire que $a \in \hat{S}(p)$. Par conséquent, l'ensemble des symptômes d'une panne p est défini par $\hat{S}(p) \cap \mathbb{O}$. Si deux causes racines p_1 et p_2 sont indépendantes, alors les effets (symptômes observés et causes intermédiaires) de ces deux pannes sont l'union des effets de chaque panne, c'est-à-dire que $\hat{S}(\{p_1, p_2\}) = \hat{S}(p_1) \cup \hat{S}(p_2)$. Notons que cette assertion n'est plus absolument vérifiée si les deux pannes sont dépendantes, parce que certains effets de l'une des pannes peuvent neutraliser des effets de l'autre, et certains symptômes peuvent donc ne pas être observés.

Le diagnostic D d'un ensemble d'observations $A \in \mathbb{O}$ est défini dans JINGXIAN et collab. [2010] JINGXIAN et collab. [2011] comme un sous-ensemble de \mathbb{P} qui explique les observations. En d'autres mots D est un diagnostic de A si et seulement si $A \subseteq \hat{S}(D)$ et $\hat{S}(D) = \cup_{p \in D} \hat{S}(p)$ si on fait l'hypothèse d'indépendance entre les pannes. Avec cette hypothèse, si D est un diagnostic de A , alors $\forall p \in \mathbb{P}, D \cup \{p\}$ est aussi un diagnostic de A . Ainsi, pour un ensemble de symptômes A on peut ordonner les diagnostics correspondants en

utilisant la relation \subseteq . L'objectif du diagnostic est donc de chercher tous les diagnostics minimaux. Un diagnostic D de A est minimal si et seulement s'il n'existe pas un autre diagnostic D' de A tel que $D' \subsetneq D$, c'est-à-dire que si D et D' sont deux diagnostics minimaux de A tel que $D' \subset D$, alors $D = D'$. La recherche de l'ensemble des diagnostics de A , est basée sur la formulation et l'explication d'hypothèses de diagnostic. Une hypothèse H est définie par : (1) un ensemble $G(H) \subseteq \mathbb{N}$ de noeuds incriminés, avec $A \subset G(H)$, (2) un ensemble $I(H) \subseteq \mathbb{N}$ de noeuds innocents, (3) un ensemble $X(H) \subseteq G(H)$ contenant des noeuds inexpliqués, c'est-à-dire des noeuds qui n'ont pas de prédécesseurs incriminés. Une hypothèse H est close lorsqu'elle explique tous les symptômes observés c'est-à-dire que $X(H) = \emptyset$.

La procédure d'inférence ou de recherche de l'ensemble (initialisé à \emptyset) des diagnostics des observations A commence avec un ensemble $\mathbb{H} = \{H_0\}$ d'hypothèses contenant la seule hypothèse $\{H_0\}$ défini par $G(H_0) = X(H_0) = A$ et $I(H_0) = \mathbb{O} \setminus A$. Cette procédure d'inférence est un algorithme itératif qui se termine lorsque chacune des hypothèses de l'ensemble \mathbb{H} est close et supprimée de l'ensemble \mathbb{H} , c'est-à-dire que la condition de terminaison est $\mathbb{H} = \emptyset$. Chaque itération traite (essaie d'expliquer) une hypothèse $H \in \mathbb{H}$, et le traitement d'une hypothèse peut conduire à la formulation d'autres hypothèses ajoutées dans l'ensemble \mathbb{H} . Si $X(H) = \emptyset$, l'hypothèse H est close et supprimée de \mathbb{H} , le diagnostic associé qui est ajouté à l'ensemble des diagnostics, est défini par $G(H) \cap \mathbb{P}$. Si $X(H) \neq \emptyset$, c'est-à-dire que l'hypothèse de diagnostic H n'explique pas tous les noeuds du graphe de dépendances, alors l'hypothèse H doit être développée. Le développement d'une hypothèse H est une sous-procédure qui tente d'expliquer les noeuds de l'ensemble $X(H)$. La sous-procédure consiste à choisir aléatoirement un noeud $n \in X(H)$, le supprimer de $X(H)$ et vérifier s'il est expliqué ou pas. Le noeud n choisit est expliqué si la condition $S^{-1}(n) = \emptyset$ ou $S^{-1}(n) \cap G(H) \neq \emptyset$ est vérifiée, c'est-à-dire que n est une cause racine ou au moins un prédécesseur de n est incriminé. Dans le cas contraire, pour chaque prédécesseur $m \in S^{-1}(n)$ de n , une nouvelle hypothèse H' est formulée telle que $G(H') = G(H) \cup \{m\}$, $X(H') = X(H) \cup \{m\}$ et $I(H') = I(H)$. La cohérence de l'hypothèse H' avec les règles de consistance causale est ensuite vérifiée. L'hypothèse H' est inconsistante si au moins un descendant de m est innocent (alors que m est supposé incriminé, ce qui est inconsistent avec la règle \mathbf{R}_1), c'est-à-dire que $\hat{S}(m) \cap I(H') \neq \emptyset$, sinon H' est consistante, et dans ce dernier cas, l'ensemble $G(H')$ est actualisé en ajoutant les noeuds de l'ensemble $\hat{S}(m)$. H' est ensuite ajoutée à l'ensemble des hypothèses \mathbb{H} . La procédure d'inférence se termine avec un ensemble de diagnostics expliquant toutes les observations A . Le diagnostic minimal est obtenu en ordonnant les diagnostics selon la relation \subseteq .

Le formalisme proposé dans [JINGXIAN et collab. \[2010\]](#) [JINGXIAN et collab. \[2011\]](#) présente deux autres caractéristiques intéressantes. La première c'est la possibilité d'effectuer des tests et collecter des observations additionnelles qui permettent de confirmer ou d'infirmer certaines hypothèses de diagnostic. La sélection des tests les plus appropriés à effectuer devient donc importante. En effet, il faut bien sélectionner les tests qui lèvent le maximum d'incertitudes dans le processus du diagnostic. On parle dans ce cas de diagnostic actif. Le nombre de tests doit être aussi réduit au maximum pour éviter de surcharger le réseau avec un trafic important. La seconde caractéristique c'est la possibilité de faire du diagnostic distribué. L'idée consiste dans ce cas à séparer le graphe de dépendances causales en plusieurs sous-graphes, partageant certains noeuds. Chaque sous-graphe modélise un segment du réseau et le diagnostic de ce segment est effectué localement dans le sous-graphe. Un algorithme de passage de messages entre les sous-

graphes, au travers des noeuds communs, permet d'agréger les diagnostics locaux tout en vérifiant leurs compatibilités.

Les graphes de dépendances causales permettent d'adapter les avancées algorithmiques en matière de théorie des graphes au diagnostic des réseaux. En effet, une fois le graphe construit, le raisonnement ou l'inférence sur ce graphe permettant de déterminer les causes racines des anomalies observées, relève plus du domaine de la théorie des graphes que de la gestion des réseaux. Au delà de la détection des causes racines, de tels modèles graphiques permettent d'expliquer ce qui se passe dans le réseau en identifiant également les conséquences des pannes, ce qui permet aux opérateurs de vérifier les décisions automatiques du modèle et la pertinence du modèle vis-à-vis des connaissances expertes relatives au fonctionnement du réseau. La difficulté majeure de cette approche est l'obtention du graphe. En effet, la construction du graphe nécessite l'acquisition de connaissances très précises relatives à la propagation des pannes et des alarmes. En plus, le graphe de dépendances n'est qu'une image figée du réseau et peut devenir obsolète si le réseau est très dynamique. D'autre part, le graphe de dépendances est un modèle déterministe qui n'est pas toujours adapté à la représentation graphique de la propagation des pannes et d'alarmes qui peut contenir beaucoup de dépendances non déterministes.

2.7 Réseaux bayésiens

Un réseau bayésien ou réseau de croyances est un graphe de dépendances acyclique probabiliste, c'est donc une extension des graphes de dépendances présentés dans la sous-section précédente. Le graphe est dit probabilisé parce que, pour chaque noeud, la force de dépendance entre ce noeud et ses prédécesseurs est quantifiée au moyen d'une distribution de probabilités conditionnelles. Chaque noeud est considéré comme une variable aléatoire prenant ses valeurs ou états dans un ensemble fini connu. La distribution donne la probabilité de chaque état du noeud conditionnellement à toutes les combinaisons d'états possibles de ses prédécesseurs. Cette distribution permet ainsi de représenter et quantifier l'incertitude pouvant exister dans la relation de dépendance entre un noeud et ses prédécesseurs. Cette représentation de l'incertitude constitue la vraie plus-value des réseaux bayésiens par rapport aux graphes de dépendances. Notons au passage que l'incertitude joue un rôle fondamental dans le raisonnement et la prise de décision nécessaires pour automatiser un processus de diagnostic.

Un réseau bayésien donne une représentation graphique factorisée de la loi de probabilités jointe de toutes les variables aléatoires du graphe. Considérons un réseau bayésien (G, Θ) dont la structure G contient n variables $\{X_1, \dots, X_n\}$, et le vecteur de paramètres $\Theta = (\theta_{i,j,k})_{i,j,k}$ tel que $\theta_{i,j,k} = \mathbb{P}(X_i = k | pa(X_i) = j)$ désigne la probabilité que la variable X_i prenne la valeur k sachant que la combinaison des valeurs de ses parents $pa(X_i)$ est j . La loi jointe des variables du réseau peut se factoriser comme le produit suivant, ou le logarithme de cette loi jointe peut se factoriser comme une somme. En effet il est plus facile d'additionner des quantités négatives avec précision que de multiplier de très petites quantités positives.

$$\begin{aligned} \mathbb{P}(G; \Theta) &= \prod_{i=1}^N \mathbb{P}(X_i | pa(X_i)) = \prod_{i=1}^N \theta_{i, pa(X_i), X_i} \\ \mathbb{L}(G; \Theta) &= \sum_{i=1}^N \log \theta_{i, pa(X_i), X_i} \end{aligned} \quad (2.1)$$

Un réseau bayésien encode les dépendances, mais aussi les indépendances entre les variables aléatoires. Étant données deux variables quelconques du réseau, elles sont dé-

pendantes si et seulement si l'une appartient à la couverture de Markov de l'autre, sinon elles sont indépendantes. La couverture de Markov d'une variable dans le réseau est l'union entre l'ensemble des parents ou prédécesseurs de cette variable, l'ensemble de ses fils ou successeurs et l'ensemble des autres parents des fils de cette variable.

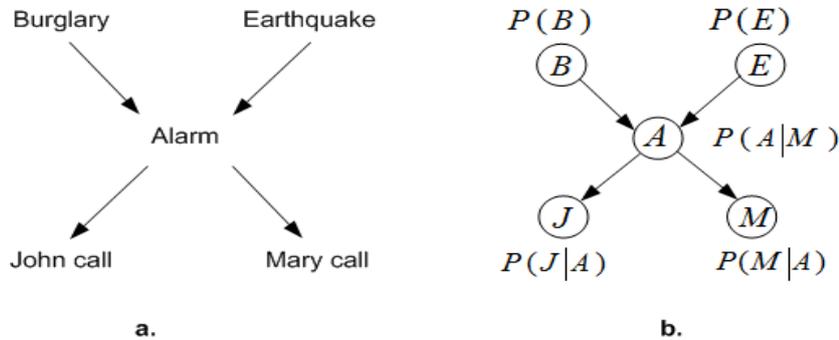


FIGURE 2.13 – Un simple réseau bayésien [RUSSEL et NORVIG \[2003\]](#).

La Figure 2.13.a présente la structure d'un simple réseau bayésien de cinq variables binaires. On imagine la situation suivante : vous avez un nouveau système d'alarme installé à la maison, il est assez fiable pour détecter un cambriolage, mais répond aussi à l'occasion, aux tremblements de terre mineurs. Vous avez également deux voisins, John et Mary, qui ont promis de vous appeler au travail quand ils entendent l'alarme. John appelle toujours quand il entend l'alarme, mais confond parfois la sonnerie de son téléphone avec l'alarme et appelle ensuite. D'autre part, Mary aime plutôt la musique forte et parfois manque complètement l'alarme. Étant donnée une évidence indiquant qui a ou n'a pas appelé, le réseau bayésien présenté par la Figure 2.13.b peut être utilisé pour estimer la probabilité d'un cambriolage [RUSSEL et NORVIG \[2003\]](#). La force de dépendance entre les variables M,A est quantifiée par la distribution de probabilités conditionnelles $\mathbb{P}(A|M)$, celle entre les variables J,A est quantifiée par la distribution $\mathbb{P}(J|A)$ et la distribution $\mathbb{P}(A|B, E)$ mesure l'incertitude dans la corrélation entre la variable A et ses parents B,E. Les variables B et E n'ayant pas de prédécesseurs, leurs distributions $\mathbb{P}(B)$ et $\mathbb{P}(E)$ sont appelées distributions a priori.

Le réseau bayésien présenté par la Figure 2.13.b permet de factoriser la loi jointe des cinq variables selon le produit suivant :

$$\mathbb{P}(B, E, A, J, M) = \mathbb{P}(J|A)\mathbb{P}(M|A)\mathbb{P}(A|B, E)\mathbb{P}(B)\mathbb{P}(E)$$

Un réseau bayésien est une représentation compacte et efficace pour le calcul des distributions de probabilités en exploitant les indépendances conditionnelles. Par exemple, sur la Figure 2.13.b, les variables J et M sont indépendantes conditionnellement à la variable A. Contrairement à un graphe de dépendances, l'inférence sur un réseau bayésien ne nécessite pas de règles déterministes sur lesquelles la procédure d'inférence se base pour déterminer les états des variables non observées connaissant les variables observées. L'inférence sur un réseau bayésien repose sur des algorithmes de renforcement de probabilités ou croyances permettant de calculer des quantités telles que des probabilités marginales conditionnelles ou de répondre à des requêtes de type MPE (Most Plausible Explanation) des variables observées. Concernant le calcul des probabilités marginales conditionnelles, l'inférence sur un réseau bayésien de n variables $G = \{X_1, \dots, X_n\}$

consiste à calculer par exemple, la probabilité jointe de l'ensemble X^m (la lettre "m" signifie "missing") des variables non observées connaissant les valeurs des variables observées X^o (la lettre "o" signifie "observed") telle que $G = X^m \cap X^o$. Si l'on note x^o le vecteur désignant la combinaison des valeurs des variables observées, $\mathbb{P}(X^m|x^o) = \frac{\mathbb{P}(X^m, x^o)}{\mathbb{P}(x^o)}$ et $\mathbb{P}(x^o) = \sum_{X^m} \mathbb{P}(X^m, x^o)$. On peut remarquer que ce calcul nécessite l'énumération de toutes les combinaisons possibles de valeurs des variables non observées, ce qui n'est pas efficace ou même pratiquement impossible si le cardinal de l'ensemble X^m est très grand. Des algorithmes qui exploitent la structure graphique du réseau bayésien pour calculer de façon très efficace une telle probabilité ont été proposés dans PEARL [1985] PEARL [1986] PEARL [1988] MADSEN et JENSEN [1999]. Ces algorithmes sont étudiés en détail dans le Chapitre 3 de cette thèse.

La combinaison de valeurs x^{m*} la plus plausible des variables non observées expliquant les observations, ou "MPE pour Most Probable Explanation", se calcule en maximisant la probabilité conditionnelle $\mathbb{P}(x^m|x^o)$ par rapport à x^m de telle sorte que $\mathbb{P}(x^{m*}|x^o) = \max_{x^m} \mathbb{P}(x^m|x^o)$. L'algorithme somme-produit proposé dans PEARL [1985] PEARL [1986] PEARL [1988] permet de calculer le vecteur x^{m*} , en laissant chaque variable X_i^m non observée, calculer localement et en interaction avec ses noeuds voisins, la fonction (BELief) $BEL^*(x_i^m) = \max_{w_{x_i}} \mathbb{P}(x_i^m, w_{x_i}|x^o)$, ou $W_{X_i} = X^m \setminus \{X_i^m\}$ et $BEL^*(x_i^m)$ désigne la probabilité que les observations x^o soient consistantes avec l'évènement $X_i^m = x_i^m$. Le vecteur $BEL^*(X_i^m)$ contenant les probabilités $BEL^*(x_i^m)$ des valeurs de la variable X_i^m est appelé le maximum à postérieur (MAP) de la variable X_i^m connaissant les observations. Notons que le problème de maximisation de $\mathbb{P}(x^m|x^o)$ par rapport à x^m est décomposé en plusieurs problèmes de maximisation de $\mathbb{P}(x_i^m, w_{x_i}|x^o)$ par rapport w_{x_i} , pour chaque variable X_i^m , en exploitant la structure du réseau bayésien. Ces problèmes de maximisation ne sont pas indépendants puisque pour deux variables distinctes quelconques X_i^m et X_j^m , $W_{X_i} \cup W_{X_j} \neq \emptyset$.

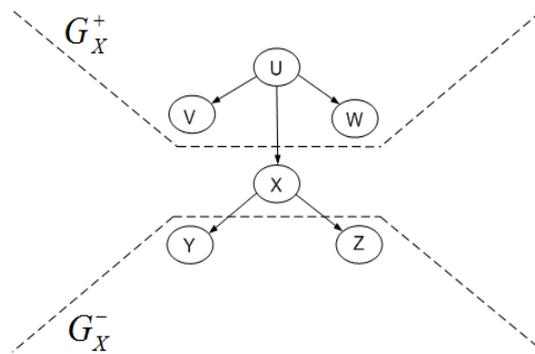


FIGURE 2.14 – Segment d'un réseau bayésien en arbre.

L'algorithme somme-produit permet de calculer efficacement les probabilités marginales conditionnelles ou le maximum à postérieur de n'importe quelle variable ou sous-ensemble de variables du réseau. C'est un algorithme de propagation et de renforcement de croyances qui exploite les indépendances conditionnelles encodées dans le réseau. Considérons par exemple le segment d'un réseau bayésien présenté par la Figure 2.14 et supposons qu'on souhaite calculer la probabilité des valeurs de la variable X étant données d'autres variables observées dans le réseau. L'algorithme somme-produit calcule une telle probabilité notée $BEL(x)$ (pour Belief de $X = x$) connaissant les observa-

tions e_X^- dans le sous-graphe G_X^- enraciné en X , et les observations e_X^+ dans le reste du graphe : $BEL(x) = \mathbb{P}(x|e_X^+, e_X^-) = \alpha\lambda(x)\pi(x)$ tel que $\lambda(x) = \mathbb{P}(e_X^-|x)$, $\pi(x) = \mathbb{P}(x|e_X^+)$ et α est une constante de normalisation. Les expressions de $\lambda(x)$ et $\pi(x)$ sont détaillées dans PEARL [1988] ainsi que dans le Chapitre 3 dédié à l'étude de l'inférence sur les réseaux bayésiens. Pour l'instant nous retiendrons simplement que l'expression factorisée de $BEL(x)$ en un produit de deux termes est vérifiée parce que la représentation graphique du réseau indique que les variables dans G_X^+ sont conditionnellement indépendantes de celles dans G_X^- étant donné X . Chaque nœud X du réseau peut évaluer localement sa probabilité $BEL(x)$ s'il reçoit la contribution $\pi(x)$ de ses parents et la contribution $\lambda(x)$ de ses fils, à la croyance en l'évènement $X = x$ connaissant les observations dans le graphe G . Plusieurs échanges de messages entre le nœud X et ses parents et ses fils sont nécessaires pour évaluer la croyance $BEL(x)$ de X , la croyance $BEL(u)$ d'un parent U de X et la croyance $BEL(y)$ d'un fils Y de X . La propagation des messages est appliquée récursivement jusqu'à ce tous les messages λ et π soient tous absorbés ou traités, et le réseau retrouve ainsi un nouvel état d'équilibre.

Plusieurs techniques de localisation de pannes proposées dans la littérature CHAO et collab. [2001] KATZELA et SCHWARTZ [1995] STEINDER et SETHI [2002] YEMINI et collab. [1996] utilisent des modèles de propagation de pannes qui encodent des relations de dépendances directes entre les pannes et les symptômes. De tels modèles sont appelés graphe de dépendances biparti, c'est-à-dire un graphe dont l'ensemble des sommets peut être partitionné en deux sous-ensembles disjoints telle que chaque arête du graphe ait une extrémité dans un sous-ensemble et l'autre extrémité dans l'autre sous ensemble. Par exemple, les réseaux de Petri introduit dans la Section 2.5 sont des graphes bipartis puisque l'ensemble des places est disjoint de l'ensemble des transitions. Les graphes bipartis sont aussi probabilisés pour obtenir un réseau bayésien. Bien que les relations entre les pannes et les symptômes sont en réalité plus complexes pour être représentées par un graphe causal biparti, l'utilisation de ce type de modèle est justifiée par les arguments suivants MALGORZATA et ADARSHPAL [2004] :

- Effectuer la localisation de pannes avec des représentations complexes est difficile. En général, le problème est NP-hard KATZELA et SCHWARTZ [1995]. Pour éviter cette complexité, les modèles très détaillés sont fréquemment réduits à des graphes bipartis en appliquant une séquence d'opérations de réductions YEMINI et collab. [1996],
- la construction de modèles complexes nécessite une connaissance profonde du système sous-jacent, tandis que l'association symptômes-pannes peut être construite en observant superficiellement le système; (3) certains sous-problèmes de localisation de pannes peuvent être modélisés avec précision par un graphe de dépendances biparti STEINDER et SETHI [2002],
- Certains sous-problèmes de localisation de pannes peuvent être modélisés avec précision par un graphe de dépendances biparti STEINDER et SETHI [2002], par conséquent nécessitent des algorithmes de localisation de pannes adaptés aux modèles à base de graphes bipartis.

Une technique de localisation de pannes adaptée à un modèle à base de graphe causal biparti est la mise à jour incrémentale des hypothèses, plus connue sous l'anglicisme *incremental hypothesis updating (IHU)* STEINDER et SETHI [2004]. La technique IHU utilise un ensemble d'hypothèses, chacune étant une explication complète des symptômes observés. Les hypothèses sont ordonnées selon une métrique de croyance ("*belief metric*"),

b. L'algorithme procède selon une architecture orientée évènement et de façon incrémentale. L'exécution déclenchée par le symptôme numéro i , s_i , crée un ensemble d'hypothèses, H_i , expliquant les symptômes s_1, \dots, s_i . L'ensemble H_i est créé en actualisant H_{i-1} avec une explication du symptôme s_i . Après le traitement du symptôme i , la métrique de croyance b_i associée à l'hypothèse $h_j \in H_i$ représente la probabilité que (1) toutes les pannes appartenant à h_j soient survenues, et (2) h_j explique chaque symptôme observé $s_k \in S_{O,i} = \{s_1, \dots, s_i\}$. Pour intégrer une explication du symptôme s_i dans l'hypothèse de pannes, à l'itération i de l'algorithme, on analyse chaque hypothèse $h_j \in H_{i-1}$. Si h_j peut expliquer le symptôme s_i , on place h_j dans H_i . Dans le cas contraire, h_j doit être étendue en lui ajoutant une panne de H_{s_i} , ou H_{s_i} est un domaine du symptôme s_i . Pour éviter l'augmentation très rapide de la taille de l'ensemble H_i , l'heuristique suivante est utilisée : la panne $f_l \in H_{s_i}$ peut être ajoutée à $h_j \in H_{i-1}$ si et seulement si la taille de h_j , $|h_j|$, est plus petite que $\mu(f_l)$, la taille minimum d'une hypothèse dans H_{i-1} qui contient f_l et explique le symptôme s_i . L'utilisation de cette heuristique est due au fait que la probabilité d'occurrence de multiples pannes simultanées est faible. Tandis que la mise à jour de l'ensemble des hypothèses, $b_i(h_j)$ est calculée itérativement basée sur $b_{i-1}(h_j)$. L'algorithme IHU crée un ensemble d'hypothèses les plus probables, qui peuvent toutes être présentées à l'administrateur du système à diagnostiquer. Plutôt que d'attendre une période de temps déterminée avant de présenter une solution, la technique IHU rend toutes ces hypothèses disponibles sur une base continue, et les met à jour en permanence avec les informations acquises de l'observation d'autres symptômes. Ainsi, l'algorithme est incrémental. Cette caractéristique permet à l'administrateur du système d'initier des actions de correction plus tôt, et d'effectuer des procédures de tests supplémentaires. Puisque plusieurs hypothèses alternatives sont disponibles, il est plus facile de remplacer une solution lorsque la plus probable se révèle inexacte.

Dans [HOUNKONNOU \[2013\]](#), une approche de modélisation semi-automatique est développée, basée sur des motifs ou "patterns" qui décrivent de façon générique les dépendances entre les ressources utilisées par les services du réseau IMS (IP Multimedia Subsystem). Un motif est un réseau bayésien, manuellement construit, mais automatiquement localisé et instancié lorsqu'une panne survient dans un service IMS quelconque. Une approche similaire de modélisation semi-automatique est proposée dans [SANCHEZ et col-lab. \[2015\]](#), appliquée au contexte des réseaux SDN (SoftWare Defined Network) et NFV (Network Functions Virtualization). Elle est basée sur des motifs qui modélisent les composants d'un réseau SDN. Un algorithme interprète la topologie courante donnée par le contrôleur SDN et un motif, basé sur un réseau bayésien, est instancié automatiquement avec d'éventuelles dépendances en plus selon la topologie. Un algorithme d'inférence sur l'instance du réseau bayésien est appliqué pour calculer les probabilités des causes racines conditionnellement aux symptômes observés.

Un modèle à base de réseau bayésien est également proposé dans [KHANAFER et col-lab. \[2008\]](#) pour le diagnostic automatique des réseaux UMTS (Universal Mobile Telecommunication System). Les composants de ce modèle sont les pannes et leurs symptômes. Les pannes sont représentées par des variables aléatoires discrètes binaires (absente ou présente). Les symptômes sont tout aussi bien les alarmes que les KPIs (Key Performances Identifier), c'est-à-dire que les problèmes de connectivité sont considérés, ainsi que les problèmes de performance. Les alarmes sont également représentées par des variables discrètes binaires. Les KPIs sont modélisées par des variables discrètes multivaluées, chaque état représentant un intervalle continu de valeurs pour les KPIs conti-

nues qui ont été discretisées. Le modèle proposé est un réseau bayésien naïf puisqu'il suppose que les symptômes observés sont indépendants les uns des autres, ce qui n'est pas toujours vrai en réalité car il est possible que plusieurs symptômes en engendrent d'autres. Le modèle graphique est construit comme suit : pour un noeud parent P , dont les états sont les causes racines pouvant déclencher les symptômes S_1, \dots, S_n , représentant les fils de P , le modèle suppose qu'une seule panne survient à la fois et qu'il n'y a pas de dépendances entre les symptômes. La simplicité de ce modèle dit naïf est très bénéfique pour l'inférence, et l'impact sur les résultats de diagnostic est assez mineur. En effet, les paramètres nécessaires pour ce modèle sont les probabilités a priori $\mathbb{P}(P = p_k)$ pour chaque panne p_k , et les probabilités conditionnelles $\mathbb{P}(S_i|P = p_k)$ pour chaque symptôme S_i . Le diagnostic basé sur le calcul des probabilités conditionnelles des pannes connaissant les symptômes observés e , se calcule facilement avec la formule de Bayes : $\mathbb{P}(P = p_k|e) = \mathbb{P}(P = p_k) \prod_{i=1}^n \mathbb{P}(S_i|P = p_k) / \mathbb{P}(e)$. Il faut remarquer dans l'équation précédente que la probabilité conjointe des symptômes conditionnellement à une panne s'écrit comme le produit des probabilités de chaque symptôme conditionnellement à la panne, traduisant l'hypothèse d'indépendance entre les symptômes faite dans ce réseau bayésien naïf.

La modélisation de la propagation de pannes et d'alarmes dans un réseau de télécommunications requiert la représentation des connaissances relatives aux dépendances causales, la représentation des incertitudes puisque les relations de causes à effets ne sont pas toujours déterministes, la gestion de données manquantes, ambiguës ou même erronées. Un réseau bayésien est un formalisme très approprié pour une telle modélisation. En effet, un réseau bayésien : (1) modélise la chaîne complète de la propagation des pannes en incluant les causes intermédiaires qui relient indirectement les causes racines aux anomalies observées très souvent sous forme d'alarmes; (2) représente les dépendances et les indépendances entre les variables du phénomène de propagation de pannes; (3) exprime et quantifie l'incertitude due au caractère non déterministe de certaines relations de causes à effets; (4) intègre d'importantes propriétés calculatoires et cognitives permettant de développer des raisonnements probabilistes exacts sur les connaissances encodées; (5) ces propriétés confèrent aux réseaux bayésien la capacité de faire face aux données manquantes, ambiguës et même erronées ou bruitées. D'autre part, un réseau bayésien peut être construit de façon modulaire ou incrémentale et les connaissances sont représentées de façon compacte et intelligible pour un expert humain. Cette représentation compacte de l'espace des hypothèses de diagnostic permet de trouver une solution pour toute combinaison d'observations sur un réseau de télécommunications. C'est un formalisme très polyvalent. Il peut tout aussi bien être utilisé pour le diagnostic réactif de pannes que pour leur prédiction ou diagnostic proactif, pour l'optimisation, l'analyse de données, l'apprentissage artificiel supervisé et non supervisé.

Le développement d'un réseau bayésien pour le diagnostic d'un réseau de télécommunication de grande dimension requiert des connaissances très approfondies sur ce réseau. Mais ceci n'est pas forcément un inconvénient puisqu'il est possible de construire un modèle initial très simple du système et d'enrichir ou compléter automatiquement ce modèle initial ensuite, en appliquant un algorithme d'apprentissage artificiel à partir des données générées par les composants du réseau. De notre point de vue, nous pensons que l'inconvénient majeur des réseaux bayésiens est la détermination très difficile des probabilités a priori exactes, pourtant ces a priori ont une influence forte sur les probabilités a posteriori à partir desquelles les décisions finales de diagnostic sont prises. Il

est difficile de détecter les pannes avec de faibles a priori, c'est-à-dire les pannes qui se produisent très peu souvent.

2.8 Discussions et conclusion

Le diagnostic de pannes dans un réseau de télécommunications de grande échelle est un problème complexe qui intéresse aussi bien les opérateurs de télécommunications que la communauté de l'intelligence artificielle. Ce problème a fait l'objet de nombreuses recherches dans le passé et différentes approches ont été proposées : (1) les systèmes experts PAU [1986], qui encodent des raisonnements spécialisés sur des tâches de diagnostic précises hébergées dans des applications informatiques, (2) les approches à base de modèle JAKOBSON et WEISSMAN [1995] GARDNER et HARLE [1997] qui développent des raisonnements sur une représentation explicite du réseau, (3) les algorithmes d'apprentissage automatique tels que les réseaux de neurones artificiels GARDNER et HARLE [1997] HERMANN [2002], le raisonnement à base de cas « case-based reasoning » LEWIS [1993] qui infère les diagnostics en s'appuyant sur l'expérience automatiquement ou artificiellement acquise lors des diagnostics passés. Table 2.1 récapitule ces différentes méthodes de diagnostic.

Les systèmes experts à base de règles sont les premières méthodes de diagnostic utilisées dans l'industrie des télécommunications. Bien qu'efficaces pour la résolution des problèmes déjà rencontrés, les systèmes experts à base de règles sont peu flexibles, difficiles et fastidieux à maintenir. C'est une approche déterministe appliquée pour décrire la propagation de pannes dans un réseau qui est pourtant un phénomène globalement non déterministe.

Quant aux méthodes de diagnostic essentiellement basées sur l'apprentissage artificiel telles que les réseaux de neurones, les cartes auto-organisatrices, l'apprentissage statistique ou la fouille de données, ce sont des "boîtes noires" généralement dites "aveugles" puisqu'elles ignorent les informations relatives à la topologie et au comportement du réseau. Le diagnostic avec ces méthodes d'apprentissage repose essentiellement sur une approximation ou une interpolation faite en analysant les données des précédents diagnostics. Si nous faisons le rapprochement avec le domaine du diagnostic médical, de telles méthodes peuvent être assimilées à celles d'un "faux médecin" qui n'a pas de connaissances sur l'anatomie du corps humain, mais qui peut quand même diagnostiquer des maladies en étudiant de nombreux cas de diagnostic fait par les médecins compétents.

Comme avec le diagnostic médical, le diagnostic précis et efficace de pannes dans un réseau requiert la prise en compte de sa topologie et de son fonctionnement durant les raisonnements. C'est justement le but des approches de diagnostic à base d'un modèle du réseau. Ces approches procèdent en deux étapes : (1) la topologie et le comportement du réseau ou propagation de pannes et d'alarmes sont modélisés, et le modèle laisse transparaître explicitement la capacité d'un composant (logiciel ou matériel) à influencer l'état d'un autre, (2) des algorithmes de raisonnement orientés événements ou objets sont ensuite dérivés du modèle. Plusieurs contributions dans la littérature se basent sur cette approche. On peut citer notamment les techniques de diagnostic à base de modèle développées dans STEINDER et SETHI [2002] STEINDER et SETHI [2004] et leur généralisation aux algorithmes de théorie de graphes. Il s'agit par exemple de représenter au moyen d'un graphe de dépendances causales la chaîne des relations de causes à effets entre les pannes

TABLE 2.1 – Tableau récapitulatif des différentes méthodes de diagnostic.

Méthodes	Avantages	Inconvénients
RBE	règles spécialisées efficacité pour les problèmes connus	peu flexible maintenance difficile induction limitée
CBR	cas spécialisés capacité d'induction résolution de nouveaux problèmes	inefficacité temporelle façonné pour le domaine
MLT	capacité d'induction résolution de nouveaux problèmes	boîte noire inintelligibilité
MBA	modélisation du réseau large portée de raisonnement possibilité d'apprendre	complexité de modélisation

Légende de la Table 2.1 : RBE (Rule-Based Expert system), CBR (Case-Based Reasoning), MLT (Machine Learning Techniques), MBA (Model-Based Approach).

ou causes racines, les causes intermédiaires et les symptômes ou alarmes observées. De tels graphes sont appropriés aux algorithmes d'inférence ou de raisonnement tels que ceux développés dans [JINGXIAN et collab. \[2010\]](#) [JINGXIAN et collab. \[2011\]](#) [GROSCLAUDE \[2008\]](#). D'autres méthodes vont plus loin en probabilisant le graphe de dépendances pour obtenir un réseau bayésien et apporter de la robustesse et du non déterminisme en appliquant un raisonnement probabiliste [PEARL \[1988\]](#). Cependant, les approches de diagnostic à base de modèle présentent deux difficultés principales : (1) obtenir le modèle explicite du réseau et, (2) adapter le modèle aux évolutions du réseau.

Les travaux menés dans le cadre de cette thèse apportent des contributions à ces problèmes. Modéliser la topologie et le fonctionnement (propagation de pannes) d'un réseau requiert la représentation des connaissances relatives aux dépendances causales, à l'incertitude (la propagation de pannes n'étant pas déterministe), à l'absence de certaines données, à l'imprécision ou l'ambiguïté de certaines données générées par les équipements, et à l'évolution temporelle éventuellement. Il faut aussi garder à l'esprit que le modèle de représentation doit être suffisamment intelligible pour permettre aux opérateurs de comprendre les diagnostics qu'il produira. D'autre part, le modèle ne doit pas se restreindre à une image figée du réseau qui est par nature un système avec une architecture matérielle et logicielle assez évolutive.

Nous proposons une méthode générique qui facilite la modélisation de la topologie et du comportement (propagation de pannes) d'un réseau de télécommunications. C'est un modèle en trois couches, basé sur les réseaux bayésiens, qui intègre les propriétés de modularité et d'extensibilité lui permettant de se reconfigurer et de s'adapter facilement aux éventuels changements sur la topologie et le comportement du réseau de télécommunications. Le principe que nous utilisons est le suivant : modéliser le comportement d'un réseau revient à modéliser le comportement des équipements ou des composants qui le constituent. Le comportement de chaque composant est modélisé par deux graphes de dépendances acycliques (DAG) : un DAG qui modélise la propagation locale de pannes et d'alarmes sur ce composant, et un autre graphe qui modélise la propagation distri-

buée de pannes et d'alarmes entre ce composant et les composants voisins auxquels il est connecté. Les deux graphes sont interconnectés puisqu'ils se partagent certaines variables. Le modèle générique est construit en trois couches. La couche 1 contient l'ensemble des DAGs modélisant la propagation distribuée de pannes et d'alarmes. La couche 2 contient l'ensemble des DAGs modélisant la propagation locale de pannes et d'alarmes. Les couches 1 et 2 sont construites de telle sorte que leur combinaison donne un large DAG G_r qui modéliserait la propagation de pannes et d'alarmes dans le réseau entier. Mais la séparation entre la propagation locale et la propagation distribuée, permet de modéliser également la topologie du réseau, et surtout décompose ce large DAG en plusieurs sous-DAGs interconnectés, apportant ainsi des propriétés de modularité et d'extensibilité au modèle générique. La couche 3 quant à elle est une représentation en arbre de jonction du DAG G_r sur laquelle on peut appliquer un algorithme d'inférence sans se soucier de la complexité des topologies des DAGs des couches inférieures. Cette méthodologie favorise la modélisation incrémentale et modulaire d'un réseau de télécommunications, ce qui permet d'ajouter facilement de nouveaux composants dans le modèle ou de retirer un composant existant du modèle lorsque l'opérateur le souhaite.

Cette thèse développe et présente également une application concrète de cette méthodologie générique à la modélisation et au diagnostic automatique du réseau d'accès optique GPON (Gigabit Passive Optical Network) de type FTTH (Fiber To The Home).

Chapitre 3

Algorithmes d'inférence exacte sur un réseau bayésien

« I consider the word probability as meaning the state of mind with respect to an assertion, a coming event, or any other matter on which absolute knowledge does not exist. »

August De Morgan, 1838

Sommaire

3.1 Introduction	45
3.2 Quelques bases en théorie des probabilités	45
3.3 Opération de marginalisation	47
3.4 Algorithme de somme-produit	48
3.4.1 Topologie en arbre	48
3.4.2 Topologie en polyarbre	51
3.4.3 Topologie avec des boucles acycliques	56
3.5 Algorithme d'inférence sur un arbre de jonction	57
3.5.1 Arbre de jonction et potentiel	57
3.5.2 Propagation d'évidences	59
3.6 Discussion et conclusion	63

3.1 Introduction

Effectuer le diagnostic de pannes sur un réseau de télécommunication à base d'un modèle graphique probabiliste connu, revient à calculer la probabilité a posteriori des pannes ou causes racines connaissant les observations collectées sur ce réseau. Ce calcul probabiliste encore appelé raisonnement ou inférence probabiliste devient très difficile lorsque la topologie graphique du modèle est complexe. Dans ce chapitre, nous étudions les algorithmes d'inférence probabiliste exacte. L'étude débute par les algorithmes d'inférence simples et progresse vers les plus complexes. Nous commençons par rappeler quelques bases de la théorie des probabilités dans la Section 2. Ensuite, l'inférence réduite à une simple opération de marginalisation est traitée dans la Section 3. Dans la Section 4, l'algorithme de somme-produit est détaillé sur chacune des trois principales topologies graphiques de réseau bayésien. La Section 5 traite de l'inférence sur une représentation graphique simplifiée et équivalente à un réseau bayésien appelée arbre de jonction. Nous terminons ce chapitre par une discussion et une conclusion dans la Section 6.

3.2 Quelques bases en théorie des probabilités

Plusieurs chercheurs COX [1946] GOOD [1950] SAVAGE [1972] DEFINETTI [1972] du vingtième siècle ont écrit des ensembles différents de propriétés que doivent satisfaire le degré de croyance en un évènement ou la probabilité que cet évènement survienne. De chacun de ces ensembles ont été dérivés de mêmes règles appelées règles de la théorie des probabilités. On note deux règles fondamentales à partir desquelles d'autres règles seront plus tard dérivées.

La règle de somme : pour tout évènement a et son complément $\neg a$,

$$\mathbb{P}(a) + \mathbb{P}(\neg a) = 1 \quad (3.1)$$

La règle de produit : pour deux évènements quelconques a et b ,

$$\mathbb{P}(a, b) = \mathbb{P}(b|a)\mathbb{P}(a) \quad (3.2)$$

ou $\mathbb{P}(b|a)$ désigne la probabilité que l'évènement b se produise sachant que l'évènement a est déjà survenu. D'autres règles seront par la suite exprimées en terme de variables aléatoires plutôt que d'évènements. Une variable aléatoire discrète prend ses valeurs dans un ensemble fini d'états ou d'évènements indépendants. Le théorème de Thomas Bayes BAYES [1958] est une règle très utilisée : pour deux variables aléatoires quelconques A et B, la probabilité conditionnelle de A sachant B est donnée par,

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A, B)}{\mathbb{P}(B)} \quad \text{tel que } \mathbb{P}(B) \neq 0 \quad (3.3)$$

Notons que $\mathbb{P}(A)$ et $\mathbb{P}(A|B)$ sont respectivement la distribution de probabilités de A avant la connaissance de B et après la connaissance de B. $\mathbb{P}(A)$ et $\mathbb{P}(A|B)$ sont respectivement appelées distribution a priori et a posteriori de A. La règle de chaîne qui est la généralisation de la règle fondamentale de produit de l'Équation (3.2) : pour un ensemble de n variables aléatoires quelconques on peut écrire,

$$\mathbb{P}(X_1, \dots, X_n) = \prod_{i=1}^n \mathbb{P}(X_i | X_1, \dots, X_{i-1}) \quad (3.4)$$

La généralisation de la règle fondamentale de somme formalisée par l'Équation (3.1), aussi appelée marginalisation : pour deux variables aléatoires A et B, on peut écrire,

$$\mathbb{P}(A) = \sum_b \mathbb{P}(A, b) \quad (3.5)$$

ou b désigne l'ensemble des valeurs que peut prendre la variable aléatoire B. La distribution de A peut aussi s'obtenir par la règle d'expansion définie comme suit :

$$\mathbb{P}(A) = \sum_b \mathbb{P}(A|b)\mathbb{P}(b) \quad (3.6)$$

Les algorithmes d'inférence probabiliste que nous aborderons dans les sections suivantes de ce chapitre s'appuient sur les règles énumérées ci-dessus. Considérons un réseau bayésien $\{X_1, \dots, X_n\}$ de n variables aléatoires discrètes et k -multivaluées chacune. Appelons X^o (avec "o" comme "observed") un sous-ensemble de variables observées du réseau et X^m (avec "m" comme "missing") l'autre sous-ensemble de variables non observées ou manquantes. Nous développons chacun des algorithmes d'inférence étudiés en calculant la distribution de probabilité d'une variable non observée $X \in X^m$ connaissant les valeurs ou les états des variables observées X^o , c'est-à-dire la distribution de probabilité a posteriori ou conditionnelle $\mathbb{P}(X|x^o)$ telle que x^o désigne la combinaison de valeurs prises par les variables observées.

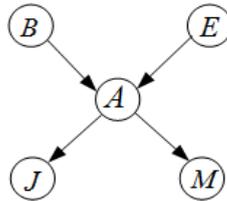


FIGURE 3.1 – Un exemple simple d'un réseau bayésien

TABLE 3.1 – Paramètres du réseau bayésien présenté à la Figure 3.1.

B	$\mathbb{P}(+b) = .001$
E	$\mathbb{P}(+e) = .002$
J	$\mathbb{P}(+j +a) = .950, \mathbb{P}(+j \neg a) = .100$
M	$\mathbb{P}(+m +a) = .990, \mathbb{P}(+m \neg a) = .300$
A	$\mathbb{P}(+a +b,+e) = .999, \mathbb{P}(+a +b,\neg e) = .710$
	$\mathbb{P}(+a \neg b,+e) = .940, \mathbb{P}(+a \neg b,\neg e) = .050$

Chaque algorithme d'inférence est illustré avec le réseau bayésien R présenté par la Figure 3.1. Il s'agit d'un petit réseau bayésien de cinq variables aléatoires binaires. Les paramètres du réseau R, c'est-à-dire les distributions de probabilité de chaque variable conditionnellement à ses prédécesseurs ou parents dans le graphe, sont données par la Table 3.1. Chaque algorithme d'inférence calcule les distributions de probabilité a posteriori $\mathbb{P}(A|+j,+m)$, $\mathbb{P}(B|+j,+m)$ et $\mathbb{P}(E|+j,+m)$, c'est-à-dire les distributions des variables non observées A, B, et E connaissant les variables observées J qui prend la valeur $+j$ et M qui prend la valeur $+m$. La combinaison des variables observées est donc $x^o = \{+j,+m\}$.

3.3 Opération de marginalisation

La marginalisation est la forme d'inférence probabiliste exacte la plus simple. Elle consiste à énumérer de façon exhaustive tous états possibles de chacune des variables non observées. C'est l'application du théorème de Bayes (3.3) et de la règle généralisée de somme formalisée par l'Équation (3.5).

$$\mathbb{P}(X^m|x^o) = \frac{\mathbb{P}(X^m, x^o)}{\mathbb{P}(x^o)} = \frac{\mathbb{P}(X^m, x^o)}{\sum_{x^m} \mathbb{P}(x^m, x^o)} \quad (3.7)$$

Nous rappelons X^m et x^m représentent respectivement l'ensemble des variables manquantes et la combinaison des valeurs possibles des ces variables. La probabilité $\mathbb{P}(x^o)$ est appelée vraisemblance des observations x^o . Nous verrons dans le Chapitre 4 que cette quantité est très utile pour l'apprentissage automatique ou l'estimation des probabilités conditionnelles d'un réseau bayésien à partir des données observées. En effet, la vraisemblance des observations permet de quantifier l'adéquation ou la convenance du réseau bayésien à l'échantillon de données d'apprentissage. Dans l'Équation (3.7), la quantité $\mathbb{P}(X^m, x^o)$ se calcule en remplaçant dans la distribution jointe de toutes les variables du réseau bayésien, donnée par les paramètres de ce réseau, les variables observées par leurs valeurs respectives. Nous noterons que $\mathbb{P}(X^m|x^o)$ est la distribution de probabilité jointe a posteriori des variables non observées connaissant les observations x^o . À partir de cette loi jointe, on peut à nouveau faire une opération de marginalisation pour extraire la distribution de probabilité d'une quelconque variable X non observée conditionnelle à x^o . Nous noterons $X^m \setminus \{X\}$ le complémentaire de $\{X\}$ dans l'ensemble X^m .

$$\mathbb{P}(X|x^o) = \sum_{w_x} \mathbb{P}(X, w_x|x^o) \quad \text{tel que } W_x = X^m \setminus \{X\} \quad (3.8)$$

Pour illustrer l'inférence par une simple opération de marginalisation, calculons les distributions a posteriori de chacune des variables non observées A, B et E du réseau bayésien R de la Figure 3.1 et de la Table 3.1. Il faut tout d'abord calculer la loi jointe du réseau conditionnellement aux évidences $x^o = (+j, +m)$. Cette loi jointe a posteriori est une fonction des variables non observées A, B et E.

$$\mathbb{P}(R|x^o) = \mathbb{P}(X^m, x^o) = \mathbb{P}(A, B, E, +j, +m) = \mathbb{P}(+j|A)\mathbb{P}(+m|A)\mathbb{P}(A|B, E)\mathbb{P}(B)\mathbb{P}(E)$$

Ensuite on peut marginaliser sur cette loi jointe pour extraire la distribution de probabilité de chaque variable non observée conditionnellement aux évidences x^o .

$$\mathbb{P}(B|x^o) = \sum_{a,e} \mathbb{P}(R|x^o) = [0.991 \quad 0.009]_{(b)}$$

$$\mathbb{P}(E|x^o) = \sum_{a,b} \mathbb{P}(R|x^o) = [0.977 \quad 0.023]_{(e)}$$

$$\mathbb{P}(A|x^o) = \sum_{b,e} \mathbb{P}(R|x^o) = [0.365 \quad 0.635]_{(a)}$$

Nous noterons que le calcul de la loi jointe $\mathbb{P}(R|x^o)$ n'est pas un produit matriciel, mais plutôt un produit tensoriel de fonctions. Par exemple le produit de la distribution $f(I) = [i_1, i_2]_{(i)}$ de la variable aléatoire binaire I et celle de la variable aléatoire binaire J, $g(J) = [j_1, j_2]_{(j)}$ est la fonction $h(I, J)$ définie par une matrice à deux dimensions. La première dimension est I et la seconde est J.

$$h(I, J) = [i_1, i_2]_{(i)} [j_1, j_2]_{(j)} = \begin{bmatrix} i_1 j_1 & i_1 j_2 \\ i_2 j_1 & i_2 j_2 \end{bmatrix}_{(i,j)} \quad (3.9)$$

Bien que très simple, la marginalisation est une opération très coûteuse calculatoirement lorsque le nombre de variables aléatoires non observées est important. La complexité calculatoire de cette opération augmente avec le nombre de variables aléatoires non observées et la dimension de l'espace des états de chacune des variables non observées. En effet, pour m variables non observées et k -multivaluées chacune, le nombre de combinaisons à énumérer est k^m . De ce fait, l'inférence par une simple opération de marginalisation reste limitée à de très petites topologies de réseau bayésien de quelques variables aléatoires, d'où l'intérêt d'étudier des algorithmes d'inférence qui passent l'échelle.

3.4 Algorithme de somme-produit

Nous nous intéressons dans cette section à l'algorithme d'inférence exacte de somme-produit [PEARL \[1985\]](#) [PEARL \[1986\]](#) [PEARL \[1988\]](#) sur diverses topologies de réseaux bayésiens.

3.4.1 Topologie en arbre

Considérons le fragment d'un réseau bayésien en arbre représenté par la Figure 3.2, c'est-à-dire que chaque noeud ou variable aléatoire X du réseau bayésien a un seul parent ou prédécesseur. Par contre plusieurs noeuds peuvent avoir un prédécesseur commun.

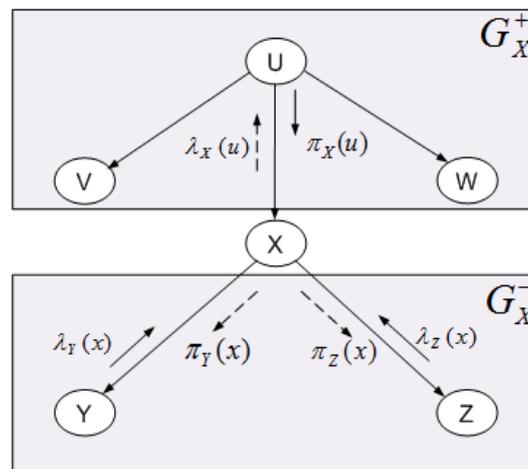


FIGURE 3.2 – Fragment d'un réseau bayésien en arbre.

La croyance aux différents états possibles de la variable X dépend des observations collectées dans deux sous-ensembles distincts du réseau [PEARL \[1988\]](#). Le sous-graphe contenant les descendants de X noté G_X^- et le sous-graphe contenant les ascendants de X et leurs autres fils noté G_X^+ , obtenu en supposant que l'arc $U \rightarrow X$ est supprimé (voir la Figure 3.2). Appelons e_X^- l'ensemble des évidences collectées dans G_X^- et e_X^+ l'ensemble des évidences collectées dans G_X^+ . Puisque le réseau bayésien a une topologie en arbre, toutes les évidences provenant de G_X^+ se résument au noeud U qui est l'unique parent du noeud X . De même toutes les évidences provenant de G_X^- se résument aux noeuds fils ou successeurs de X que sont Y et Z . La croyance $BEL(x)$ ("BELief") en l'évènement $X = x$

peut donc s'écrire telle que dans l'Équation (3.10) développée ci-dessous PEARL [1988] :

$$\begin{aligned}
\text{BEL}(x) &= \mathbb{P}(x|e_X^+, e_X^-) = [\mathbb{P}(e_X^+, e_X^-)]^{-1} \mathbb{P}(e_X^+, x, e_X^-) \\
&= \alpha \mathbb{P}(e_X^- | e_X^+, x) \mathbb{P}(x | e_X^+) \\
&= \alpha \mathbb{P}(e_X^- | x) \mathbb{P}(x | e_X^+) \\
&= \alpha \lambda(x) \pi(x)
\end{aligned} \tag{3.10}$$

tel que $\alpha = [\mathbb{P}(e_X^+, e_X^-)]^{-1}$ est une constante de normalisation puisque ne dépendant pas de la valeur de x , $\lambda(x) = \mathbb{P}(e_X^- | x)$ et $\pi(x) = \mathbb{P}(x | e_X^+)$. Notons que dans l'Équation (3.10), $\mathbb{P}(e_X^- | e_X^+, x) = \mathbb{P}(e_X^- | x)$ par ce que les variables dans G_X^- sont indépendantes de celles dans G_X^+ connaissant X . La croyance en l'évènement $X = x$ se factorise en un produit de deux termes, la contribution $\lambda(x) = \mathbb{P}(e_X^- | x)$ apportée par les fils Y et Z de X et la contribution $\pi(x) = \mathbb{P}(x | e_X^+)$ apportée par l'unique parent U de X . Puisque les évidences e_X^- provenant de G_X^- se résument en Y et Z , on peut écrire $e_X^- = e_Y^- \cup e_Z^-$. Par ailleurs, les enfants Y et Z sont indépendants connaissant leur parent commun X . Par conséquent la contribution des enfants de X à la croyance en l'évènement $X = x$ du parent se factorise.

$$\begin{aligned}
\lambda(x) &= \mathbb{P}(e_X^- | x) = \mathbb{P}(e_Y^-, e_Z^- | x) = \mathbb{P}(e_Y^- | x) \mathbb{P}(e_Z^- | x) \\
&= \lambda_Y(x) \lambda_Z(x)
\end{aligned} \tag{3.11}$$

tel que $\lambda_Y(x) = \mathbb{P}(e_Y^- | x)$ et $\lambda_Z(x) = \mathbb{P}(e_Z^- | x)$. Puisque U est le seul parent de X , la contribution de U à la croyance en l'évènement $X = x$ peut se développer en appliquant la règle d'expansion de l'Équation (3.6). Connaissant U , la variable X est indépendante de toutes les autres variables de G_X^+ .

$$\begin{aligned}
\pi(x) &= \mathbb{P}(x | e_X^+) = \sum_u \mathbb{P}(x | e_X^+, u) \mathbb{P}(u | e_X^+) = \sum_u \mathbb{P}(x | u) \mathbb{P}(u | e_X^+) \\
&= \sum_u \mathbb{P}(x | u) \pi_X(u)
\end{aligned} \tag{3.12}$$

Notons que $\mathbb{P}(x | u)$ est donnée par la distribution de probabilités conditionnelles $\mathbb{P}(X|U)$ connue. La quantité $\pi_X(u) = \mathbb{P}(u | e_X^+)$ peut être calculée localement par le noeud U et délivrée comme un message à son fils X . Finalement, en remplaçant les Équations (3.11) et (3.12) dans l'expression de $\text{BEL}(x)$ donnée par l'Équation (3.10), on obtient l'expression finale de la probabilité a posteriori de l'évènement $X = x$ connaissant les observations x^o collectées dans le graphe tout entier.

$$\text{BEL}(x) = \mathbb{P}(x | x^o) = \alpha \lambda_Y(x) \lambda_Z(x) \sum_u \mathbb{P}(x | u) \pi_X(u) \tag{3.13}$$

Dans l'Équation (3.13), les expressions des quantités $\lambda_Y(x)$, et $\lambda_Z(x)$ respectivement envoyées par les noeuds Y et Z à leur parent commun X sont calculées dans PEARL [1988]. Considérons dans un premier de temps le cas de $\lambda_Y(x)$. Si la variable Y est observée alors $\lambda_Y(x) = \mathbb{P}(e_Y^- | x)$ est réduite à la probabilité conditionnelle connue $\mathbb{P}(y' | x)$ telle que y' est la valeur observée de Y . Si Y n'est pas observé, on conditionne $\mathbb{P}(e_Y^- | x)$ à Y en utilisant la règle d'expansion comme suit :

$$\begin{aligned}
\lambda_Y(x) &= \sum_y \mathbb{P}(e_Y^- | y, x) \mathbb{P}(y | x) \\
&= \sum_y \mathbb{P}(e_Y^- | y) \mathbb{P}(y | x) \\
&= \sum_y \lambda(y) \mathbb{P}(y | x) = M_{y|x} \bullet \lambda(y)
\end{aligned} \tag{3.14}$$

ou $M_{y|x} = \mathbb{P}(y|x)$ est une probabilité conditionnelle connue et $M_{y|x} \cdot \lambda(y)$ est le produit scalaire entre les vecteurs $M_{y|x}$ et $\lambda(y)$. De même, si z' est la valeur observée de Z , $\lambda_Z(x) = \mathbb{P}(z'|x) = M_{z'|x}$, dans le cas contraire $\lambda_Z(x) = M_{z|x} \cdot \lambda(z)$.

Une fois que la croyance $BEL(x)$ du noeud X est calculée par l'Équation (3.13), le noeud X doit à présent envoyer un message à ses fils non observés. La quantité $\pi_Z(x) = \mathbb{P}(x|e_Z^+)$ est envoyée au fils Z si Y est observé, ou la quantité $\pi_Y(x) = \mathbb{P}(x|e_Y^+)$ est envoyée au fils Y si c'est plutôt Z qui est observé. Nous noterons que $e_Z^+ = e_X^+ \cup e_Y^-$ et $e_Y^+ = e_X^+ \cup e_Z^-$. Ainsi, on peut utiliser la formule de Bayes pour développer les expressions $\pi_Z(x)$ et $\pi_Y(x)$ comme suit :

$$\begin{aligned}
 \pi_Y(x) &= \mathbb{P}(x|e_Y^+) = \mathbb{P}(x|e_X^+, e_Z^-) \\
 &= [\mathbb{P}(e_X^+, e_Z^-)]^{-1} \mathbb{P}(e_Z^-|x, e_X^+) \mathbb{P}(x|e_X^+) \\
 &= \alpha \mathbb{P}(e_Z^-|x) \mathbb{P}(x|e_X^+) \\
 &= \alpha \lambda_Z(x) \pi(x) \\
 &= \alpha \frac{\lambda_Y(x) \lambda_Z(x) \pi(x)}{\lambda_Y(x)} = \alpha \frac{BEL(x)}{\lambda_Y(x)}
 \end{aligned} \tag{3.15}$$

ou $\alpha = [\mathbb{P}(e_X^+, e_Z^-)]^{-1}$ est une constante de normalisation. En suivant le même raisonnement ci-dessus, si c'est plutôt Y qui est observé, $\pi_Z(x) \propto \frac{BEL(x)}{\lambda_Z(x)}$.

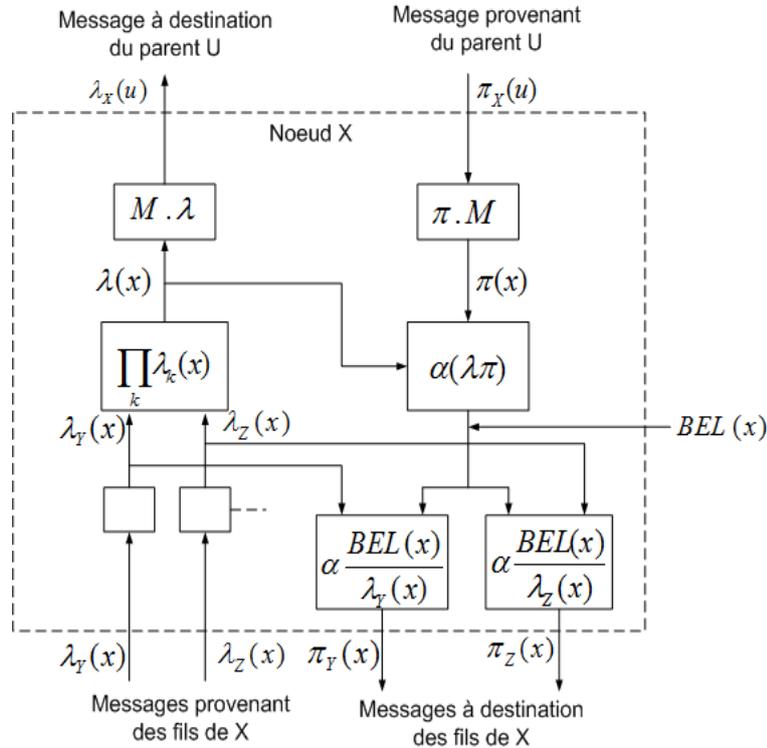


FIGURE 3.3 – Calcul local de la croyance d'un noeud dans un réseau bayésien en arbre.

La Figure 3.3 extraite du livre PEARL [1988], résume les calculs effectués au niveau d'un noeud quelconque X pour évaluer la croyance en l'évènement $X = x$ ou la probabilité conditionnelle de l'évènement $X = x$ connaissant les observations ou évidences x^o dans le réseau. Notons que chaque variable X du réseau bayésien se comporte comme un processeur autonome et peut donc évaluer localement sa croyance $BEL(x) = \mathbb{P}(x|x^o)$ en

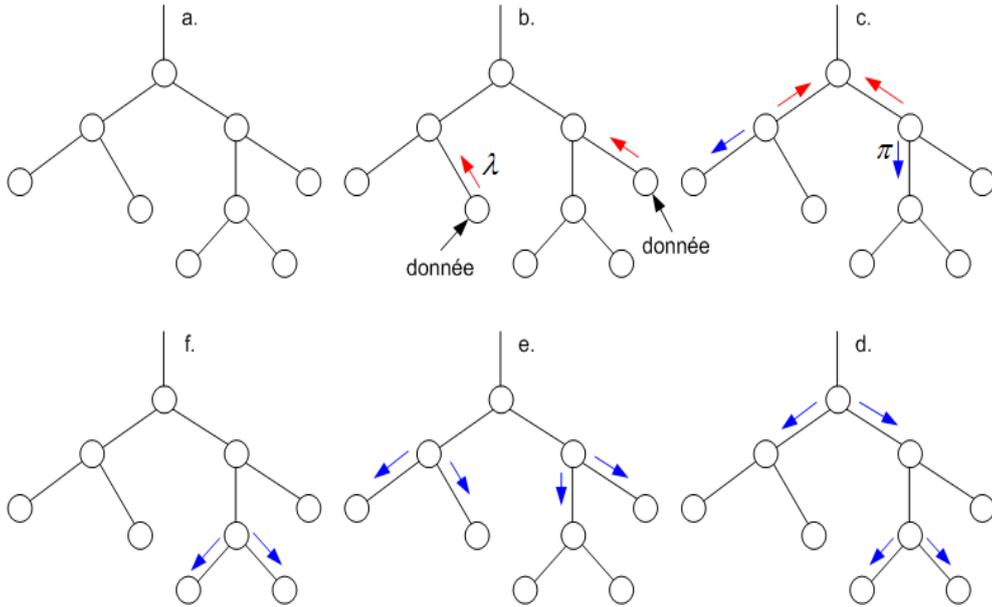


FIGURE 3.4 – Mécanisme de propagation d'évidences dans un réseau bayésien en arbre.

l'évènement $X = x$ s'il reçoit la contribution de son parent et les contributions de ses fils.

La Figure 3.4 extraite du livre PEARL [1988] décrit le mécanisme de propagation des évidences à travers les noeuds d'un réseau bayésien en arbre. La Figure 3.4.a décrit un état d'équilibre du réseau bayésien, c'est-à-dire un état durant lequel aucun calcul probabiliste n'est en cours et aucune donnée ou observation n'est collectée. Cet état d'équilibre est perturbé par l'arrivée de deux données (c'est-à-dire deux variables observées) sur la Figure 3.4.b. Chacune de ces deux variables calcule et délivre une valeur λ à son parent. Dans la Figure 3.4.c, chaque parent calcule et délivre à son tour une valeur λ à son parent et une valeur π à tous ses fils sauf au fils qui lui avait précédemment délivré le message λ . Le noeud à la racine de l'arbre reçoit donc deux valeurs λ distinctes. Dans la Figure 3.4.d, le noeud à la racine calcule et délivre une valeur π à chacun de ses deux fils. Le calcul distribué sur le graphe se poursuit en respectant les règles de propagation de la Figure 3.3 et se termine lorsque chaque noeud de l'arbre n'ayant pas de successeurs reçoit tous les messages π de son parent. Le réseau bayésien retrouve donc l'état d'équilibre initial puisque chaque feuille de l'arbre ne délivre pas un message λ à son unique parent par lequel il reçoit un message π .

3.4.2 Topologie en polyarbre

La seule et unique différence entre un réseau bayésien en arbre et un réseau bayésien en polyarbre est qu'un noeud X peut avoir plusieurs prédécesseurs dans un polyarbre. Ce qui veut dire que dans l'expression de la croyance $BEL(x) = \alpha \lambda(x) \pi(x)$ d'un noeud X , seule la fonction $\pi(x)$ représentant la contribution des parents de X change. Quant à la contribution $\lambda(x)$ des fils de X , elle est calculée de la même façon que dans un réseau en arbre. Ainsi, pour un noeud X ayant m fils Y_1, \dots, Y_m , $\lambda(x) = \prod_{j=1}^m \lambda_{Y_j}(x)$. De même, la quantité $\pi_{Y_k}(x)$ envoyée par le noeud X à son k -ième fils, est toujours calculée comme dans l'Équation (3.15), c'est-à-dire comme suit :

$$\pi_{Y_k}(x) = \alpha \frac{BEL(x)}{\lambda_{Y_k}(x)} = \pi(x) \prod_{j=1, j \neq k}^m \lambda_{Y_j}(x) \quad (3.16)$$

Par contre l'expression de la croyance $BEL(x)$ n'est plus la même que dans le cas d'un réseau en arbre, puisque la façon de calculer la fonction $\pi(x)$ dans un polyarbre change. En effet, les observations collectées dans le sous-graphe G_X^+ ne se résument plus sur l'unique parent de X comme c'est le cas dans un arbre, mais sur tous les parents de X dans le polyarbre. Il faut donc pouvoir évaluer la contribution $\pi(x)$ de la combinaison $U = U_1, \dots, U_n$ des parents de X à la croyance en l'évènement $X = x$.

La Figure 3.5 présente un fragment d'un réseau bayésien en polyarbre encore appelé réseau simplement connecté. Le sous-graphe G_X^+ peut se diviser en n sous-graphes $G_{U_i X}^+, \dots, G_{U_n X}^+$ tel que $G_{U_i X}^+$, pour $i \in \{1, \dots, n\}$, est le sous-graphe obtenu en déconnectant le lien $U_i \rightarrow X$ et contenant le noeud U_i . Les évidences e_X^+ peuvent également se subdiviser en n partitions $e_{U_1 X}^+, \dots, e_{U_n X}^+$ telles que les évidences $e_{U_i X}^+ \subset e_X^+$, collectées dans le sous-graphe $G_{U_i X}^+$, se résument au niveau du parent U_i du noeud X . Notons $u = (u_1, \dots, u_n)$ une combinaison d'états des parents de X . En appliquant la règle d'expansion (3.6) l'expression de la contribution des parents peut se développer comme dans l'Équation (3.17). Dans l'Équation (3.17), $\pi_X(u_i) = \mathbb{P}(u_i | e_{U_i X}^+)$ est la contribution du parent U_i du noeud X à la croyance en l'évènement $X = x$.

$$\begin{aligned}
\pi(x) &= \mathbb{P}(x | e_X^+) = \sum_u \mathbb{P}(x | e_X^+, u) \mathbb{P}(u | e_X^+) \\
&= \sum_u \mathbb{P}(x | u) \mathbb{P}(u | e_X^+) \\
&= \sum_{u_1, \dots, u_n} \mathbb{P}(x | u_1, \dots, u_n) \mathbb{P}(u_1, \dots, u_n | e_{U_1 X}^+, \dots, e_{U_n X}^+) \\
&= \sum_{u_1, \dots, u_n} \mathbb{P}(x | u_1, \dots, u_n) \prod_{i=1}^n \mathbb{P}(u_i | e_{U_i X}^+) \tag{3.17} \\
&= \sum_u \mathbb{P}(x | u) \prod_{i=1}^n \mathbb{P}(u_i | e_{U_i X}^+) \\
&= \sum_u \mathbb{P}(x | u) \prod_{i=1}^n \pi_X(u_i)
\end{aligned}$$

L'expression finale de la croyance en l'évènement $X = x$ pour un réseau bayésien en polyarbre s'écrit comme suit :

$$BEL(x) = \alpha \left[\prod_{j=1}^m \lambda_{Y_j}(x) \right] \left[\sum_u \mathbb{P}(x | u) \prod_{i=1}^n \pi_X(u_i) \right] \tag{3.18}$$

ou la quantité $\lambda_{Y_j}(x)$ envoyée par le noeud Y_j à son parent X est calculée de la même façon que la quantité $\lambda_X(u_i) = \mathbb{P}(e_{U_i X}^- | u_i)$ envoyée par le noeud X à son parent U_i (voir les Équations (3.19) et (3.20)). L'expression du message $\lambda_X(u_i)$ envoyé par le noeud X à un parent U_i a été développée dans PEARL [1988]. Subdivisons à présent le sous-graphe G_X^+ en deux sous-graphes $G_{U_i X}^+$ et $G_{V X}^+$ tel que $V = U_1, \dots, U_{i-1}, U_{i+1}, \dots, U_n$. Les évidences e_X^+ se subdivisent également en deux parties $e_{U_i X}^+$ et $e_{V X}^+$ représentant respectivement les observations collectées dans les sous-graphes $G_{U_i X}^+$ et $G_{V X}^+$. La Figure 3.6 représente graphiquement cette subdivision.

Notons $e_{U_i X}^-$ la différence entre la totalité des observations e dans le graphe entier et les observations $e_{U_i X}^+$. Dans l'Équation (3.19), la notation $e \setminus e_{U_i X}^+$ désigne le complémentaire de l'ensemble des évidences $e_{U_i X}^+$ dans l'ensemble e de toutes les observations.

$$e_{U_i X}^- = e \setminus e_{U_i X}^+ = e_X^- \cup e_{V X}^+ \tag{3.19}$$

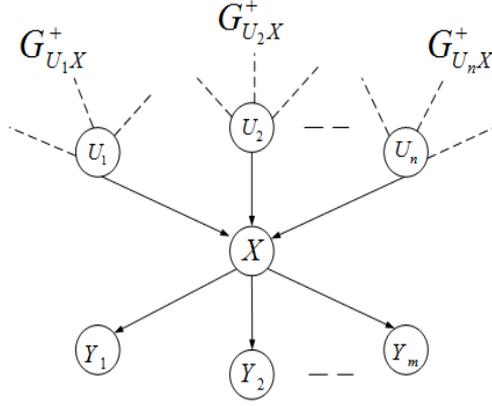


FIGURE 3.5 – Fragment d'un réseau bayésien avec une topologie en polyarbre

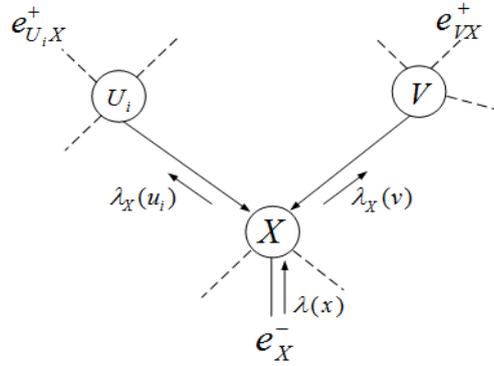


FIGURE 3.6 – Message $\lambda_X(u_i)$ envoyé par le noeud X à un parent U_i

$$\begin{aligned}
\lambda_X(u_i) &= \mathbb{P}(e_{U_iX}^- | u_i) = \mathbb{P}(e_X^-, e_{VX}^+ | u_i) \\
&= \sum_x \sum_v \mathbb{P}(e_X^-, e_{VX}^+ | u_i, v, x) \mathbb{P}(x, v | u_i) \\
&= \sum_x \sum_v \mathbb{P}(e_X^-, e_{VX}^+ | v, x) \mathbb{P}(x, v | u_i) \\
&= \sum_x \sum_v \mathbb{P}(e_X^- | x) \mathbb{P}(e_{VX}^+ | v) \mathbb{P}(x, v | u_i) \\
&= \beta \sum_x \sum_v \mathbb{P}(e_X^- | x) \frac{\mathbb{P}(v | e_{VX}^+)}{\mathbb{P}(v)} \mathbb{P}(x | v, u_i) \mathbb{P}(v | u_i) \\
&= \beta \sum_x \sum_v \mathbb{P}(e_X^- | x) \mathbb{P}(v | e_{VX}^+) \mathbb{P}(x | u) \\
&= \beta \sum_x \sum_{k \neq i} \lambda(x) \prod_{k \neq i} \mathbb{P}(u_k | e_{U_kX}^+) \mathbb{P}(x | u) \\
&= \beta \sum_x \lambda(x) \sum_{k \neq i} \mathbb{P}(x | u) \prod_{k \neq i} \pi_X(u_k)
\end{aligned} \tag{3.20}$$

Dans l'Équation (3.20) nous noterons que la constante $\beta = \mathbb{P}(e_{VX}^+)$ est la vraisemblance des évidences e_{VX}^+ . Puisque les parents du noeud X sont marginalement indépendants les uns des autres, $\mathbb{P}(v | u_i) = \mathbb{P}(v)$. La Figure 3.7 montre comment chaque noeud X dans un réseau bayésien en polyarbre peut calculer localement sa croyance $BEL(x) = \mathbb{P}(x | x^o)$ connaissant les évidences x^o collectées dans le réseau. Il faut tout simplement que le noeud X reçoive les messages $\lambda_{Y_j}(x)$ de ses fils et les messages $\pi_X(u_i)$ de ses parents PEARL [1988]. Dans

la Figure 3.7, M désigne la probabilité conditionnelle $\mathbb{P}(X|U_1, \dots, U_n)$ donnée par les paramètres du réseau.

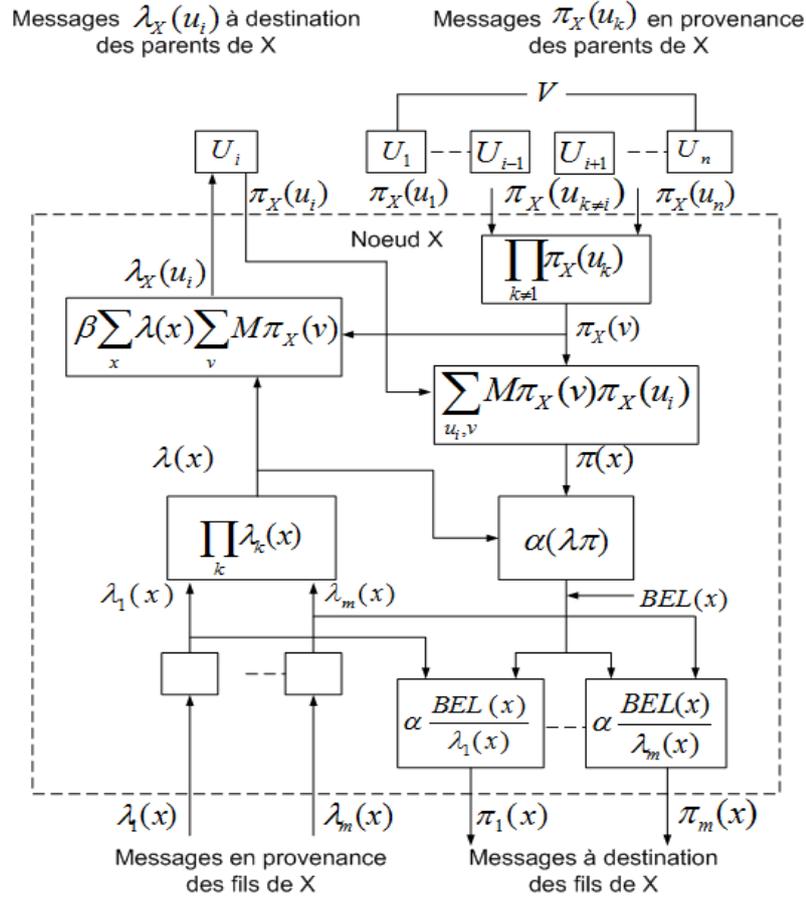


FIGURE 3.7 – Calcul de la croyance d'un noeud dans un réseau bayésien en polyarbre.

Revenons maintenant à notre exemple de réseau bayésien de la Figure 3.1 et illustrons l'algorithme somme-produit sur ce petit réseau en polyarbre. Calculons la distribution de probabilités a postériori aux évidences $x^o = (+j, +m)$ de chaque variable non observée. Mais avant, intéressons nous tout d'abord aux distributions de probabilités des variables à l'état d'équilibre du réseau, c'est-à-dire avant la survenance des événements $J = +j$ et $M = +m$. Dans cet état, les messages λ envoyés par les fils vers les parents sont des vecteurs unitaires, par exemple $\lambda_j(A) = \lambda_M(A) = [1, 1]$. Ce qui veut dire d'après l'Équation (3.10) que les distributions sont égales aux quantités π . À l'équilibre initial du réseau, les quantités π sont calculées comme suit :

$$\begin{aligned} \pi(B) = \pi_A(B) = \mathbb{P}(B) &= \begin{bmatrix} 0.999 \\ 0.001 \end{bmatrix}_{(b)} & \pi(E) = \pi_A(E) = \mathbb{P}(E) &= \begin{bmatrix} 0.998 \\ 0.002 \end{bmatrix}_{(e)} \\ \pi(A) = \pi_J(A) = \pi_M(A) &= \sum_{b,e} \pi_A(B) \pi_A(E) \mathbb{P}(A|B, E) \\ &= \sum_{b,e} \begin{bmatrix} 9.47e-01 & 4.98e-02 \\ 1.19e-04 & 1.87e-03 \\ 2.89e-04 & 7.08e-04 \\ 2.00e-09 & 1.99e-06 \end{bmatrix}_{(b,e,a)} \\ &= \begin{bmatrix} 0.947 \\ 0.052 \end{bmatrix}_{(a)} \end{aligned}$$

Nous rappelons que $\mathbb{P}(B)$, $\mathbb{P}(E)$ et $\mathbb{P}(A|B, E)$ sont données par la Table 3.1. Lorsque les évènements $J = +j$ et $M = +m$ surviennent, chacun des noeuds J et M délivre un message λ à son parent A .

$$\lambda_J(A) = \mathbb{P}(+j|A) = \begin{bmatrix} 0.100 \\ 0.950 \end{bmatrix}_{(a)} \quad \lambda_M(A) = \mathbb{P}(+m|A) = \begin{bmatrix} 0.300 \\ 0.990 \end{bmatrix}_{(a)}$$

Le Noeud A reçoit les contributions $\lambda_J(A)$ et $\lambda_M(A)$ respectivement de ses fils J et M à la croyance aux évènements $x^o = (+j, +m)$. Le noeud A peut donc évaluer localement sa distribution a posteriori $\mathbb{P}(A|x^o)$. Nous rappelons que α est une constante de normalisation.

$$\begin{aligned} \lambda(A) &= \mathbb{P}(+j, +m|A) = \mathbb{P}(+j|A)\mathbb{P}(+m|A) \\ &= \lambda_J(A)\lambda_M(A) \\ &= \begin{bmatrix} 0.030 \\ 0.940 \end{bmatrix}_{(a)} \end{aligned}$$

$$\begin{aligned} \mathbb{P}(A|x^o) &= \alpha\lambda(A)\pi(A) = \alpha \begin{bmatrix} 0.030 \\ 0.940 \end{bmatrix}_{(a)} \begin{bmatrix} 0.947 \\ 0.052 \end{bmatrix}_{(a)} \\ &= \begin{bmatrix} 0.365 & 0.635 \end{bmatrix}_{(a)} \end{aligned}$$

Le noeud A doit poursuivre la propagation des évidences en délivrant à son tour les messages $\lambda_A(B)$ et $\lambda_A(E)$ respectivement à chacun de ses parents B et E .

Calcul du message $\lambda_A(B)$:

$$\begin{aligned} \sum_e \mathbb{P}(A|B, E)\pi_A(E) &= \sum_e \begin{bmatrix} 9.49e-01 & 4.99e-02 \\ 1.20e-04 & 1.88e-03 \\ 2.89e-01 & 7.08e-01 \\ 2.00e-06 & 1.99e-03 \end{bmatrix}_{(b,e,a)} \\ &= \begin{bmatrix} 0.948 & 0.051 \\ 0.289 & 0.710 \end{bmatrix}_{(b,a)} \end{aligned}$$

$$\begin{aligned} \lambda(B) = \lambda_A(B) &= \sum_a \lambda(A) \sum_e \mathbb{P}(A|B, E)\pi_A(E) = \sum_a \begin{bmatrix} 0.028 & 0.008 \\ 0.048 & 0.668 \end{bmatrix}_{(a,b)} \\ &= \begin{bmatrix} 0.077 \\ 0.676 \end{bmatrix}_{(b)} \end{aligned}$$

Calcul du message $\lambda_A(E)$:

$$\begin{aligned} \sum_b \mathbb{P}(A|B, E)\pi_A(B) &= \sum_b \begin{bmatrix} 9.49e-01 & 4.99e-02 \\ 5.99e-02 & 9.39e-01 \\ 2.90e-04 & 7.10e-04 \\ 1.00e-06 & 9.99e-04 \end{bmatrix}_{(b,e,a)} \\ &= \begin{bmatrix} 0.949 & 0.050 \\ 0.059 & 0.940 \end{bmatrix}_{(e,a)} \end{aligned}$$

$$\begin{aligned}\lambda(E) = \lambda_A(E) &= \sum_a \lambda(A) \sum_b \mathbb{P}(A|B, E) \pi_A(B) = \sum_a \begin{bmatrix} 0.028 & 0.001 \\ 0.047 & 0.884 \end{bmatrix}_{(a,e)} \\ &= \begin{bmatrix} 0.076 \\ 0.885 \end{bmatrix}_{(e)}\end{aligned}$$

Le noeud B reçoit le message $\lambda_A(B)$ en provenance de son fils A. Il peut donc évaluer localement sa distribution a postérieure $\mathbb{P}(B|x^o)$.

$$\begin{aligned}\mathbb{P}(B|x^o) &= \alpha \lambda(B) \pi(B) = \alpha \begin{bmatrix} 0.077 \\ 0.676 \end{bmatrix}_{(b)} \begin{bmatrix} 0.999 \\ 0.001 \end{bmatrix}_{(b)} \\ &= \begin{bmatrix} 0.991 & 0.009 \end{bmatrix}_{(b)}\end{aligned}$$

Le noeud E reçoit à son tour le message $\lambda_A(E)$ en provenance de son fils A. Il peut donc aussi évaluer localement sa distribution a postérieure $\mathbb{P}(E|x^o)$.

$$\begin{aligned}\mathbb{P}(E|x^o) &= \alpha \lambda(E) \pi(E) = \alpha \begin{bmatrix} 0.076 \\ 0.885 \end{bmatrix}_{(e)} \begin{bmatrix} 0.998 \\ 0.002 \end{bmatrix}_{(e)} \\ &= \begin{bmatrix} 0.977 & 0.023 \end{bmatrix}_{(e)}\end{aligned}$$

3.4.3 Topologie avec des boucles acycliques

Un réseau bayésien avec une topologie contenant des boucles acycliques est aussi appelé réseau bayésien multi-connecté. Il s'agit de la classe de réseaux bayésiens les plus complexes. Les boucles sont dues au fait qu'un noeud ou plusieurs noeuds du réseau peuvent à la fois faire partir des descendants et des ascendants d'un autre noeud. L'algorithme somme-produit étant naturellement un algorithme de passage de messages entre tous les noeuds du réseau, sa convergence n'est plus garantie sur ce type de topologie. En effet, l'hypothèse d'absence de boucles dans le réseau qui a permis de factoriser la croyance $BEL(x) = \alpha \lambda(x) \pi(x)$ n'est plus juste sur un réseau multi-connecté. L'algorithme somme-produit ne peut donc pas être appliqué directement sur ce type de topologie.

Une méthode de transformation d'un réseau bayésien multi-connecté en un arbre non orienté à variables composées a été proposée et développée dans LAURITZEN et SPIEGELHALTER [1988] MADSEN et JENSEN [1999] JENSEN et collab. [1990] LAURITZEN [1996] JENSEN et NIELSEN [2007]. Il s'agit d'un arbre de jonction qui est une représentation simplifiée et équivalente à un réseau bayésien quel que soit sa topologie. Judéa Pearl a ensuite proposé dans son livre PEARL [1988], une adaptation de l'algorithme somme-produit sur un arbre de jonction orienté conformément à l'orientation du réseau bayésien initial. Mais cette réorientation de l'arbre de jonction n'est pas utile puisque l'arbre de jonction même non orienté est déjà équivalent au réseau bayésien initial. Dans la section qui suit nous allons étudier l'algorithme d'inférence sur un arbre de jonction non orienté LAURITZEN et SPIEGELHALTER [1988] MADSEN et JENSEN [1999] JENSEN et collab. [1990] LAURITZEN [1996] JENSEN et NIELSEN [2007].

3.5 Algorithme d'inférence sur un arbre de jonction

Le principe utilisé pour construire un arbre de jonction à partir d'un réseau bayésien est de fusionner des noeuds du réseau bayésien en appliquant successivement trois opérations : la moralisation du graphe de dépendances, la triangulation du graphe moral et le calcul du "maximal weighted spanning tree" du graphe triangulé.

3.5.1 Arbre de jonction et potentiel

La moralisation d'un graphe orienté se fait en deux étapes. On commence par désorienter tous les arcs du graphe. Ensuite on ajoute une arête entre chaque couple de parents d'un même noeud, si l'arête en question n'existe pas déjà. Autrement dit on "marie" les parents d'un même noeud [NAIM et collab. \[2008\]](#). De façon formelle, le graphe moral $G^m = (V, E^m)$ du graphe orienté $\vec{G} = (V, E)$ est obtenu par l'Équation (3.21). Les notations V , E et E^m désignent respectivement l'ensemble des noeuds de chacun des deux graphes, l'ensemble des arcs du graphe \vec{G} et l'ensemble des arêtes du graphe G^m .

$$(u, v) \in E^m \iff [(u \rightarrow v) \in E] \vee [(v \rightarrow u) \in E] \vee [\exists w \in V \mid u, v \subset \text{parent}(w)] \quad (3.21)$$

Le graphe moral a donc la propriété de créer une clique pour chaque noeud et ses parents [NAIM et collab. \[2008\]](#). Une clique est un ensemble de noeuds du graphe moral adjacents deux à deux.

La triangulation d'un graphe moral consiste à ajouter, dans chaque cycle de longueur supérieure à trois, une corde ou une arête entre deux noeuds non adjacents. Autrement dit, le graphe moral est triangulé si et seulement si tout cycle de longueur supérieure à trois possède une corde. Tous les cycles minimaux d'un graphe triangulé sont donc de longueur trois. Notons $G^c = (V, E^c)$ le graphe triangulé du graphe moralisé $G^m = (V, E^m)$. Notons ζ l'ensemble des cycles de G^m , $ve(C)$ l'ensemble des sommets d'un cycle $C \in \zeta$, $|ve(C)|$ le nombre de sommets de C et $ed(C)$ le nombre d'arêtes de C . Le graphe triangulé G_c est défini par l'Équation (3.22).

$$(u, v) \in E^c \iff [(u, v) \in E^m] \vee \exists C \in \zeta \text{ with } |ve(C)| > 3 \text{ tel que } [u, v \in ve(C) \wedge (u, v) \notin ed(C)] \quad (3.22)$$

Il faut noter que la qualité de la triangulation déterminera largement l'efficacité de l'algorithme d'inférence sur l'arbre de jonction qui est en train d'être construit. La triangulation est un problème d'optimisation NP-difficile. En effet, une triangulation optimale est par exemple celle qui ajoute le moins de cordes ou d'arêtes dans le graphe moralisé, ainsi appelée triangulation minimale ou "minimum fill-in triangulation". Dans [BERRY et collab. \[2003\]](#), un algorithme de triangulation minimale est proposé.

La troisième et dernière étape pour construire un arbre de jonction est de calculer le "maximal weighted spanning tree" du graphe triangulé G^c . Le "maximal spanning tree" du graphe G^c est un sous-graphe avec une topologie en arbre qui connecte toutes les cliques du graphe G^c , en maximisant la somme totale des poids des arêtes. Le poids désigne la taille de l'intersection entre les cliques adjacentes, c'est-à-dire le nombre de variables non composées que se partagent les cliques adjacentes. Le "maximal weighted

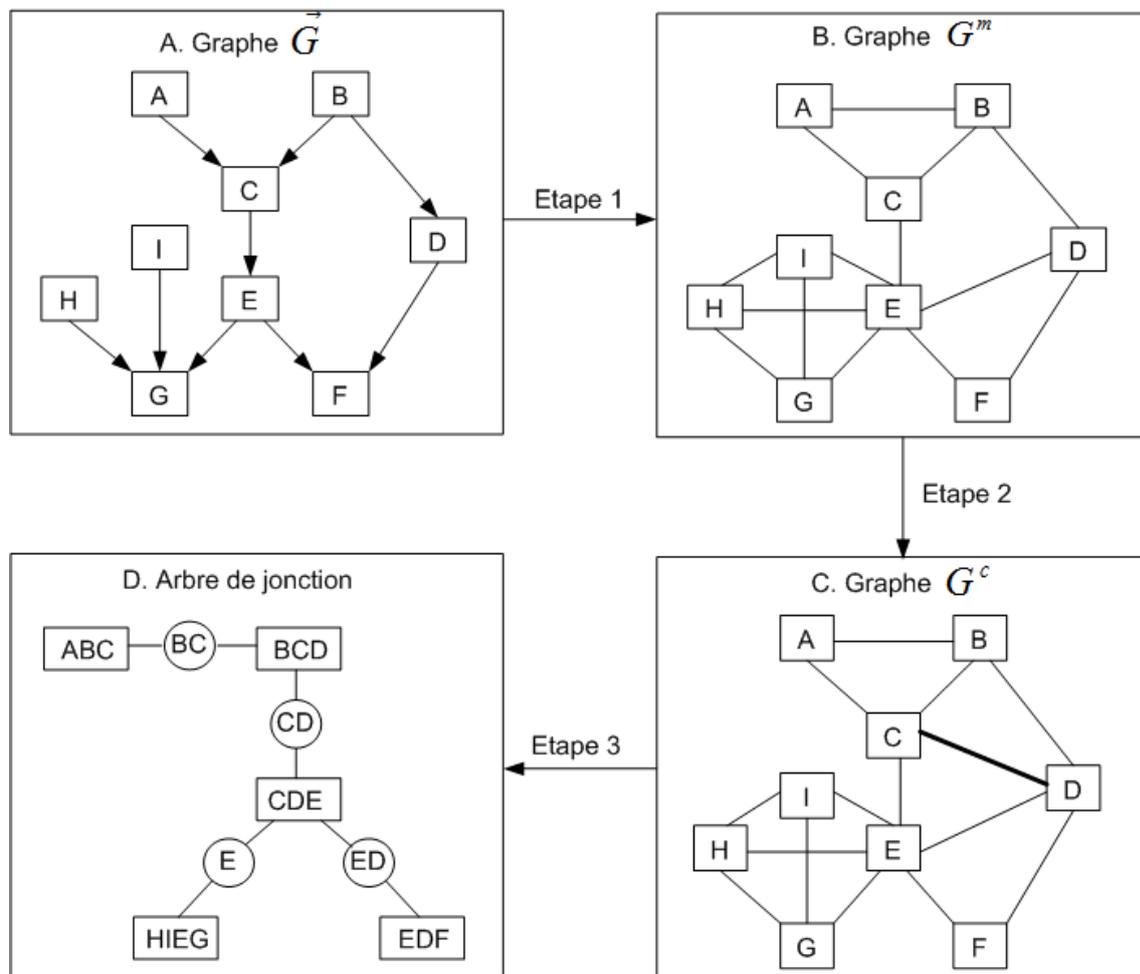


FIGURE 3.8 – Illustration des étapes de construction d'un arbre de jonction.

spanning tree" d'un graphe non orienté peut se calculer avec l'algorithme de Kruskal. Un arbre de jonction vérifie la propriété d'intersection courante très utile pour la propagation d'évidences dans cette structure. La propriété d'intersection courante nous dit que pour deux cliques quelconques C_1 et C_2 de l'arbre, tous les cliques et séparateurs qui forment le chemin entre C_1 et C_2 dans l'arbre contiennent $C_1 \cap C_2$.

La Figure 3.8 illustre au moyen d'un exemple simple, les trois étapes de construction d'un arbre de jonction. Les Figures 3.8.A, 3.8.B, 3.8.C et 3.8.D présentent respectivement un exemple de graphe orienté \vec{G} , son graphe moralisé G^m , son graphe triangulé G^c et l'arbre de jonction équivalent au graphe orienté \vec{G} . L'arbre de jonction 3.8.D contient cinq cliques $\{ABC\}$, $\{BCD\}$, $\{CDE\}$, $\{EDF\}$, $\{HIEG\}$ et quatre séparateurs $\{BC\}$, $\{CD\}$, $\{ED\}$, $\{E\}$. Il faut noter pour cet exemple que dans l'étape 1, il existe un seul cycle du graphe moralisé de longueur supérieure à 3 dont les sommets ne sont adjacents deux à deux, il s'agit du cycle (B, C, E, D). La triangulation à l'étape 2 consiste à transformer ce cycle en une ou plusieurs cliques, ce qui peut être fait en ajoutant une corde entre les sommets C, D et une seconde corde entre les sommets B, E. Mais il ne s'agirait pas dans ce cas d'une triangulation minimale puisque l'ajout d'une seule corde entre les sommets C, D ou entre les sommets B, E suffit pour trianguler le graphe G^m .

La Figure 3.9.a représente l'arbre de jonction du réseau bayésien R de la Figure 3.1. On

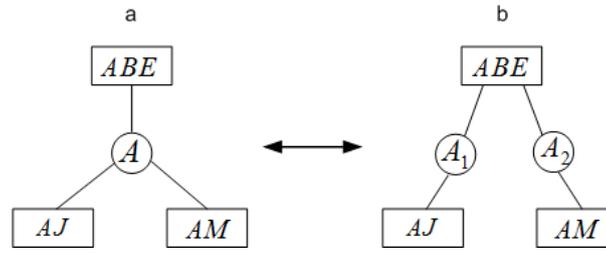


FIGURE 3.9 – Arbre de jonction du réseau bayésien de la Figure 3.1

note trois cliques et le séparateur commun A qui est l'intersection entre les trois cliques. Dans la Figure 3.9.b qui représente aussi l'arbre de jonction du réseau bayésien R, le séparateur A a été dupliqué. Les deux représentations sont donc équivalentes. Mais la représentation 3.9.b permet de mieux appréhender l'algorithme de propagation des évidences sur cette structure.

3.5.2 Propagation d'évidences

L'algorithme d'inférence ou de propagation d'évidences sur un arbre de jonction repose sur la notion de potentiels de cliques et de séparateurs. En effet, l'arbre de jonction est initialisé en associant une fonction appelée potentiel à chaque clique et séparateur de l'arbre de jonction. Le potentiel d'une clique est une fonction des variables de cette clique. La valeur initiale du potentiel d'une clique C_p adjacente à une autre clique C_q ainsi que celle du séparateur S_{pq} entre les deux cliques sont calculées comme suit. La notation $pa(X_i)$ désigne l'ensemble des parents de la variable X_i dans le réseau bayésien initial.

$$\begin{aligned} \phi_{C_p} &= \prod_{X_i \in C_p, pa(X_i) \subset C_p \vee pa(X_i) = \emptyset} \mathbb{P}(X_i | pa(X_i)) \\ \phi_{S_{pq}} &= 1 \end{aligned} \quad (3.23)$$

Nous rappelons que le produit des distributions $\mathbb{P}(X_i | pa(X_i))$ de l'Équation 3.23 n'est pas un produit matriciel, mais plutôt un produit tensoriel de fonctions. En effet, une distribution $\mathbb{P}(X_i | pa(X_i))$ est une fonction des variables $(X_i, pa(X_i))$. Le potentiel initial d'une clique se calcule donc comme un produit de fonctions tel que formalisé par l'Équation (3.9).

Les valeurs initiales des potentiels des cliques dans l'arbre de jonction de la Figure 3.9 se calculent comme suit : $\phi_{ABE} = \mathbb{P}(A|B, E)\mathbb{P}(B)\mathbb{P}(E)$, $\phi_{AJ} = \mathbb{P}(J|A)$, $\phi_{AM} = \mathbb{P}(M|A)$. La valeur initiale du potentiel du séparateur est $\phi_{A_1} = \phi_{A_2} = 1$.

$$\begin{aligned} \phi_{AJ} = \mathbb{P}(J|A) &= \begin{bmatrix} 0.90 & 0.10 \\ 0.05 & 0.95 \end{bmatrix}_{(a,j)} & \phi_{AM} = \mathbb{P}(M|A) &= \begin{bmatrix} 0.70 & 0.30 \\ 0.01 & 0.99 \end{bmatrix}_{(a,m)} \\ \phi_{ABE} = \mathbb{P}(A|B, E)\mathbb{P}(B)\mathbb{P}(E) &= \begin{bmatrix} 9.47e-01 & 4.98e-02 \\ 1.19e-04 & 1.87e-03 \\ 2.89e-04 & 7.08e-04 \\ 2.00e-09 & 1.99e-06 \end{bmatrix}_{(b,e,a)} \end{aligned}$$

L'algorithme de propagation des évidences x^o sur l'arbre de jonction consiste essentiellement à actualiser les potentiels des cliques C_p et C_q au travers de leur séparateur commun S_{pq} . Supposons par exemple que le potentiel de la clique C_p a déjà été actualisé

et celui de la clique C_q n'a pas encore été actualisé, c'est-à-dire que la propagation des évidences dans l'arbre de jonction a déjà atteint la clique C_p mais n'a pas encore atteint la clique C_q adjacente. L'algorithme de propagation procède en deux étapes. Le potentiel du séparateur S_{pq} est actualisé en appliquant l'opération de marginalisation de l'Équation (3.24). Le potentiel de la clique C_q est ensuite actualisé à son tour en appliquant le produit de l'Équation (3.25). La notation $C_p \setminus S_{pq}$ représente l'ensemble des variables de la clique C_p qui n'appartiennent pas au séparateur S_{pq} .

$$\Phi_{S_{pq}}^* = \sum_{C_p \setminus S_{pq}} \Phi_{C_p}^* \quad (3.24)$$

$$\Phi_{C_q}^* = \Phi_{C_q} \frac{\Phi_{S_{pq}}^*}{\Phi_{S_{pq}}} \quad (3.25)$$

On dit que la clique C_q absorbe les évidences en provenance de la clique C_p MADSEN et JENSEN [1999] ou que la clique C_p apporte les évidences à la clique C_q . Notons que l'algorithme de propagation commence par recalculer le potentiel de chaque clique observée. Une clique est dite observée lorsqu'au moins une variable de cette clique est observée. Pour revenir à notre exemple, les cliques AJ et AM sont observées puisque les variables J et M sont observées avec les valeurs respectives $+j$ et $+m$. Les potentiels des cliques observées AJ et AM sont recalculés et ne sont plus que des fonctions de la variable aléatoire A.

$$\Phi_{AJ}^o = \begin{bmatrix} 0.10 \\ 0.95 \end{bmatrix}_{(a)} \quad \Phi_{AM}^o = \begin{bmatrix} 0.30 \\ 0.99 \end{bmatrix}_{(a)}$$

Après ce calcul, les opérations de mise à jour des potentiels des cliques et des séparateurs sont faites en deux phases récursives. La première phase appelée opération de collecte est initiée par la collecte d'évidences en partant des cliques feuilles de l'arbre de jonction vers les cliques à la racine de l'arbre. Ceci suppose qu'on se donne arbitrairement un sens d'orientation sur l'arbre en désignant certaines cliques pendantes de l'arbre comme les cliques à la racine de l'arbre. Les autres cliques pendantes sont donc considérées comme les cliques feuilles de l'arbre. Les cliques à la racine et les cliques feuilles désignent respectivement des cliques n'ayant pas de prédécesseurs et des cliques n'ayant pas de successeurs dans l'arbre de jonction selon le sens d'orientation qu'on s'est donné. La collecte des évidences vers une clique est faite en collectant les évidences vers tous les successeurs de cette clique suivie par l'absorption des évidences partant de chaque successeur.

Concernant notre exemple, nous choisissons la clique ABE comme étant la racine de l'arbre. Les cliques AJ et AM sont donc les cliques feuilles de l'arbre. La phase de collecte s'exécute selon l'ordre des équations suivantes. Le potentiel du séparateur A_1 est actualisé ainsi que celui de la clique ABE. Le potentiel actualisé de la clique ABE est noté Φ_{ABE}^{*1} durant l'absorption des évidences partant de la clique AJ vers la clique ABE au travers du séparateur commun A_1 .

$$\Phi_{A_1}^* = \sum_j \Phi_{AJ}^o = \Phi_{AJ}^o = \begin{bmatrix} 0.10 \\ 0.95 \end{bmatrix}_{(a)}$$

$$\begin{aligned}\Phi_{ABE}^{*1} &= \Phi_{ABE} \frac{\Phi_{A_1}^*}{\Phi_{A_1}} = \begin{bmatrix} 9.47e-01 & 4.98e-02 \\ 1.19e-04 & 1.87e-03 \\ 2.89e-04 & 7.08e-04 \\ 2.00e-09 & 1.99e-06 \end{bmatrix}_{(b,e,a)} \begin{bmatrix} 0.10 \\ 0.95 \end{bmatrix}_{(a)} \\ &= \begin{bmatrix} 9.47e-02 & 1.19e-05 \\ 2.89e-05 & 2.00e-10 \\ 4.73e-02 & 1.78e-03 \\ 6.73e-04 & 1.89e-06 \end{bmatrix}_{(b,e,a)}\end{aligned}$$

Le potentiel du séparateur A_2 est actualisé à son tour. Celui de la clique ABE est actualisé une seconde fois. Le potentiel actualisé de la clique ABE est noté Φ_{ABE}^{*2} durant l'absorption des évidences partant de la clique AM vers la clique ABE au travers du séparateur commun A_2 .

$$\Phi_{A_2}^* = \sum_m \Phi_{AM}^o = \Phi_{AM}^o = \begin{bmatrix} 0.30 \\ 0.99 \end{bmatrix}_{(a)}$$

$$\begin{aligned}\Phi_{ABE}^{*2} &= \Phi_{ABE}^{*1} \frac{\Phi_{A_2}^*}{\Phi_{A_2}} = \begin{bmatrix} 9.47e-02 & 1.19e-05 \\ 2.89e-05 & 2.00e-10 \\ 4.73e-02 & 1.78e-03 \\ 6.73e-04 & 1.89e-06 \end{bmatrix}_{(b,e,a)} \begin{bmatrix} 0.30 \\ 0.99 \end{bmatrix}_{(a)} \\ &= \begin{bmatrix} 2.84e-02 & 4.68e-02 \\ 3.59e-06 & 1.76e-03 \\ 8.68e-06 & 6.66e-04 \\ 6.00e-11 & 1.87e-06 \end{bmatrix}_{(b,e,a)}\end{aligned}$$

Notons que le choix d'actualiser le potentiel de la clique ABE au travers du séparateur A_1 une première fois puis au travers du séparateur A_2 une seconde fois est arbitraire. Nous aurions pu inverser cet ordre et le résultat final Φ_{ABE}^{*2} du potentiel actualisé de la clique ABE aurait été le même.

La seconde phase appelée opération de distribution consiste à distribuer les évidences en partant des cliques à la racine de l'arbre de jonction vers les cliques feuilles de l'arbre. La distribution des évidences partant d'une clique consiste à apporter les évidences à chaque successeur de cette clique suivi par la distribution des évidences partant de chaque successeur. Illustrons également cette seconde phase sur notre exemple. La clique ABE distribue les évidences aux cliques AJ et AM respectivement au travers du séparateur commun A. Le potentiel du séparateur A est donc actualisé.

$$\Phi_A^* = \sum_{b,e} \Phi_{ABE}^{*2} = \begin{bmatrix} 0.028 \\ 0.049 \end{bmatrix}_{(a)}$$

Le potentiel de chacune des cliques AJ et AM est ensuite actualisé. Notons que ces potentiels Φ_{AJ}^* et Φ_{AM}^* ne dépendent pas respectivement des variables J et M puisqu'elles sont observées.

$$\Phi_{AJ}^* = \Phi_{AJ}^o \frac{\Phi_A^*}{\Phi_{A_1}^*} = \Phi_A^*$$

$$\Phi_{AM}^* = \Phi_{AM}^o \frac{\Phi_A^*}{\Phi_{A_2}^*} = \Phi_A^*$$

Notons que la propriété d'intersection courante vérifiée par un arbre de jonction garantit la cohérence des potentiels actualisés lorsque les deux phases de l'algorithme de propagation sont terminées. La cohérence des potentiels actualisés garantit que la probabilité marginale d'une variable aléatoire X soit identique dans chaque clique contenant X . Le potentiel actualisé $\Phi_{C_p}^*$ d'une clique quelconque C_p contenant X devient la loi jointe des variables de cette clique et des évidences x^o .

$$\Phi_{C_p}^* = \mathbb{P}(C_p, x^o) \quad (3.26)$$

Ce qui signifie que la vraisemblance $\mathbb{P}(x^o)$ des évidences x^o , ainsi que la probabilité marginale $\mathbb{P}(X|x^o)$ de la variable $X \in C_p$ peuvent se calculer en appliquant les deux opérations de marginalisations suivantes.

$$\mathbb{P}(x^o) = \sum_{C_p} \Phi_{C_p}^* \quad \mathbb{P}(X|x^o) = \sum_{C_p \setminus X} \mathbb{P}(C_p|x^o) \quad (3.27)$$

Notons que $\mathbb{P}(C_p|x^o) = \Phi_{C_p}^* / \mathbb{P}(x^o)$ désigne la distribution de probabilités jointe a posteriori des variables de la clique C_p connaissant les évidences x^o . Pour notre exemple la vraisemblance des évidences $x^o = (+j, +m)$ se calcule donc en sommant tous les éléments du potentiel actualisé de n'importe quelle clique de l'arbre de jonction.

$$\mathbb{P}(x^o) = \sum_{abe} \Phi_{ABE}^{*2} = \sum_{aj} \Phi_{AJ}^* = \sum_{AM} \Phi_{AM}^* = 0.077$$

La distribution de probabilités a posteriori aux évidences de chacune des variables B et E se calcule en normalisant le résultat de deux opérations de marginalisation sur le potentiel actualisé Φ_{ABE}^{*2} de la clique ABE .

$$\mathbb{P}(B|x^o) = \frac{1}{\mathbb{P}(x^o)} \sum_{ae} \Phi_{ABE}^{*2} = [0.991 \quad 0.009]_{(b)}$$

$$\mathbb{P}(E|x^o) = \frac{1}{\mathbb{P}(x^o)} \sum_{ab} \Phi_{ABE}^{*2} = [0.977 \quad 0.023]_{(e)}$$

La distribution de probabilités a posteriori aux évidences de la variable A se calcule à partir du potentiel actualisé de n'importe quelle clique de l'arbre de jonction. En effet, la variable A appartient aux trois cliques de l'arbre.

$$\begin{aligned} \mathbb{P}(A|x^o) &= \frac{1}{\mathbb{P}(x^o)} \sum_{be} \Phi_{ABE}^{*2} = [0.365 \quad 0.635]_{(a)} \\ &= \frac{1}{\mathbb{P}(x^o)} \sum_j \Phi_{AJ}^* = [0.365 \quad 0.635]_{(a)} \\ &= \frac{1}{\mathbb{P}(x^o)} \sum_m \Phi_{AM}^* = [0.365 \quad 0.635]_{(a)} \end{aligned}$$

3.6 Discussion et conclusion

Nous avons étudié dans ce chapitre, les algorithmes d'inférence exacte sur un réseau bayésien. La marginalisation, est la forme d'inférence la plus simple, mais reste calculatoirement impraticable sur des grandes topologies avec des variables dont la dimension de l'espace des états est grande. L'algorithme somme-produit est un algorithme de renforcement de croyances. Chaque noeud du réseau peut évaluer localement sa croyance en un évènement ou une observation sur le réseau s'il reçoit l'évaluation de la contribution de ses prédécesseurs et de ses successeurs à la croyance en cet évènement. L'algorithme somme-produit est donc naturellement un algorithme de passage de messages. Par conséquent, il ne peut converger lorsque la topologie du réseau contient des boucles acycliques entre certains noeuds du réseau. L'algorithme somme-produit n'est de ce fait adapté qu'aux réseaux en arbre et aux réseaux en polyarbre ou réseaux simplement connectés. Quant aux réseaux multi-connectés, le concept d'arbre de jonction est introduit. Il s'agit d'une représentation simplifiée et équivalente à un réseau bayésien. Chaque noeud de cet arbre est une clique de variables du réseau initial. L'algorithme d'inférence sur cette structure calcule la distribution de probabilités jointes a posteriori aux observations de chaque clique. La distribution de probabilités a posteriori aux observations d'une variable appartenant à une clique est donc obtenue par une simple opération de marginalisation. L'algorithme d'inférence sur un arbre de jonction est générique puisque son principe reste inchangé, quelle que soit la complexité de la topologie du réseau bayésien.

Chapitre 4

Apprentissage artificiel des paramètres d'un réseau bayésien

« Imagination is more important than knowledge. Knowledge is limited. Imagination encircles the world. »

Albert Einstein, 1929

Sommaire

4.1 Introduction	65
4.2 Estimation paramétrique au sens du Maximum de vraisemblance (MLE) 65	
4.2.1 Les bases de l'algorithme MLE	65
4.2.2 L'algorithme MLE sur un réseau bayésien	67
4.3 L'algorithme MLE avec des données incomplètes	68
4.3.1 Description des données incomplètes	68
4.3.2 L'algorithme d'Espérance-Maximisation (EM)	68
4.3.3 Stabilité numérique et convergence de l'algorithme EM	69
4.4 L'algorithme EM sur un réseau bayésien	71
4.4.1 Étape d'Espérance	71
4.4.2 Calculs de l'étape E basés sur la structure graphique du réseau	72
4.4.3 Étape de Maximisation	73
4.4.4 Pseudo-code de l'algorithme EM sur un réseau bayésien	74
4.4.5 Exemple illustratif de l'algorithme EM sur un réseau bayésien	74
4.5 Discussion et conclusion	78

4.1 Introduction

La modélisation du comportement d'un réseau est un problème central pour les méthodes à base de modèles de diagnostic automatique des réseaux de télécommunication. Il existe deux approches principales pour construire de tels modèles. La première approche est de construire le modèle du réseau à partir des connaissances dites "expertes" acquises en étudiant les standards ou les normes du réseau. La seconde approche consiste à appliquer des algorithmes de découverte et d'extraction de connaissances pour apprendre artificiellement le modèle du réseau à partir des données générées par les équipements. Le comportement d'un réseau encore appelé propagation de pannes est un ensemble de relations de causes à effets non déterministes qui peut être modélisé par un réseau bayésien. Les relations de causes à effets qui forment le graphe de dépendances d'un tel réseau bayésien peuvent être devinées par des experts du réseau de télécommunication à modéliser. Par contre, il est très difficile pour ces experts de déduire sur la seule base de leurs connaissances "expertes", les paramètres exacts ou les distributions de probabilités conditionnelles qui quantifient les incertitudes sur les dépendances. Nous étudions dans ce chapitre l'algorithme d'apprentissage artificiel des paramètres d'un réseau bayésien basé sur le concept du maximum de vraisemblance (MLE) des données générées par le réseau de télécommunication. Nous commençons par présenter dans la Section 2, les bases de l'algorithme MLE et son application sur un réseau bayésien lorsque les données sont complètes. Pour le cas des données incomplètes, nous étudions l'algorithme d'Espérance-maximisation (EM) dans la Section 3. Dans la Section 4, nous appliquons l'EM sur un réseau bayésien. Ce chapitre se termine par une discussion et une conclusion faites dans la Section 5.

4.2 Estimation paramétrique au sens du Maximum de vraisemblance (MLE)

En statistique, l'estimation paramétrique consiste à déduire ou inférer les valeurs inconnues des paramètres d'un modèle statistique en se basant sur un échantillon de données empiriques de ce modèle.

4.2.1 Les bases de l'algorithme MLE

Considérons la variable aléatoire X qui prend ses valeurs dans l'espace fini $\{v_1, \dots, v_k\}$ avec le vecteur de probabilités $\theta = \{p_1, \dots, p_k\}$ telles que $\mathbb{P}(X = v_i) = p_i$. Supposons que l'ensemble $D = \{x^1, x^2, \dots, x^T\}$ est une suite de T réalisations ou observations indépendantes et identiquement distribuées de X . Pour cet exemple simple, l'estimation des probabilités p_i à partir des données observées D est un problème d'estimation paramétrique.

L'algorithme MLE [PFANZAGL et collab. \[1994\]](#) [EDGEWORTH et FRANCIS \[1909\]](#) [SAVAGE \[1976\]](#) est une méthode particulière d'estimation des paramètres d'un modèle statistique lorsque l'on dispose d'un échantillon de données de ce modèle.

$$\hat{\theta} = \underset{\theta}{\text{Arg max}} L(D; \theta) \quad (4.1)$$

Le principe de cet algorithme est de sélectionner l'ensemble $\hat{\theta}$ des valeurs des paramètres qui maximisent le logarithme de la vraisemblance (log-vraisemblance) des don-

nées $L(D; \theta)$. Autrement dit, il s'agit de trouver les valeurs des paramètres du modèle statistique pour lesquelles les données observées conviennent mieux au modèle. La log-vraisemblance des données est calculée par l'Équation (4.2). La probabilité jointe des observations (vraisemblance) peut se factoriser en un produit puisque nous avons fait l'hypothèse que les observations sont indépendantes et identiquement distribuées.

$$\begin{aligned}
L(D; \theta) &= \log \mathbb{P}(x^1, x^2, \dots, x^T; \theta) \\
&= \log \prod_{t=1}^T \mathbb{P}(x^t; \theta) \\
&= \sum_{t=1}^T \log \mathbb{P}(x^t; \theta) \\
&= \sum_{t=1}^T \sum_{i=1}^k \log \mathbb{P}(x^t = v_i; \theta) \mathbb{I}_{x^t=v_i} \\
&= \sum_{i=1}^k \log \mathbb{P}(x^t = v_i; \theta) \sum_{t=1}^T \mathbb{I}_{x^t=v_i} \\
L(D; \theta) &= \sum_{i=1}^k N_i \log p_i
\end{aligned} \tag{4.2}$$

Nous notons $N_i = \sum_{t=1}^T \mathbb{I}_{x^t=v_i}$, le nombre de réalisations de l'évènement $X = v_i$ dans l'échantillon de données D . La notation \mathbb{I} représente la fonction indicatrice. Ainsi $\mathbb{I}_{x^t=v_i}$ est égale à 1 si $x^t = v_i$ ou égale à 0 sinon. En substituant l'Équation (4.2) dans l'Équation (4.1), on obtient l'Équation (4.3) qui formalise un problème de maximisation avec contrainte.

$$\text{Arg max}_{p_i} \sum_{i=1}^k N_i \log p_i \quad \text{sous la contrainte} \quad \sum_{i=1}^k p_i = 1 \tag{4.3}$$

Le problème de maximisation (4.3) peut être facilement résolu avec la méthode des multiplicateurs de Lagrange [LAGRANGE \[1888\]](#) [ZIEMER \[1989\]](#). On introduit donc la fonction $\Phi(p_i, \lambda)$ définie comme suit :

$$\Phi(p_i, \lambda) = \sum_{i=1}^k N_i \log p_i + \lambda \left(1 - \sum_{i=1}^k p_i\right) \tag{4.4}$$

La résolution du problème (4.3) se fait en maximisant la fonction Φ par rapport à p_i et par rapport au multiplicateur de Lagrange λ comme suit :

$$\begin{aligned}
\frac{\partial \Phi}{\partial p_i} &= N_i \times \frac{1}{p_i} - \lambda = 0 \Rightarrow p_i = \frac{N_i}{\lambda} \\
\frac{\partial \Phi}{\partial \lambda} &= 1 - \sum_{i=1}^k p_i = 0 \Rightarrow \sum_{i=1}^k p_i = 1
\end{aligned} \tag{4.5}$$

La résolution du système d'équations (4.5) donne la valeur du multiplicateur de Lagrange $\lambda = \sum_{i=1}^k N_i = T$. La valeur $\hat{\theta}$ du vecteur de paramètres θ qui maximise la fonction logarithmique de la vraisemblance des observations est donnée par l'Équation (4.6).

$$\hat{\theta} = \left[p_i = \frac{N_i}{\sum_{i=1}^k N_i} = \frac{N_i}{T}, i \in \{1, \dots, k\} \right] \tag{4.6}$$

L'Équation (4.6) montre que les proportions p_i sont obtenues par un simple calcul de fréquences empiriques, c'est-à-dire en divisant le nombre N_i d'occurrences de l'évènement $X = v_i$ par le nombre total T d'observations dans l'échantillon de données D .

4.2.2 L'algorithme MLE sur un réseau bayésien

Considérons à présent le problème de l'estimation paramétrique d'un modèle statistique représenté par un réseau bayésien. Un réseau bayésien [PEARL \[1985\]](#) [PEARL \[1986\]](#) [PEARL \[1988\]](#) [NAIM et collab. \[2008\]](#) [CORNÉJOLS et MICLET \[2013\]](#) est un modèle graphique probabiliste représenté par un graphe de dépendances orienté et acyclique. Les noeuds du graphe sont des variables aléatoires. La force de dépendance entre chaque noeud et ses prédécesseurs dans le graphe est quantifiée au moyen d'une distribution de probabilités conditionnelles. Considérons un réseau bayésien de N variables aléatoires $\{X_1, \dots, X_N\}$. Notons $\theta_{i,j,k}$ la probabilité conditionnelle que la variable X_i prenne la valeur k sachant que ses prédécesseurs sont dans la configuration j . Une configuration de plusieurs variables aléatoires est une combinaison d'états observés de ces variables. Les paramètres du réseau bayésien ainsi défini sont les distributions de probabilités conditionnelles $\theta_{i,j,k}$ pour $i \in \{1, \dots, N\}$. Nous notons $\theta = \{\theta_{i,j,k}\}$ l'ensemble des paramètres du réseau bayésien défini tel que :

$$\theta_{i,j,k} = \mathbb{P}(X_i = k | pa(X_i) = j) \quad (4.7)$$

Un réseau bayésien est un modèle factorisé. La loi jointe des variables aléatoires peut s'écrire comme un produit de facteurs. Chaque paramètre $\theta_{i,j,k}$ est un facteur de cette loi jointe. L'échantillon de données $D = \{x_1^t, \dots, x_N^t\}$ pour $t \in \{1, \dots, T\}$ contient T réalisations de chacune des variables aléatoires du réseau telles que x_i^t est l'observation numéro t de la variable X_i et $pa(x_i^t)$ est la configuration des prédécesseurs de X_i lorsque $X_i = x_i^t$. La log-vraisemblance des données est développée par l'Équation (4.8) ci-dessous.

$$\begin{aligned} L(D; \theta) &= \log \mathbb{P}(x_1^t, x_2^t, \dots, x_N^t; \theta) \\ &= \sum_{t=1}^T \log \prod_{i=1}^N \theta_{i, pa(x_i^t), x_i^t} \\ &= \sum_{t=1}^T \sum_{i=1}^N \log \theta_{i, pa(x_i^t), x_i^t} \\ &= \sum_{t=1}^T \sum_{i=1}^N \sum_j \sum_k \log \theta_{i,j,k} \mathbb{I}_{\{x_i^t=k, pa(x_i^t)=j\}} \\ &= \sum_{i=1}^N \sum_j \sum_k \sum_{t=1}^T (\mathbb{I}_{\{x_i^t=k, pa(x_i^t)=j\}}) \log \theta_{i,j,k} \\ L(D; \theta) &= \sum_{i=1}^N \sum_j \sum_k N_{i,j,k} \log \theta_{i,j,k} \end{aligned} \quad (4.8)$$

Dans l'Équation (4.8), $N_{i,j,k} = \sum_{t=1}^T (\mathbb{I}_{\{x_i^t=k, pa(x_i^t)=j\}})$ représente le nombre d'occurrences de l'évènement $(X_i = k, pa(X_i) = j)$ dans l'échantillon de données D . Les valeurs $\hat{\theta}_{i,j,k}$ qui maximisent la log-vraisemblance des données s'obtiennent en résolvant N problèmes de maximisation sous contrainte. Le problème de maximisation numéro i est formalisé par l'Équation (4.9).

$$\hat{\theta}_{i,j,k} = \underset{\theta_{i,j,k}}{\text{Arg max}} \sum_{i=1}^N \sum_j \sum_k N_{i,j,k} \log \theta_{i,j,k} \quad \text{sous la contrainte} \quad \sum_k \theta_{i,j,k} = 1 \quad \forall i, j \quad (4.9)$$

Comme le problème de maximisation (4.3), chaque problème i peut être également résolu avec la méthode des multiplicateurs de Lagrange. Les solutions sont :

$$\hat{\theta}_{i,j,k} = \frac{N_{i,j,k}}{\sum_h N_{i,j,h}} = \frac{N_{i,j,k}}{N_{i,j}} \quad (4.10)$$

ou $N_{i,j} = \sum_h N_{i,j,h}$ désigne le nombre d'occurrences de l'évènement $pa(X_i) = j$. Nous notons donc que les valeurs $\hat{\theta}_{i,j,k}$ qui maximisent la log-vraisemblance des données observées d'un réseau bayésien s'obtiennent par un simple calcul de fréquences empiriques, c'est-à-dire en divisant le nombre d'occurrences de l'évènement $(X_i = k, pa(X_i) = j)$ par le nombre d'occurrences de l'évènement $pa(X_i) = j$.

4.3 L'algorithme MLE avec des données incomplètes

4.3.1 Description des données incomplètes

Dans la réalité, un échantillon de données statistiques de plusieurs variables aléatoires est très souvent incomplet, c'est-à-dire que certaines variables ne sont pas toujours observées. De ce fait, les paramètres du modèle statistique ne peuvent pas être estimés en calculant des fréquences empiriques sur l'échantillon de données. Désignons par X l'ensemble des variables observées et Y l'ensemble des variables manquantes ou cachées. La log-vraisemblance $L(x; \theta)$ des données observées peut être très lourde à calculer lorsque l'échantillon est incomplet.

$$L(x, \theta) = \log \sum_y \mathbb{P}(x, y; \theta) \quad (4.11)$$

En effet, il faut énumérer toutes les combinaisons possibles des valeurs des variables non observées, ce qui est calculatoirement infaisable lorsque le nombre de variables manquantes et les dimensions de leurs espaces d'états sont grands. Par conséquent, la fonction de vraisemblance des observations n'est pas calculable en pratique et ne peut donc pas être maximisée avec les méthodes d'optimisation classique telles que la méthode des multiplicateurs de Lagrange [LAGRANGE \[1888\]](#) [ZIEMER \[1989\]](#) et les méthodes de descente de gradient qui nécessitent le calcul de la fonction à maximiser. L'algorithme d'Espérance-Maximisation (EM) [DEMPSTER et collab. \[1977\]](#) [CELEUX et DIEBOLT \[1985\]](#) [CELEUX et GOVAERT \[1991\]](#) [LAURITZEN \[1995\]](#) [LAURITZEN \[1996\]](#) permet de faire face à ce problème.

4.3.2 L'algorithme d'Espérance-Maximisation (EM)

Le problème de l'algorithme MLE sur des données incomplètes peut être résolu avec l'algorithme EM [DEMPSTER et collab. \[1977\]](#) [CELEUX et DIEBOLT \[1985\]](#) [CELEUX et GOVAERT \[1991\]](#) [LAURITZEN \[1995\]](#) [LAURITZEN \[1996\]](#). Comme nous l'avons déjà expliqué ci-dessus, calculer la log-vraisemblance $L(x; \theta)$ des données observées n'est généralement pas possible en pratique. Cependant, le calcul de la log-vraisemblance $L(x, Y; \theta)$ aurait été possible si les variables de l'ensemble Y étaient aussi observées comme celles de l'ensemble X . On aurait donc pu maximiser la log-vraisemblance $L(x, Y; \theta)$ par rapport θ .

Le principe de l'algorithme EM est le suivant. Puisque Y est manquant, au lieu de maximiser $L(x, Y; \theta)$ par rapport θ , l'algorithme EM essaie plutôt de maximiser itérativement l'espérance de la log-vraisemblance des données complétées (x, Y) . En effet, les

variables manquantes ou cachées de l'ensemble Y sont inférées par un algorithme d'inférence probabiliste. Une étude de ces algorithmes d'inférence a été faite dans le Chapitre 3. L'algorithme EM calcule donc l'espérance mathématique de la fonction $L(x, Y; \theta)$ conditionnellement aux observations x sous le modèle statistique de paramètre θ' . Nous introduisons la fonction $Q(\theta, \theta')$ qui se définit aussi comme l'espérance mathématique des variables observées et des variables manquantes (dont les valeurs ont été inférées) conditionnellement aux variables observées sous le modèle de paramètre θ' .

$$Q(\theta, \theta') = \mathbb{E}(L(x, Y; \theta) | x; \theta') \quad (4.12)$$

L'algorithme EM est itératif. Chaque itération r se décompose en deux étapes successives : une première étape d'espérance et une seconde étape de maximisation.

Dans l'étape d'Espérance (étape E), on calcule la fonction $Q(\theta, \theta^r)$, c'est-à-dire la distribution de probabilités des données complétées sous le modèle de paramètre θ^r (les paramètres courants du modèle). En pratique, on calcule des statistiques suffisantes qui résumement cette distribution de probabilités conditionnelles aux observations comme nous l'expliquerons plus tard dans le cas particulier d'un réseau bayésien. L'étape E est la plus complexe et la plus exigeante de l'algorithme EM.

Dans l'étape de Maximisation (étape M), on maximise $Q(\theta, \theta^r)$ par rapport à θ :

$$\theta^{r+1} = \underset{\theta}{\text{Arg max}} Q(\theta, \theta^r) \quad (4.13)$$

Les paramètres θ^{r+1} sont les nouveaux paramètres courants du modèle statistique qui seront utilisés dans l'étape E de l'itération suivante $r + 1$.

L'algorithme EM débute avec une valeur initiale θ^0 des paramètres du modèle. À l'itération 1, l'étape E calcule la fonction $Q(\theta, \theta^0)$, l'étape M calcule $\theta^1 = \underset{\theta}{\text{Arg max}} Q(\theta, \theta^0)$. À l'itération 2, l'étape E calcule $Q(\theta, \theta^1)$, l'étape M calcule $\theta^2 = \underset{\theta}{\text{Arg max}} Q(\theta, \theta^1)$ et ainsi de suite jusqu'à ce que l'algorithme converge vers une valeur stable de θ . La valeur initiale θ^0 peut être choisie aléatoirement, mais nous verrons dans la sous section suivante que l'extremum local de la fonction de vraisemblance des données, c'est-à-dire la valeur stable de θ trouvée par l'algorithme EM, est influencée par le point d'initialisation θ^0 .

4.3.3 Stabilité numérique et convergence de l'algorithme EM

Nous allons démontrer que la séquence $(\theta^r)^r$ des valeurs du paramètre θ calculées par l'algorithme EM converge vers un extremum ou un point selle de la fonction de vraisemblance des données observées. Cette fonction peut avoir plusieurs extrema. L'extremum trouvé par l'algorithme EM n'est pas nécessairement global. En effet, si le point d'initialisation est plus proche d'un extremum local que des autres extrema de la fonction de vraisemblance, l'EM aura tendance à converger vers cet extremum local. Par contre si le point d'initialisation est plus proche de l'extremum global que des autres extrema, l'EM convergera vers l'extremum global. Il convient donc de bien choisir le point initial θ^0 si l'on veut éviter que l'algorithme reste bloqué sur un extremum local de la fonction de vraisemblance des données.

La propriété de stabilité numérique et de convergence de l'algorithme EM est formalisée par l'Équation (4.14). Elle dit que la log-vraisemblance des données observées augmente après chaque itération de l'EM.

$$L(x; \theta^{r+1}) \geq L(x; \theta^r) \quad (4.14)$$

Preuve de la propriété de stabilité numérique : La fonction $Q(\theta, \theta^r)$ peut être décomposée en la somme de deux fonctions :

$$\begin{aligned} Q(\theta, \theta^r) &= \mathbb{E}(L(x, Y; \theta) | x; \theta^r) \\ &= \mathbb{E}(L(x; \theta) | x; \theta^r) + \mathbb{E}(L(Y | x; \theta) | x; \theta^r) \\ &= L(x; \theta) + H(\theta, \theta^r) \end{aligned}$$

Où la fonction $H(\theta, \theta^r) = \mathbb{E}(L(Y | x; \theta) | x; \theta^r)$. Dans l'étape M de l'algorithme EM, la fonction $Q(\theta, \theta^r)$ est maximisée par rapport à θ de sorte que :

$$\begin{aligned} Q(\theta^{r+1}, \theta^r) &\geq Q(\theta^r, \theta^r) \\ L(x; \theta^{r+1}) + H(\theta^{r+1}, \theta^r) &\geq L(x; \theta^r) + H(\theta^r, \theta^r) \\ L(x; \theta^{r+1}) - L(x; \theta^r) &\geq H(\theta^r, \theta^r) - H(\theta^{r+1}, \theta^r) \end{aligned}$$

Pour prouver la propriété de stabilité numérique $L(x; \theta^{r+1}) \geq L(x; \theta^r)$, il suffit donc de montrer l'inégalité suivante :

$$H(\theta^{r+1}, \theta^r) \leq H(\theta^r, \theta^r)$$

Cette inégalité résulte directement de l'inégalité de Jensen. En effet,

$$\begin{aligned} H(\theta^{r+1}, \theta^r) - H(\theta^r, \theta^r) &= \mathbb{E}(L(Y | x; \theta^{r+1}) | x; \theta^r) - \mathbb{E}(L(Y | x; \theta^r) | x; \theta^r) \\ &= \mathbb{E} \left[\log \left(\frac{\mathbb{P}(Y | x; \theta^{r+1})}{\mathbb{P}(Y | x; \theta^r)} \right) | x; \theta^r \right] \end{aligned}$$

Inégalité de Jensen dans le contexte de la théorie des probabilités : si U est une variable aléatoire et ϕ une fonction convexe, alors $\mathbb{E}(\phi(U)) \geq \phi(\mathbb{E}(U))$. Posons $\phi = -\log$ et $U = \frac{\mathbb{P}(Y | x; \theta^{r+1})}{\mathbb{P}(Y | x; \theta^r)}$. U est une variable aléatoire puisque Y (l'ensemble des variables manquantes) est une variable aléatoire. Il résulte de la concavité de la fonction logarithme que,

$$\mathbb{E}(\log(U)) \leq \log(\mathbb{E}(U))$$

Par conséquent, en appliquant l'inégalité de Jensen, nous pouvons écrire que :

$$\mathbb{E} \left[\log \left(\frac{\mathbb{P}(Y | x; \theta^{r+1})}{\mathbb{P}(Y | x; \theta^r)} \right) | x; \theta^r \right] \leq \log \mathbb{E} \left[\left(\frac{\mathbb{P}(Y | x; \theta^{r+1})}{\mathbb{P}(Y | x; \theta^r)} \right) | x; \theta^r \right]$$

L'espérance mathématique qui apparaît dans le terme de droite de l'inéquation ci-dessus est égale à 1. En effet, ce terme peut être interprété comme une somme pondérée par le coefficient $\mathbb{P}(Y | x; \theta^r)$ qui représente la distribution de probabilités de la variable Y conditionnellement à x sous le modèle de paramètre θ^r .

$$\begin{aligned} \mathbb{E} \left[\left(\frac{\mathbb{P}(Y | x; \theta^{r+1})}{\mathbb{P}(Y | x; \theta^r)} \right) | x; \theta^r \right] &= \sum_Y \left[\frac{\mathbb{P}(Y | x; \theta^{r+1})}{\mathbb{P}(Y | x; \theta^r)} \right] \mathbb{P}(Y | x; \theta^r) \\ &= \sum_Y \mathbb{P}(Y | x; \theta^{r+1}) = 1 \end{aligned}$$

Par conséquent $H(\theta^{r+1}, \theta^r) - H(\theta^r, \theta^r) \leq 0$. Ce dernier résultat termine la démonstration de la propriété de stabilité numérique de l'algorithme EM donnée par l'Équation (4.14). Si la fonction de vraisemblance des données observées $L(x; \theta)$ est continue par rapport à θ , et si l'espace des valeurs possibles de θ est un ensemble borné, il résulte de la propriété de stabilité numérique (4.14) que la séquence $(\theta^r)^r$ des valeurs de θ calculées par l'algorithme EM converge vers un extremum local ou un point selle de la fonction de vraisemblance des données observées.

4.4 L'algorithme EM sur un réseau bayésien

L'algorithme EM a été adapté dans LAURITZEN [1995] pour le cas particulier d'un modèle statistique basé sur un réseau bayésien de N variables aléatoires $\{X_1, \dots, X_N\}$. Dans un tel modèle, les variables sont les noeuds du graphe de dépendances. Certains noeuds X_i peuvent ne pas être observés. Dans ce contexte, l'algorithme EM peut être utilisé pour inférer les probabilités conditionnelles $\hat{\theta}_{i,j,k} = \mathbb{P}(X_i = k | pa(X_i) = j)$ à partir des données observées, c'est-à-dire à partir des noeuds du graphe de dépendances qui ne sont pas manquants.

Définissons explicitement le fait que certains noeuds du réseau bayésien soient observés tandis que d'autres noeuds ne le sont pas. Nous notons $x_i^{o,t}$ la valeur prise par le noeud X_i dans la t -ième observation du réseau bayésien. La lettre "o" signifie que X_i est observée. $X_i^{m,t}$ désigne la variable aléatoire associée au noeud X_i lorsqu'il n'est pas observé dans la t -ième réalisation du réseau bayésien. La lettre "m" signifie que X_i est manquante. Une observation ou une configuration t du réseau désigne la combinaison $X^{o,t}$ des valeurs prises par les noeuds observés. Les ensembles $x^{o,t}$ et $X^{m,t}$ sont respectivement appelés données observées et données manquantes pour la configuration t du réseau bayésien. L'échantillon de données D contenant T observations du réseau bayésien, utilisé par l'algorithme EM pour l'estimation des paramètres du réseau, se définit donc par $D = \{(x^{o,1}, X^{m,1}), \dots, (x^{o,T}, X^{m,T})\}$.

4.4.1 Étape d'Espérance

Nous avons déjà montré dans l'Équation (4.8) que la log-vraisemblance des données observées $x^o = \{x^{o,1}, x^{o,2}, \dots, x^{o,T}\}$ dans le cas d'un réseau bayésien s'écrit comme suit :

$$L(x^o; \theta) = \sum_{i=1}^N \sum_j \sum_k \sum_{t=1}^T (\mathbb{I}_{\{x_i^{o,t}=k, pa(x_i^{o,t})=j\}}) \log \theta_{i,j,k}$$

La fonction $Q(\theta, \theta^r)$ à maximiser dans l'étape M peut se développer comme le montre l'Équation (4.15) :

$$\begin{aligned} Q(\theta, \theta^r) &= \mathbb{E}(\log \mathbb{P}((x^{o,1}, X^{m,1}), \dots, (x^{o,T}, X^{m,T}); \theta | x^o; \theta^r)) \\ &= \sum_{i=1}^N \sum_j \sum_k \sum_{t=1}^T \mathbb{E}(\mathbb{I}_{\{X_i=k, pa(X_i)=j\}} | x^{o,t}; \theta^r) \log \theta_{i,j,k} \\ &= \sum_{i=1}^N \sum_j \sum_k \sum_{t=1}^T \mathbb{P}(X_i = k, pa(X_i) = j | x^{o,t}; \theta^r) \log \theta_{i,j,k} \\ &= \sum_{i=1}^N \sum_j \sum_k \sum_{t=1}^T \gamma_{i,j,k}^t \log \theta_{i,j,k} \end{aligned} \tag{4.15}$$

Notons que le passage de la deuxième à la troisième ligne de l'Équation (4.15) se fait en appliquant la propriété énoncée comme suit : l'espérance de la fonction indicatrice d'un ensemble est égale à la probabilité de cet ensemble.

$$\begin{aligned} \gamma_{i,j,k}^t &= \mathbb{E}(\mathbb{I}_{\{X_i=k, pa(X_i)=j\}} | x^{o,t}; \theta^r) \\ &= \mathbb{P}(X_i = k, pa(X_i) = j | x^{o,t}; \theta^r) \end{aligned} \quad (4.16)$$

Nous appelons réalisations attendues ou expérées (expected counts en anglais) des variables manquantes de la famille $f = (X_i \cup pa(X_i))$, la distribution de probabilités jointe des variables de la famille f conditionnellement à l'ensemble des observations définit par $x^o = \{x^{o,1}, \dots, x^{o,T}\}$. Nous notons cette distribution $\hat{N}(X_i = k, pa(X_i) = j | x^o; \theta^r)$ telle que :

$$\hat{N}(X_i = k, pa(X_i) = j | x^o; \theta^r) = \sum_{t=1}^T \gamma_{i,j,k}^t \quad (4.17)$$

En substituant l'Équation (4.17) dans l'Équation (4.15), on obtient l'expression finale de la fonction $Q(\theta, \theta^r)$ des réalisations expérées des variables non observées de la famille $(X_i, pa(X_i))$ calculée dans l'étape E de l'itération r .

$$Q(\theta, \theta^r) = \sum_{i=1}^N \sum_j \sum_k \hat{N}(X_i = k, pa(X_i) = j | x^o; \theta^r) \log \theta_{i,j,k} \quad (4.18)$$

De façon synthétique, nous pouvons dire que l'étape E calcule pour chaque noeud X_i du réseau, les réalisations expérées $\sum_{t=1}^T \gamma_i^t$ telle que γ_i^t est distribution de probabilités jointes conditionnelles $\mathbb{P}(X_i, pa(X_i) | x^{o,t}; \theta^r)$ à l'itération r . Cette distribution de probabilités est calculée en utilisant le graphe de dépendance du réseau bayésien.

4.4.2 Calculs de l'étape E basés sur la structure graphique du réseau

N'importe lequel des algorithmes d'inférence probabiliste étudiés dans le Chapitre 3 peut être utilisé pour calculer les distributions γ_i^t . Nous développons ici les calculs de l'étape E en utilisant l'algorithme d'inférence sur un arbre de jonction LAURITZEN et SPIEGELHALTER [1988] JENSEN et collab. [1990] LAURITZEN [1996] JENSEN et NIELSEN [2007] puisqu'il convient à n'importe quelle topologie de réseau bayésien quelle que soit sa complexité. L'algorithme d'inférence sur un arbre de jonction, ayant déjà été étudié dans le Chapitre 3, nous ne montrerons ici que son utilisation dans l'étape E de l'algorithme EM.

Au début d'une itération r , l'arbre de jonction est initialisé en associant un potentiel à chaque clique de tout couple de cliques (C_p, C_q) adjacentes ainsi qu'à leur séparateur commun S_{pq} . Les valeurs initiales des potentiels sont calculées à partir des paramètres courants θ^r .

$$\begin{aligned} \phi_{S_{pq}} &= 1 \\ \phi_{C_p}(\theta^r) &= \prod_{X_i \in C_p, pa(X_i) \subset C_p \vee pa(X_i) = \emptyset} \mathbb{P}(X_i | pa(X_i); \theta^r) \end{aligned} \quad (4.19)$$

Nous notons $JT(\theta^r)$, l'arbre de jonction dont les potentiels des cliques sont initialisés à partir des paramètres θ^r . Pour chaque observation t du réseau bayésien, l'algorithme de propagation des évidences $x^{o,t}$ sur l'arbre de jonction $JT(\theta^r)$ actualise le potentiel de

chaque clique de tout couple de cliques (C_p, C_q) adjacentes au travers de leur séparateur commun S_{pq} .

$$\phi_{S_{pq}}^*(\theta^r) = \sum_{C_p \setminus S_{pq}} \phi_{C_p}^*(\theta^r) \quad (4.20)$$

$$\phi_{C_q}^*(\theta^r) = \phi_{C_q}(\theta^r) \frac{\phi_{S_{pq}}^*(\theta^r)}{\phi_{S_{pq}}(\theta^r)} \quad (4.21)$$

En supposant que le potentiel de la clique C_p a déjà été actualisé, l'algorithme de propagation procède en deux étapes. Le potentiel du séparateur S_{pq} est actualisé par l'opération de marginalisation (4.20). Ensuite le potentiel de la clique C_q est actualisé par l'Équation (4.21). La clique C_q absorbe donc les évidences en provenance de la clique adjacente C_p au travers du séparateur S_{pq} . Le lecteur peut se référer au Chapitre 3 pour avoir plus de détails sur l'algorithme d'inférence sur un arbre de jonction.

Lorsque l'algorithme d'inférence sur l'arbre de jonction $JT(\theta^r)$ se termine, le potentiel actualisé $\phi_{C_p}^*$ d'une quelconque clique C_p devient la distribution de probabilités jointes des évidences $x^{o,t}$ et des variables aléatoires appartenant à la clique C_p .

$$\phi_{C_p}^*(\theta^r) = \mathbb{P}(C_p, x^{o,t}; \theta^r) \quad (4.22)$$

A l'itération r , nous pouvons donc calculer la vraisemblance des évidences $x^{o,t}$ sous le modèle de paramètre θ^r en marginalisant sur le potentiel actualisé de la clique C_p .

$$\mathbb{P}(x^{o,t}; \theta^r) = \sum_{C_p} \phi_{C_p}^*(\theta^r) \quad (4.23)$$

L'Équation (4.19) montre qu'il existe au moins une clique C_z dans l'arbre de jonction telle que la famille :

$$f = \{X_i \cup pa(X_i)\} \subseteq C_z \quad (4.24)$$

Ce qui signifie que pour calculer la distribution $\gamma_i^t = \mathbb{P}(X_i, pa(X_i) | x^{o,t}; \theta^r)$, nous n'avons qu'à choisir parmi les cliques de l'arbre de jonction $JT(\theta^r)$, une clique C_z qui vérifie l'assertion (4.24) ci-dessus. Par conséquent, nous pouvons calculer γ_i^t par une simple opération de marginalisation sur le potentiel actualisé de la clique C_z .

$$\gamma_i^t = \mathbb{P}(X_i, pa(X_i) | x^{o,t}; \theta^r) = \frac{\sum_{C_z \setminus f} \phi_{C_z}^*(\theta^r)}{\mathbb{P}(x^{o,t}; \theta^r)} \quad (4.25)$$

4.4.3 Étape de Maximisation

À l'étape M de chaque itération r , la fonction $Q(\theta, \theta^r)$ est maximisée par rapport à $\theta_{i,j,k}$ sous la contrainte $\sum_k \theta_{i,j,k} = 1, \forall i, j$. La solution de ce problème de maximisation sous contrainte est la valeur θ^{r+1} des paramètres qui seront utilisés pour initialiser l'arbre de jonction à l'itération $r+1$ suivante.

$$\begin{aligned} \sum_k \theta_{i,j,k} &= 1 \quad \forall i, j \\ \theta_{i,j,k}^{r+1} &= \underset{\theta_{i,j,k}}{\text{Arg max}} \sum_{i,j,k} \log \theta_{i,j,k} \hat{N}(X_i = k, pa(X_i) = j | x^o; \theta^r) \end{aligned} \quad (4.26)$$

De la même façon que le problème de maximisation sous contrainte (4.3), ce problème de maximisation sous contrainte se résout également en utilisant la méthode des multiplicateurs de Lagrange (4.4).

$$\theta_{i,j,k}^{r+1} = \frac{\hat{N}(X_i = k, pa(X_i) = j|x^o)}{\sum_h \hat{N}(X_i = h, pa(X_i) = j|x^o)} \quad (4.27)$$

4.4.4 Pseudo-code de l'algorithme EM sur un réseau bayésien

Nous présentons ci-dessous le pseudo-code de l'algorithme EM appliqué pour apprendre artificiellement les distributions de probabilités conditionnelles d'un réseau bayésien à partir d'un échantillon de données incomplet. La convergence et la terminaison de l'algorithme sont régulées par un nombre $\epsilon > 0$. Pour éviter d'interrompre les itérations sans que l'algorithme ait convergé, ϵ doit être très petit.

Algorithm 1 Algorithme EM pour un réseau bayésien

Require: $\epsilon > 0, \theta^0 = \{\theta_{i,j,k}; \forall i, j, k\}$ et les données $D = \{x^o, X^m\}$ de taille T

Ensure: calcul du vecteur de paramètres $\hat{\theta} = \{\hat{\theta}_{i,j,k}, \forall i, j, k\}$

- 1: $r \leftarrow 0$
 - 2: **repeat**
 - 3: **étape E :** Pour chaque noeud X_i , on calcule les réalisations expérées :
 - 4: $\hat{N}(X_i = k, pa(X_i) = j|x^o; \theta^r) = \sum_{t=1}^T \gamma_{i,j,k}^t \quad \forall j, k$
 - 5: tel que $\gamma_{i,j,k}^t = \mathbb{P}(X_i = k, pa(X_i) = j|x^{o,t}; \theta^r)$
 - 6: **étape M :** On utilise les réalisations expérées comme si elles étaient réelles :
 - 7: $\hat{\theta}_{i,j,k} = \frac{\hat{N}(X_i=k, pa(X_i)=j|x^o)}{\sum_h \hat{N}(X_i=h, pa(X_i)=j|x^o)}$
 - 8: **Calcul de la log-vraisemblance des données D :**
 - 9: $\log \mathbb{P}(D; \theta^r) = \sum_{t=1}^T \log \mathbb{P}(x^{o,t}; \theta^r)$, voir l'Équation (4.23)
 - 10: $\theta^{r+1} \leftarrow \hat{\theta}$ tel que $\hat{\theta} = \{\hat{\theta}_{i,j,k}, \forall i, j, k\}, r \leftarrow r + 1$
 - 11: **until** $|\log(\mathbb{P}(D; \theta^r)) - \log(\mathbb{P}(D; \theta^{r-1}))| \leq \epsilon$
 - 12: **return** le vecteur de paramètres $\hat{\theta}$
-

La Figure 4.1 est un schéma synoptique qui synthétise le principe général de l'algorithme EM d'après le tutoriel FRIEDMAN et GOLDSZMIDT [1998]. Sur cette figure (G, θ^r) désigne le graphe de dépendance G et les paramètres θ^r du réseau bayésien.

4.4.5 Exemple illustratif de l'algorithme EM sur un réseau bayésien

Considérons le réseau bayésien R de la Figure 3.1 du Chapitre 3. Supposons que les paramètres initiaux θ^0 du réseau sont donnés par la Table 4.2. À titre d'exemple, considérons l'échantillon de données incomplet D décrit par la Table 4.1 qui contient trois observations du réseau R . Les deux premières lignes de données contiennent chacune deux variables manquantes. La troisième ligne est une observation complète (toutes les cinq variables du réseau sont observées). Nous allons développer la première itération de l'algorithme EM sur le réseau R à partir de l'échantillon de données incomplet D . Rappelons que les variables du réseau sont binaires. La variable A par exemple prend ses valeurs dans l'ensemble ordonné $\{\neg a, +a\}$.

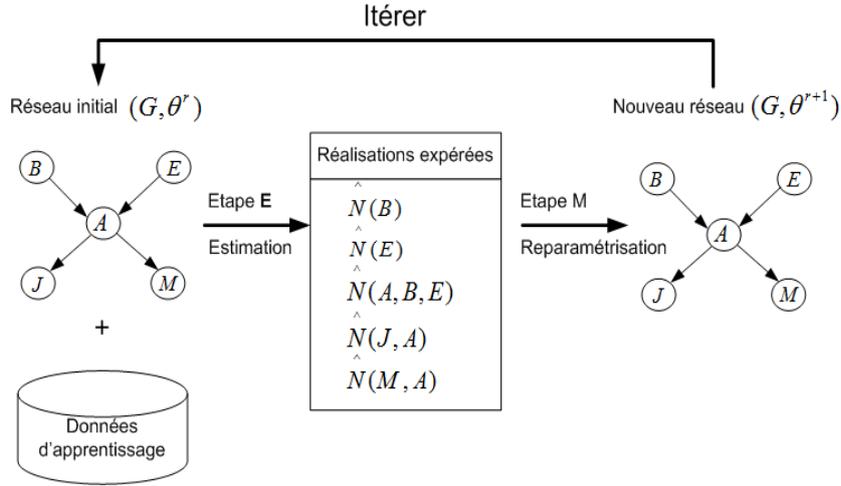


FIGURE 4.1 – Principe général de l’algorithme EM.

TABLE 4.1 – Un exemple d’échantillon de données incomplet.

E	B	A	J	M
-	-	+a	+j	+m
$\neg e$	$\neg b$	$\neg a$	-	-
+e	$\neg b$	+a	+j	+m

Première itération : Les cliques ABE, AJ et AM de l’arbre de jonction équivalent au réseau R (voir la Figure 3.9) sont initialisés avec des potentiels calculés à partir des paramètres courants θ^0 donnés par la Table 4.2. La valeur initiale du potentiel du séparateur est $\phi_A = 1$.

$$\phi_{AJ} = \mathbb{P}(J|A) = \begin{bmatrix} 0.90 & 0.10 \\ 0.05 & 0.95 \end{bmatrix}_{(a,j)} \quad \phi_{AM} = \mathbb{P}(M|A) = \begin{bmatrix} 0.70 & 0.30 \\ 0.01 & 0.99 \end{bmatrix}_{(a,m)}$$

$$\phi_{ABE} = \mathbb{P}(A|B, E)P(B)P(E) = \begin{bmatrix} 9.47e-01 & 4.98e-02 \\ 1.19e-04 & 1.87e-03 \\ 2.89e-04 & 7.08e-04 \\ 2.00e-09 & 1.99e-06 \end{bmatrix}_{(b,e,a)}$$

Etape E : On calcule pour chaque noeud du réseau, les réalisations expérées de sa famille. Les réalisations expérées d’une famille (un noeud du réseau et ses parents) s’obtiennent en associant un nombre compris dans l’intervalle $[0, 1]$ à chaque combinaison de valeurs (configuration) des variables de cette famille. Pour une quelconque configuration c de la famille f considérée et pour une observation ou une configuration t du réseau, on distingue trois situations possibles. Si la configuration c est consistante avec la configuration t , c’est-à-dire que la valeur prise par chaque variable de la famille f dans la configuration c , est la même que dans la configuration t du réseau, alors le nombre associé à la configuration c est 1. Si c et t sont inconsistant, le nombre associé à c est 0. Si au moins une variable de la famille f n’est pas observée dans la configuration t du réseau, le nombre associé à c est la probabilité conditionnelle $\mathbb{P}(c|x^o, t)$. Cette probabilité conditionnelle est calculée par l’algorithme d’inférence sur l’arbre de jonction JT(θ^0).

TABLE 4.2 – Paramètres θ^0 du réseau bayésien 3.1.

B	$\mathbb{P}(+b) = .001$
E	$\mathbb{P}(+e) = .002$
J	$\mathbb{P}(+j +a) = .950, \mathbb{P}(+j \neg a) = .100$
M	$\mathbb{P}(+m +a) = .990, \mathbb{P}(+m \neg a) = .300$
A	$\mathbb{P}(+a +b,+e) = .999, \mathbb{P}(+a +b,\neg e) = .710$
	$\mathbb{P}(+a \neg b,+e) = .940, \mathbb{P}(+a \neg b,\neg e) = .050$

Réalisations expérées de la famille (B) :

$$\begin{aligned}
 \hat{N}(B|x^o; \theta^0) &= \sum_{t=1}^3 \Upsilon_B^t \\
 &= \mathbb{P}(B|+a,+j,+m) + \mathbb{P}(B|\neg e,\neg b,\neg a) + \mathbb{P}(B|+e,\neg b,+a,+j,+m) \\
 &= \begin{bmatrix} 0.986 \\ 0.014 \end{bmatrix}_{(b)} + \begin{bmatrix} 1.000 \\ 0.000 \end{bmatrix}_{(b)} + \begin{bmatrix} 1.000 \\ 0.000 \end{bmatrix}_{(b)} \\
 &= \begin{bmatrix} 2.986 \\ 0.014 \end{bmatrix}_{(b)}
 \end{aligned}$$

Réalisations expérées de la famille (E) :

$$\begin{aligned}
 \hat{N}(E|x^o; \theta^0) &= \sum_{t=1}^3 \Upsilon_E^t \\
 &= \mathbb{P}(E|+a,+j,+m) + \mathbb{P}(E|\neg e,\neg b,\neg a) + \mathbb{P}(E|+e,\neg b,+a,+j,+m) \\
 &= \begin{bmatrix} 0.964 \\ 0.036 \end{bmatrix}_{(e)} + \begin{bmatrix} 1.000 \\ 0.000 \end{bmatrix}_{(e)} + \begin{bmatrix} 0.000 \\ 1.000 \end{bmatrix}_{(e)} \\
 &= \begin{bmatrix} 1.964 \\ 1.036 \end{bmatrix}_{(e)}
 \end{aligned}$$

Réalisations expérées de la famille (A, B, E) :

$$\begin{aligned}
 \hat{N}(A,B,E|x^o; \theta^0) &= \sum_{t=1}^3 \Upsilon_A^t \\
 &= \mathbb{P}(A,B,E|+a,+j,+m) + \mathbb{P}(A,B,E|\neg e,\neg b,\neg a) + \mathbb{P}(A,B,E|+e,\neg b,+a,+j,+m) \\
 &= \begin{bmatrix} 9.50e-01 & 0.000 \\ 3.58e-02 & 0.000 \\ 1.35e-02 & 0.000 \\ 3.81e-05 & 0.000 \end{bmatrix}_{(b,e,a)} + \begin{bmatrix} 1.000 & 0.000 \\ 0.000 & 0.000 \\ 0.000 & 0.000 \\ 0.000 & 0.000 \end{bmatrix}_{(b,e,a)} + \begin{bmatrix} 0.000 & 0.000 \\ 0.000 & 1.000 \\ 0.000 & 0.000 \\ 0.000 & 0.000 \end{bmatrix}_{(b,e,a)} \\
 &= \begin{bmatrix} 1.95e+00 & 0.000 \\ 3.58e-02 & 1.000 \\ 1.35e-02 & 0.000 \\ 3.81e-05 & 0.000 \end{bmatrix}_{(b,e,a)}
 \end{aligned}$$

Réalisations expérées de la famille (J, A) :

$$\begin{aligned}
 \hat{N}(J, A | x^o; \theta^0) &= \sum_{t=1}^3 Y_J^t \\
 &= \mathbb{P}(J, A | +a, +j, +m) + \mathbb{P}(J, A | \neg e, \neg b, \neg a) + \mathbb{P}(J, A | +e, \neg b, +a, +j, +m) \\
 &= \begin{bmatrix} 0.000 & 0.000 \\ 0.000 & 1.000 \end{bmatrix}_{(a,j)} + \begin{bmatrix} 0.900 & 0.100 \\ 0.000 & 1.000 \end{bmatrix}_{(a,j)} + \begin{bmatrix} 0.000 & 0.000 \\ 0.000 & 1.000 \end{bmatrix}_{(a,j)} \\
 &= \begin{bmatrix} 0.900 & 0.100 \\ 0.000 & 2.000 \end{bmatrix}_{(a,j)}
 \end{aligned}$$

Réalisations expérées de la famille (M, A) :

$$\begin{aligned}
 \hat{N}(M, A | x^o; \theta^0) &= \sum_{t=1}^3 Y_M^t \\
 &= \mathbb{P}(M, A | +a, +j, +m) + \mathbb{P}(M, A | \neg e, \neg b, \neg a) + \mathbb{P}(M, A | +e, \neg b, +a, +j, +m) \\
 &= \begin{bmatrix} 0.000 & 0.000 \\ 0.000 & 1.000 \end{bmatrix}_{(a,m)} + \begin{bmatrix} 0.700 & 0.300 \\ 0.000 & 0.000 \end{bmatrix}_{(a,m)} + \begin{bmatrix} 0.000 & 0.000 \\ 0.000 & 1.000 \end{bmatrix}_{(a,m)} \\
 &= \begin{bmatrix} 0.700 & 0.300 \\ 0.000 & 2.000 \end{bmatrix}_{(a,m)}
 \end{aligned}$$

Etape M : On utilise les réalisations expérées comme si elles étaient réelles. Les distributions de probabilités conditionnelles du réseau R qui seront utilisées par l'EM à la prochaine itération s'obtiennent en faisant de simples calculs de fréquences empiriques.

$$\begin{aligned}
 \hat{\theta}_B = \mathbb{P}(B) &= \frac{\hat{N}(B | x^o; \theta^0)}{\sum_b \hat{N}(B | x^o; \theta^0)} = \frac{1}{3} \begin{bmatrix} 2.986 \\ 0.014 \end{bmatrix}_{(b)} \\
 &= \begin{bmatrix} 0.995 & 0.005 \end{bmatrix}_{(b)}
 \end{aligned}$$

$$\begin{aligned}
 \hat{\theta}_E = \mathbb{P}(E) &= \frac{\hat{N}(E | x^o; \theta^0)}{\sum_e \hat{N}(E | x^o; \theta^0)} = \frac{1}{3} \begin{bmatrix} 1.964 \\ 1.036 \end{bmatrix}_{(e)} \\
 &= \begin{bmatrix} 0.655 & 0.345 \end{bmatrix}_{(e)}
 \end{aligned}$$

$$\begin{aligned}
 \hat{\theta}_A = \mathbb{P}(A | B, E) &= \frac{\hat{N}(A, B, E | x^o; \theta^0)}{\sum_a \hat{N}(A, B, E | x^o; \theta^0)} \\
 &= \begin{bmatrix} 1.95e+00 & 1.03e+00 \\ 1.35e-02 & 3.81e-05 \end{bmatrix}_{(b,e)}^{-1} \begin{bmatrix} \begin{bmatrix} 1.95e+00 & 0.000 \\ 3.58e-02 & 1.000 \end{bmatrix} \\ \begin{bmatrix} 1.35e-02 & 0.000 \\ 3.81e-05 & 0.000 \end{bmatrix} \end{bmatrix}_{(b,e,a)} \\
 &= \begin{bmatrix} \begin{bmatrix} 9.99e-01 & 5.12e-10 \\ 3.45e-02 & 9.65e-01 \end{bmatrix} \\ \begin{bmatrix} 9.99e-01 & 7.40e-08 \\ 9.99e-01 & 2.62e-05 \end{bmatrix} \end{bmatrix}_{(b,e,a)}
 \end{aligned}$$

$$\begin{aligned}\hat{\theta}_J = \mathbb{P}(J|A) &= \frac{\hat{N}(J, A|x^o; \theta^0)}{\sum_j \hat{N}(J, A|x^o; \theta^0)} = \begin{bmatrix} 1.000 \\ 2.000 \end{bmatrix}_{(a)}^{-1} \begin{bmatrix} 0.900 & 0.100 \\ 0.000 & 2.000 \end{bmatrix}_{(j,a)} \\ &= \begin{bmatrix} 0.900 & 0.100 \\ 0.001 & 0.999 \end{bmatrix}_{(j,a)}\end{aligned}$$

$$\begin{aligned}\hat{\theta}_M = \mathbb{P}(M|A) &= \frac{\hat{N}(M, A|x^o; \theta^0)}{\sum_m \hat{N}(M, A|x^o; \theta^0)} = \begin{bmatrix} 1.000 \\ 2.000 \end{bmatrix}_{(a)}^{-1} \begin{bmatrix} 0.700 & 0.300 \\ 0.000 & 2.000 \end{bmatrix}_{(a,m)} \\ &= \begin{bmatrix} 0.700 & 0.300 \\ 0.001 & 0.999 \end{bmatrix}_{(a,m)}\end{aligned}$$

4.5 Discussion et conclusion

Nous avons étudié dans ce chapitre l'apprentissage artificiel des paramètres d'un modèle statistique à partir d'un échantillon de données. Deux cas de figure se distinguent. L'apprentissage des paramètres à partir d'un échantillon de données complet et l'apprentissage à partir d'un échantillon de données incomplet. Indépendamment du cas considéré, le principe de l'apprentissage est de chercher les valeurs des paramètres du modèle qui expliquent ou qui conviennent mieux à l'échantillon de données observées. Il s'agit de chercher les paramètres qui maximisent la vraisemblance de l'échantillon. Cependant, selon que l'échantillon soit complet ou incomplet, la méthode utilisée par le système apprenant pour résoudre ce problème de maximisation diffère.

Lorsque l'échantillon est complet, l'algorithme d'apprentissage calcule la fonction de vraisemblance des données qu'il peut ensuite maximiser en utilisant des techniques d'optimisation classiques telles que les méthodes du gradient, de Newton, ou des multiplieurs de Lagrange. Par contre avec un échantillon incomplet, la fonction de vraisemblance des données est pratiquement incalculable compte tenu du nombre très important de combinaisons de valeurs possibles des variables aléatoires manquantes. On est maintenant face à un problème d'optimisation non linéaire avec potentiellement de très nombreux optima locaux. De ce fait, les techniques d'optimisation classiques qui nécessitent de calculer au préalable la fonction à optimiser ne peuvent plus être utilisées. L'algorithme d'Espérance-Maximisation (EM) permet de contourner cette problématique.

À défaut de calculer la fonction de vraisemblance des données, l'EM calcule et maximise plutôt de façon itérative l'espérance mathématique des données complétées (par un algorithme d'inférence probabiliste) conditionnellement aux données observées. Par contre, l'extremum de la fonction de vraisemblance des données, calculé par l'EM n'est pas nécessairement global. En effet, cet extremum dépend encore du point d'initialisation de la fonction de vraisemblance. Il convient donc d'avoir un minimum de connaissances sur les données empiriques si l'on souhaite choisir un point d'initialisation de telle sorte que l'algorithme EM ne reste pas bloqué sur un extremum local donc l'image par la fonction de vraisemblance est très inférieure à celle de l'extremum global.

Chapitre 5

Modèle générique pour le diagnostic automatique d'un réseau de télécommunication

« Model traversing techniques use formal representation of a communication system with clearly marked relationships among network entities. »

Malgorzata, 2004

Sommaire

5.1 Introduction	80
5.2 Propagation de pannes dans un réseau	80
5.3 Description du modèle générique	81
5.3.1 Principe général	82
5.3.2 Première couche	83
5.3.3 Deuxième couche	84
5.3.4 Troisième couche	84
5.4 Formalisme du modèle générique	85
5.5 Calcul de diagnostic du modèle générique	86
5.6 Capacité de reconfiguration du modèle générique	87
5.6.1 Les agents de reconfiguration	87
5.6.2 Reconfiguration initiée par la couche 2	87
5.6.3 Reconfiguration initiée par la couche 1	89
5.7 Conclusion et perspectives	91

5.1 Introduction

La nature dynamique et distribuée d'un réseau de télécommunication complexifie la conception de méthodes à base de modèles pour le diagnostic automatique de ces réseaux. En effet, ces méthodes de diagnostic développent des raisonnements basés sur une représentation explicite du réseau à diagnostiquer. Un tel effort de modélisation nécessite beaucoup de connaissances dites "expertes" sur la structure et le fonctionnement du réseau. Cependant, malgré cet effort important, le modèle obtenu n'est qu'une image figée du réseau et peut devenir rapidement obsolète si le réseau est très dynamique. Nous proposons dans ce chapitre un modèle générique reconfigurable en trois couches pour le diagnostic automatique des réseaux de télécommunications. Ce modèle intègre des capacités de reconfiguration dynamique afin de suivre ou de s'adapter automatiquement aux éventuels changements sur la topologie et le comportement du réseau encore appelé propagation de pannes dans le contexte du diagnostic des réseaux. Nous définissons et décrivons ce phénomène de propagation de pannes dans la Section 2. Les différentes couches du modèle générique sont décrites dans la Section 3. La Section 4 expose le formalisme mathématique du modèle générique. Les capacités de reconfiguration du modèle sont décrites, formalisées par des machines à états et illustrées par des exemples de scénarios dans la Section 6. Nous terminons ce chapitre par une conclusion et quelques perspectives discutées dans la Section 7.

5.2 Propagation de pannes dans un réseau

Dans le contexte du diagnostic des réseaux, une panne se définit comme la cause racine d'une ou de plusieurs anomalies survenues sur le réseau et observées sous forme d'alarmes ou de paramètres hors du standard de fonctionnement nominal du réseau. Les alarmes ou symptômes sont des manifestations externes des pannes [JAKOBSON et WEISSMANI \[1993\]](#). Puisqu'un réseau de télécommunication est par nature un système distribué, une panne survenue sur le réseau se propage et engendre d'autres pannes et alarmes qui à leurs tours en engendrent d'autres. Ce phénomène est appelé propagation de pannes. Il s'agit d'un ensemble de relations de causes à effets entre les causes racines, les causes intermédiaires et les alarmes observées. La conséquence de la propagation de pannes et d'alarmes est qu'une seule et unique cause racine peut engendrer une cascade complexe et distribuée de plusieurs autres pannes et de leurs alarmes correspondantes [HOUNKONNOU \[2013\]](#). Ceci est d'autant plus vrai lorsque plusieurs pannes se propagent simultanément au travers de l'architecture distribuée d'un réseau de télécommunication de très grande dimension.

Comme dans tout système distribué, dans un réseau, les pannes se propagent typiquement entre les composants connectés du réseau. Nous appelons ce type de propagation, la propagation distribuée de pannes, c'est-à-dire qu'un dysfonctionnement sur un composant affecte l'état d'un ou de plusieurs autres composants voisins qui lui sont connectés physiquement ou même logiquement. Par exemple, une atténuation trop importante sur une liaison optique entre deux équipements quelconques engendre une dégradation voire même la perte de la communication entre ces deux équipements; dans un système de partage temporel d'un canal de communication entre plusieurs équipements, un équipement qui ne respecte pas correctement la fenêtre de temps qui lui est allouée peut provoquer la désynchronisation de plusieurs autres équipements qui utilisent le même canal. Cependant, il existe aussi des situations pour lesquelles une panne

peut se propager localement sur un composant du réseau sans affecter l'état d'un composant voisin. Nous appelons ceci, la propagation locale de pannes. Par exemple, une mauvaise alimentation d'un émetteur/récepteur optique peut engendrer un courant très faible qui à son tour engendre une très faible puissance d'émission.

La propagation de pannes dans un réseau de télécommunication de grande dimension est un phénomène très complexe principalement dû à son caractère non déterministe. En effet, une seule panne peut générer plusieurs alarmes distinctes et une alarme peut être déclenchée par plusieurs pannes distinctes. Cette complexité relative au contexte réseau est d'autant plus grande que les alarmes générées par les équipements correspondent à des comportements très variés du réseau de télécommunication :

- la multiplication ou la redondance d'alarmes due à plusieurs tentatives sans succès d'une même opération telle que la synchronisation entre deux équipements du réseau,
- l'absence d'alarmes pouvant être due soit à une perte (dans le cas d'une signalisation en bande "inband signaling"), soit à un filtrage ou à des règles de priorités entre les alarmes mises en place par les fabricants d'équipements ou par l'opérateur conduisant ainsi à une propagation incomplète des pannes,
- les alarmes transitoires dues à des états temporaires du réseau parmi lesquelles il faut pouvoir identifier et distinguer les alarmes transitoires non répétitives de celles qui se répètent régulièrement, notifiant des pannes chroniques. Il faut donc pouvoir distinguer les « pépins » des pannes chroniques qui non seulement affectent la qualité des communications lorsqu'elles surviennent, mais aussi indiquent une dégradation des performances et d'autres pannes potentielles dans le réseau [SASI-SEKHARAM et collab. \[1996\]](#),
- les alarmes ambiguës qui apparaissent très souvent lorsque les anomalies d'une couche sont reportées à la couche supérieure [HOUNKONNOU \[2013\]](#),
- les alarmes retardées du fait que la cascade d'alarmes engendrée par une seule panne n'est pas toujours une séquence temporelle totalement ordonnée puisque même si les horloges d'équipements sont parfaitement synchronisées, les temps de propagation d'alarmes varient selon les supports de transmissions.

En somme, la propagation de pannes dans un réseau de télécommunication est un phénomène dynamique et distribué par ce que le réseau l'est aussi, non déterministe par ce que les relations de causes à effets entre causes racines, causes intermédiaires et alarmes ne sont pas toujours absolument certaines. En pratique, sur un réseau, la propagation distribuée de pannes entre composants connectés est plus fréquente que celle locale à un équipement.

5.3 Description du modèle générique

Le modèle générique propose une séparation entre la propagation locale de pannes sur un composant du réseau et la propagation distribuée de pannes entre composants connectés. Cette séparation permet également de dissocier la modélisation de la topologie du réseau de celle de son comportement ou propagation de pannes.

5.3.1 Principe général

Le modèle générique est construit en trois couches. La couche 1 est un graphe non orienté qui modélise la topologie du réseau. Chaque noeud de ce graphe non orienté est un graphe de dépendances acycliques (DAG) qui symbolise un composant du réseau et modélise la propagation distribuée de pannes entre ce composant et les composants voisins auxquels il est connecté. La couche 2 est un ensemble de DAGs modélisant chacun la propagation locale de pannes sur un composant du réseau. La couche 3 est une représentation simplifiée et équivalente de la combinaison des couches 1 et 2.

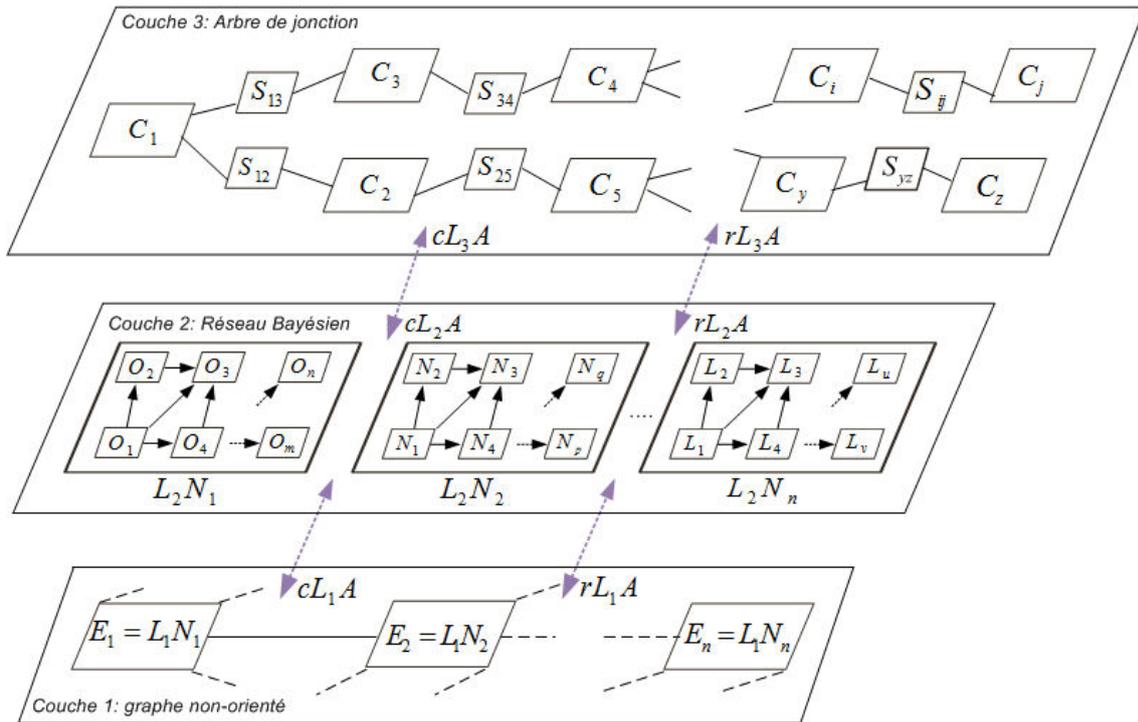


FIGURE 5.1 – Modèle générique d'un réseau de télécommunication.

Le principe de modélisation utilisé est le suivant : modéliser le comportement d'un réseau de télécommunication revient à modéliser le comportement de chacun de ses composants. Un réseau de n composants est modélisé avec $2n$ DAGs interconnectés. Le composant i du réseau est modélisé avec deux DAGs. Un DAG à la couche 2 noté L_2N_i qui modélise la propagation locale de pannes sur ce composant. Un DAG à la couche 1 noté L_1N_i qui modélise la propagation distribuée de pannes entre ce composant et les composants auxquels il est connecté. La Figure 5.1 représente graphiquement cette approche de modélisation qui décompose la propagation de pannes sur un réseau de grande dimension en plusieurs parties dépendantes. Cette décomposition apporte deux propriétés fondamentales au modèle générique qui facilitent sa reconfiguration : la modularité et l'extensibilité. Sur la Figure 5.1, les DAGs L_1N_i symbolisent les composants E_i du réseau et sont appelés noeuds de la couche 1. Les variables O_m , N_p et L_u sont respectivement les noeuds des DAGs L_2N_1 , L_2N_2 et L_2N_n de la couche 2. Ces variables sont appelées noeuds de la couche 2. Les noeuds C_y , C_z et S_{yz} de la couche 3 sont des cliques et des séparateurs de cliques adjacentes de l'arbre de jonction équivalent à la combinaison des couches 1 et 2. Les rôles des agents de calcul cL_1A , cL_2A , cL_3A et des agents de reconfiguration rL_1A , rL_2A , rL_3A seront discutés un peu plus tard dans ce chapitre.

Afin de prendre en compte, le caractère non déterministe du phénomène de propagation de pannes, les graphes de dépendances des couches 1 et 2 peuvent être transformés en graphes de dépendances probabilistes communément appelés réseaux bayésiens ou réseaux de croyances. Il s'agit de quantifier au moyen d'une distribution de probabilités conditionnelles, la force de dépendances que certains noeuds d'un DAG exercent sur leur successeur commun.

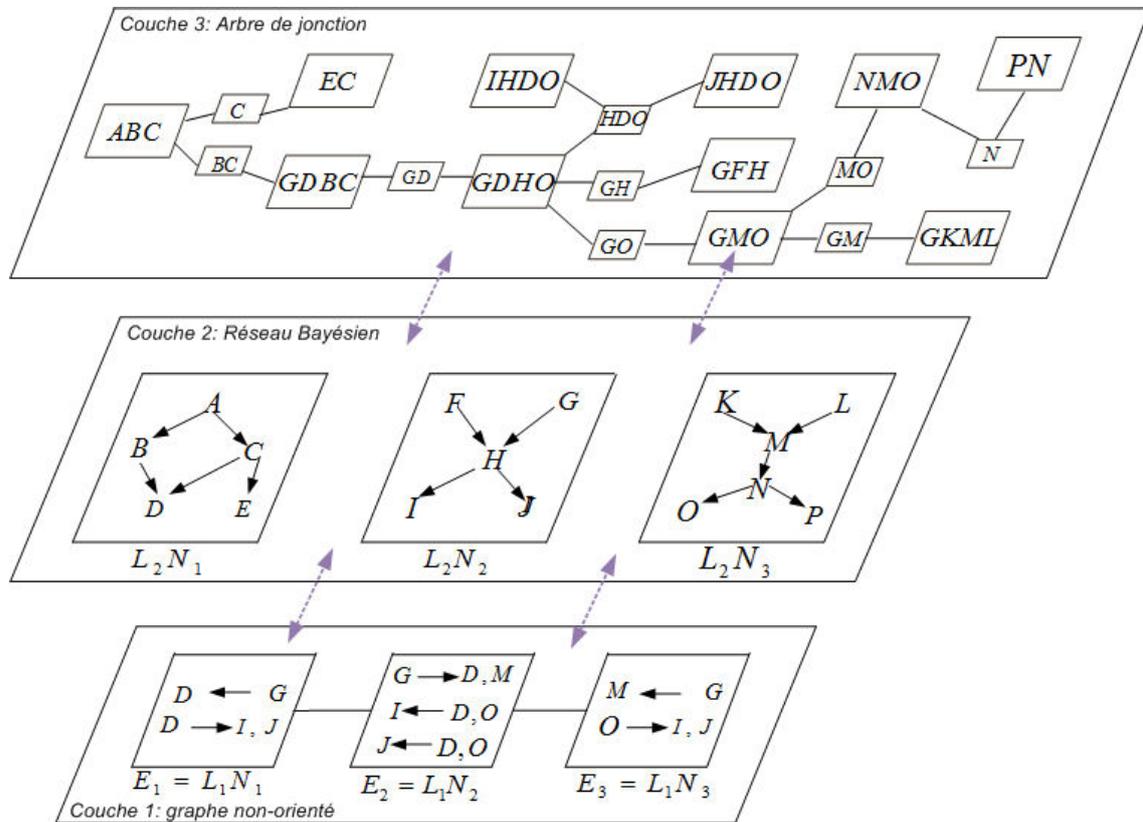


FIGURE 5.2 – Un exemple illustratif du modèle générique.

Ainsi, pour modéliser le comportement d'un réseau de télécommunication de grande échelle, on peut débuter par modéliser le comportement de quelques segments pour lesquels les connaissances "expertes" sur leurs fonctionnements sont déjà très bien acquises. Ensuite, au fur et à mesure que de nouvelles connaissances "expertes" sur le fonctionnement des autres segments sont acquises, le modèle initial peut facilement être étendu grâce aux propriétés de modularité et d'extensibilité. De même, un ou plusieurs segments du réseau peuvent facilement être retirés du modèle si l'opérateur le souhaite pour une quelconque raison. La Figure 5.2 présente un exemple simple illustratif du modèle générique. Il pourrait s'agir par exemple du modèle d'un segment de réseau de trois composants connectés selon une topologie en bus. La propagation de pannes sur ce petit réseau est modélisée par six DAGs interconnectés.

5.3.2 Première couche

La couche 1 modélise la topologie du réseau de télécommunication et la propagation distribuée de pannes entre les composants connectés. C'est un graphe non orienté dont les noeuds symbolisent les composants du réseau et les arêtes représentent les liaisons bidirectionnelles entre eux (voir la Figure 5.1). Chaque noeud de cette couche est un DAG L_1N_i qui symbolise le composant E_i et qui contient les dépendances distribuées entre

le composant E_i est ses voisins auxquels il est connecté. Sur la Figure 5.1, la couche 1 représente la topologie d'un réseau de n composants E_1, \dots, E_n respectivement symbolisés par les DAGs L_1N_1, \dots, L_1N_n . Par exemple, sur la Figure 5.2, les composants E_1 et E_3 sont connectés au composant E_2 . Les DAGs L_1N_1 et L_1N_3 représentent respectivement les composants E_1 et E_3 et contiennent respectivement les dépendances distribuées entre les composants E_1 et E_2 et entre les composants E_2 et E_3 . Le DAG L_1N_2 symbolise le composant E_2 et contient les dépendances distribuées entre E_2 et ses voisins E_1 et E_3 .

5.3.3 Deuxième couche

La couche 2 est un ensemble de DAGs. Chaque DAG L_2N_i modélise la propagation locale de pannes sur le composant E_i du réseau. Les DAGs L_2N_i et L_2N_j qui modélisent respectivement la propagation locale de pannes sur les composants E_i et E_j connectés sont reliés par les dépendances distribuées de la couche 1 qui modélisent la propagation distribuée de pannes entre E_i et E_j . Par exemple, sur la Figure 5.2, les graphes L_2N_1 et L_2N_2 contiennent respectivement les dépendances locales aux composants E_1 et E_2 . Ces deux DAGs sont interconnectés par les dépendances distribuées du noeud L_1N_1 de la couche 1. De même, les DAGs L_2N_2 et L_2N_3 de la couche 2 sont reliés par le DAG L_1N_3 de la couche 1.

Il faut remarquer que l'union des DAGs de la couche 2 et ceux de la couche 1 donne un large graphe que nous notons G qui modélise la propagation de pannes dans le réseau tout entier de n composants. Cependant, la séparation entre la propagation locale de pannes sur un composant et la propagation distribuée de pannes entre composants connectés, décompose G en $2n$ sous graphes interconnectés.

$$G = \bigcup_{i=1}^n L_2N_i \cup L_1N_i \quad (5.1)$$

Ainsi la construction des couches 1 et 2 implémente cette décomposition qui apporte les propriétés de modularité et d'extensibilité nécessaires à la reconfiguration automatique du modèle générique. Notons que l'Équation (5.1) ne garantit pas que le graphe G soit acyclique, ce qui veut dire que les DAGs L_2N_i et L_1N_i pour $i \in \{1, \dots, n\}$ doivent être construits de telle sorte que l'union G des $2n$ DAGs soit aussi un DAG.

5.3.4 Troisième couche

La nature distribuée d'un réseau de télécommunication introduit des dépendances mutuelles entre les composants connectés. Ce qui signifie que certaines dépendances relatives à la propagation distribuée de pannes entre composants connectés peuvent introduire des boucles acycliques dans le graphe de dépendances orientées G défini par l'Équation (5.1). Des boucles peuvent également apparaître dans la propagation locale de pannes sur un composant du réseau. Ces boucles locales sont principalement dues au fait que les couches adjacentes du modèle OSI (Open Systems Interconnection) d'ISO (International Standard Organization) se rendent mutuellement service.

La présence des boucles locales ou distribuées dans la propagation de pannes complexifie la structure du graphe G qui devient un réseau graphique multi-connecté. C'est pourquoi nous avons choisi de construire une troisième couche qui est une représentation graphique simplifiée et équivalente de la combinaison des deux premières couches. Cette représentation est un arbre de jonction sur lequel l'inférence peut être faite sans

se soucier de la présence ou non des boucles au niveau des couches 1 et 2. Nous rap- pelons que la construction d'un arbre de jonction et l'algorithme d'inférence sur cette structure ont été étudiés au Chapitre 3. Les cliques et les séparateurs de cliques adjacentes de l'arbre de jonction sont les noeuds de la couche 3. Comme tout arbre de jonction, la couche 3 vérifie la propriété d'intersection courante qui assure que l'intersection $C_i \cap C_j$ est un sous-ensemble de chaque clique et séparateur sur le chemin entre les cliques C_i et C_j . Par exemple, sur la Figure 5.2, l'intersection $G = \{\text{GDBC}\} \cap \{\text{GKML}\}$ appartient aux cliques $\{\text{GDHO}\}$, $\{\text{CGMO}\}$ et aux séparateurs $\{\text{SGD}\}$, $\{\text{SGO}\}$, $\{\text{SGM}\}$ qui forment le chemin entre les cliques $\{\text{GDBC}\}$ et $\{\text{GKML}\}$.

5.4 Formalisme du modèle générique

Soient L_2N_i et $L_2N_{j \neq i}$ deux graphes de dépendances distincts de la couche 2. Une dépendance locale au graphe L_2N_i est un tuple (u, v) tel que u et v appartiennent à l'ensemble des noeuds de L_2N_i . Une dépendance distribuée entre les graphes L_2N_i et $L_2N_{j \neq i}$ est tuple (u, v) tel que u appartient à l'ensemble des noeuds du graphe L_2N_i et v appartient à l'ensemble des noeuds du graphe $L_2N_{j \neq i}$. Par exemple, sur la Figure 5.2, (B, D) est une dépendance qui fait partir du modèle de propagation locale de pannes sur le compo- sant E_1 . (G, D) est une dépendance distribuée entre les composants connectés E_1 et E_2 .

La couche 1 que nous notons L_1Graph est un graphe non orienté dont les noeuds sont des graphes orientés, $L_1Graph = (L_1Nodes, L_1Edges)$. L_1Nodes est l'ensemble des n noeuds de la couche 1 qui symbolisent les composants du réseau modélisé.

$$L_1Nodes = \{L_1N_1, L_1N_2, \dots, L_1N_n\} \quad (5.2)$$

L_1Edges est l'ensemble des liens bidirectionnels entre les composants. Chaque noeud L_1N_i est défini comme suit : $L_1N_i = (L_1V_i, L_1E_i)$ tel que L_1V_i est le sous-ensemble de noeuds de la couche 2 appartenant aussi au graphe orienté L_1N_i . Chaque noeud de la couche 2 appartenant à L_1V_i fait partie d'au moins une dépendance distribuée de l'en- semble L_1E_i des dépendances distribuées entre le composant E_i et ses $k_i \geq 1$ voisins aux- quels il est connecté, L_1V_i est défini comme suit :

$$L_1V_i = N_i \cup \left(\bigcup_{j=1}^{k_i} N_j^i \right) \quad (5.3)$$

Dans l'Équation (5.3), $N_i \subset L_2V_i$, $N_j^i \subset L_2V_j$, $j \in \{1, \dots, n\} \setminus \{i\}$ tel que $\forall u \in N_i, \exists v \in N_j^i | u \in \text{parent}(v) \vee v \in \text{parent}(u)$. L_2V_i désigne le sous-ensemble de noeuds de la couche 2 ap- appartenant au graphe orienté L_2N_i de la couche 2 qui modélise la propagation locale de pannes sur le composant E_i (voir la Figure 5.1). Nous notons $\text{parent}(x)$ l'ensemble des parents d'un noeud x de la couche 2 et nous supposons dans l'Équation (5.3) que le noeud L_1N_i de la couche 1 (qui symbolise le composant E_i) est connecté à k_i composants voi- sins. Par exemple, sur la Figure 5.2, $L_1N_2 = (L_1V_2, L_1E_2)$ tel que $L_1V_2 = N_2 \cup N_1^2 \cup N_3^2$ avec $N_2 = \{G, I, J\}$, $N_1^2 = \{D\}$ et $N_3^2 = \{M, O\}$. $L_1E_2 = \{(G, D), (G, M), (D, I), (O, I), (D, J), (O, J)\}$ est l'en- semble des dépendances distribuées entre le composant E_2 ou L_1N_2 et ses voisins L_1N_1 et L_1N_3 auxquels il est connecté.

La couche 2 est un ensemble de n DAGs. Le DAG $L_2N_i = (L_2V_i, L_2E_i)$ modélise la pro- propagation locale de pannes sur le composant E_i ou L_1N_i . L_2E_i désigne l'ensemble des dé- dépendances locales au graphe L_2N_i de la couche 2. La construction de l'arbre de jonction qui constitue la couche 3 a été étudiée dans le Chapitre 3.

5.5 Calcul de diagnostic du modèle générique

Comme nous l'avons déjà expliqué, la construction des couches 1 et 2 apportent des propriétés de reconfiguration automatique au modèle générique. La couche 3 quant à elle est essentiellement utile pour la prise de décision du modèle générique puisque l'inférence peut être faite sur cette couche sans se soucier de la complexité relative à la topologie et au comportement du réseau de télécommunication modélisés par les couches 1 et 2. Les calculs de diagnostics effectués par le modèle sont basés sur l'algorithme d'inférence sur l'arbre de jonction qui constitue la couche 3. Cet algorithme d'inférence a été étudié au Chapitre 3. Nous ne montrerons donc ici que son utilisation par les agents de calcul du modèle générique. En effet, les opérations de mise à jour des potentiels des noeuds de la couche 3 (cliques et séparateurs de cliques adjacentes) sont réalisées par l'agent de calcul de la couche 3 que nous avons appelé cL_3A (Computing Layer 3 Agent) sur la Figure 5.1. Le cL_3A actualise le potentiel de chaque noeud de la couche 3 lorsqu'il reçoit une évidence de son équivalent de la couche 2 appelé cL_2A . Une évidence est une configuration observée de certains noeuds ou variables de la couche 2. Lorsque le cL_2A reçoit les potentiels actualisés des noeuds de la couche 3, il calcule la croyance ou distribution de probabilités marginales de chaque variable de la couche 2 conditionnelle à l'évidence observée. Ces deux agents peuvent communiquer en utilisant un simple mécanisme de mémoire partagée à travers l'interface entre la couche 2 et la couche 3 appelée "Interface L2-L3" (voir la Figure 5.3).

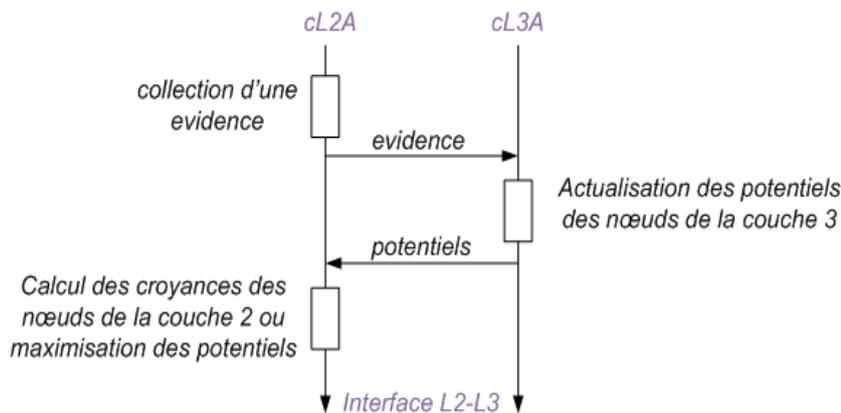


FIGURE 5.3 – Communication entre les couches 2 et la couche 3 pour le calcul des croyances des noeuds de la couche 2.

Comme nous l'avons vu dans le Chapitre 3, pour un quelconque noeud X de la couche 2 appartenant à une clique C de la couche 3, la croyance $\mathbb{P}(X|e)$ consistante avec l'évidence e se calcule en marginalisant sur le potentiel actualisé ϕ_C^* de la clique C .

$$\mathbb{P}(X|e) = \frac{\sum_{C \ni X} \phi_C^*}{\sum_C \phi_C^*} \quad (5.4)$$

Nous rappelons que lorsque l'inférence se termine à la couche 3, c'est-à-dire que tous les potentiels des cliques et séparateurs de cliques adjacentes sont actualisés, la propriété d'intersection courante vérifiée par la couche 3 garantit la cohérence des potentiels actualisés. Ce qui veut dire que la croyance de la variable X est la même, quelle que soit la clique C contenant X sur laquelle l'opération de marginalisation (5.4) est faite. Dans le contexte du diagnostic, on s'intéresse particulièrement à l'état le plus probable de la variable X consistante avec l'évidence e . Cet état est celui ayant la plus grande probabilité

dans la distribution a posteriori $\mathbb{P}(X|e)$. Notons aussi qu'il existe une seconde approche permettant de trouver les états les plus probables des noeuds de la couche 2 sans appliquer des opérations de marginalisation sur les potentiels actualisés. Avec cette approche, le diagnostic r^* est calculé directement à partir de l'explication la plus plausible w^* des observations e .

$$w^* = \bigcup_{C \in \{Couche3\}} w_C^* \text{ tel que } \phi_C(w_C^*) = \max_{w_C} \phi_C(w_C) \quad (5.5)$$

Dans l'Équation (5.5), w_C est une configuration des variables de la couche 2 appartenant à la clique C et w_C^* est la configuration qui maximise le potentiel ϕ_C de C . Le diagnostic r^* est l'état le plus probable des noeuds de la couche 2 n'ayant pas de prédécesseur (causes racines) consistant avec les observations e . Si nous notons i^* l'état le plus probable des noeuds n'étant pas à la racine de la couche 2, le diagnostic se définit par l'Équation (5.6). Il s'agit du complémentaire de i^* dans w^* .

$$r^* = w^* \setminus i^* \quad (5.6)$$

5.6 Capacité de reconfiguration du modèle générique

Dans cette section, nous montrons les capacités du modèle générique à facilement se reconfigurer automatiquement pour suivre d'éventuels changements sur la topologie et le comportement (propagation de pannes) du réseau de télécommunication. Les changements sur la topologie concernent essentiellement l'ajout d'un nouveau composant ou la suppression d'un composant existant du réseau. Les changements relatifs au comportement du réseau sont principalement la mise à jour de lois conditionnelles de certains noeuds de la couche 2 suite à l'ajout, la suppression d'une variable de la couche 2, l'ajout d'une nouvelle dépendance entre deux noeuds de la couche 2 ou la suppression d'une dépendance existante.

5.6.1 Les agents de reconfiguration

Sur la Figure 5.1, chacune des trois couches du modèle générique a un agent de reconfiguration qui peut communiquer par un simple mécanisme de partage de la mémoire avec ses homologues des couches adjacentes. L'agent de reconfiguration rL_1A de la couche 1 est système apprenant qui traque les changements liés à la topologie du réseau ainsi que les dépendances distribuées entre les composants. Le rL_1A communique avec son homologue de la couche 2 appelé rL_2A à travers l'interface L1-L2 entre les couches 1 et 2. Le rL_2A est également un système apprenant qui traque les dépendances locales aux composants du réseau et les éventuels changements sur les lois conditionnelles de certains noeuds de la couche 2. L'agent de reconfiguration de la couche 3 rL_3A est sollicité par le rL_2A pour reconstruire l'arbre de jonction qui forme cette couche ou tout simplement mettre à jour les potentiels initiaux des noeuds de cette couche. La communication entre rL_2A et rL_3A se fait via l'interface L2-L3 entre les couches 2 et 3.

5.6.2 Reconfiguration initiée par la couche 2

Le rL_2A est un système d'apprentissage artificiel qui implémente un algorithme d'extraction de données et de découverte de connaissances à partir de l'énorme quantité de

données que génèrent les composants du réseau. Cet agent doit être capable de découvrir de nouvelles dépendances locales à un composant du réseau, de nouveaux états de certaines variables de la couche 2, des connaissances statistiques qui exigent la réestimation des distributions de probabilités conditionnelles de certains noeuds de la couche 2. Par exemple, les lois conditionnelles peuvent être recalculées lorsque la vraisemblance des données générées par le réseau (données d'apprentissage) sous le modèle existant à l'instant t devient supérieure à la vraisemblance à l'instant $t - \delta t$. Les lois conditionnelles peuvent aussi être recalculées si par exemple le score du modèle à l'instant t est meilleur que celui à l'instant $t - \delta t$. Dans ce dernier cas, le choix de la fonction de score est laissé libre selon l'implémentation souhaitée. Lorsque le rL_2A apprend de nouvelles connaissances pertinentes sur le comportement du réseau, il reconfigure la couche 2 du modèle et envoie une commande de reconfiguration (*reconf cmd*) à ses homologues des couches 1 et 3. La Figure 5.4 présente graphiquement le mécanisme de reconfiguration initié à la couche 2. Le message *reconf done* est un acquittement à la commande *reconf cmd*.

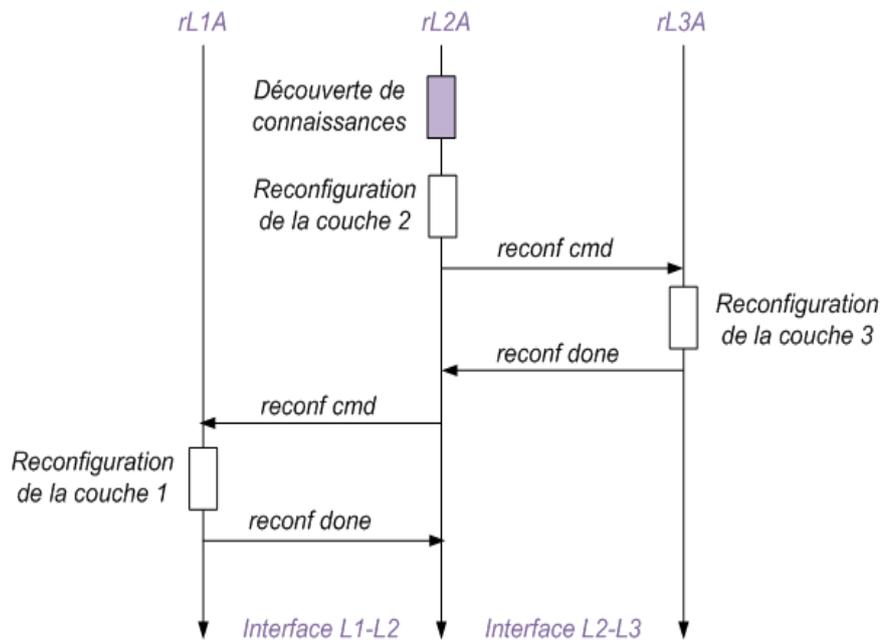


FIGURE 5.4 – Processus de reconfiguration initié à la couche 2.

Notons que lorsque le processus de reconfiguration est initié par le rL_2A , celui-ci n'envoie une commande de reconfiguration au rL_1A que si la nouvelle connaissance apprise est une dépendance distribuée. En effet, les dépendances locales ne sont pas connues à la couche 1. Notons aussi que c'est le type de la connaissance apprise au niveau 2 qui permet au rL_3A , après avoir reçu la commande de reconfiguration envoyée par le rL_2A , de savoir s'il doit reconfigurer ou reconstruire entièrement l'arbre de jonction qui forme la couche 3. Par exemple, si la loi conditionnelle d'un noeud de la couche 2 change, le rL_3A n'a besoin que de déclencher un évènement notifiant le cL_3A de recalculer les paramètres concernés de l'arbre de jonction, c'est-à-dire recalculer le potentiel initial de chacune des cliques dont ce noeud de la couche 2 appartient. Par contre, si de nouvelles dépendances locales ou distribuées sont apprises, le rL_3A doit reconstruire la couche 3.

La Figure 5.5 spécifie le comportement dynamique du rL_2A au moyen d'une machine à trois états. Initialement, le rL_2A est à l'état *Data mining*, c'est-à-dire qu'il cherche de nouvelles connaissances relatives à la propagation de pannes sur le réseau à partir des

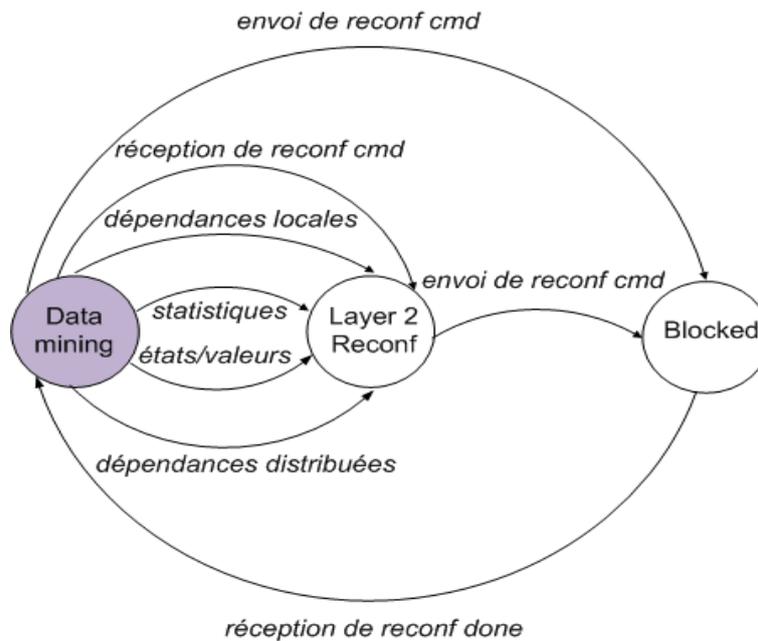


FIGURE 5.5 – Machine à états de l’agent de reconfiguration de la couche 2.

données générées par les équipements. Lorsqu’une nouvelle connaissance est apprise, le rL_2A transite à l’état *Layer 2 Reconf* dans lequel il intègre cette nouvelle connaissance dans la couche 2 du modèle. Le rL_2A déclenche ensuite la reconfiguration des couches adjacentes en envoyant une commande aux agents de reconfiguration de ces couches. L’état *Blocked* implémente la synchronisation entre deux processus de reconfiguration successifs, c’est-à-dire que le rL_2A attend que le processus de reconfiguration courant se termine avant de déclencher éventuellement un autre. Ainsi, la complétude et la consistance du modèle restent garanties après l’intégration d’une nouvelle connaissance apprise dans le modèle.

5.6.3 Reconfiguration initiée par la couche 1

La fonction du rL_1A est principalement de traquer les éventuels changements sur la topologie du réseau tels que l’ajout ou la suppression d’un composant. Le rL_1A peut être un système de découverte automatique de la topologie d’un réseau [BEACON \[2000\]](#) [RAMANATHAN et collab. \[2000\]](#) ou une interface d’administration permettant aux experts du réseau modélisé de spécifier et d’ajouter un nouveau composant dans le modèle ou de supprimer un composant du modèle. La spécification d’un nouveau composant inclut : le nom ou l’identifiant de ce composant, les composants voisins déjà existants dans le modèle auxquels il est connecté, les dépendances locales (le modèle de propagation locale de pannes sur ce composant) et les dépendances distribuées entre ce composant et chaque composant voisin.

Comme on peut le remarquer sur la Figure 5.6, tout changement sur la topologie du réseau ou relatif à la propagation distribuée de pannes entre plusieurs composants entraîne la reconfiguration des trois couches du modèle. La Figure 5.6 présente ce processus de reconfiguration initié par le rL_1A . Le comportement dynamique du rL_1A est modélisé par une machine à trois états présentée par la Figure 5.7. Initialement, le modèle est dans un état d’équilibre, c’est-à-dire qu’aucun processus de reconfiguration n’est en cours. Le

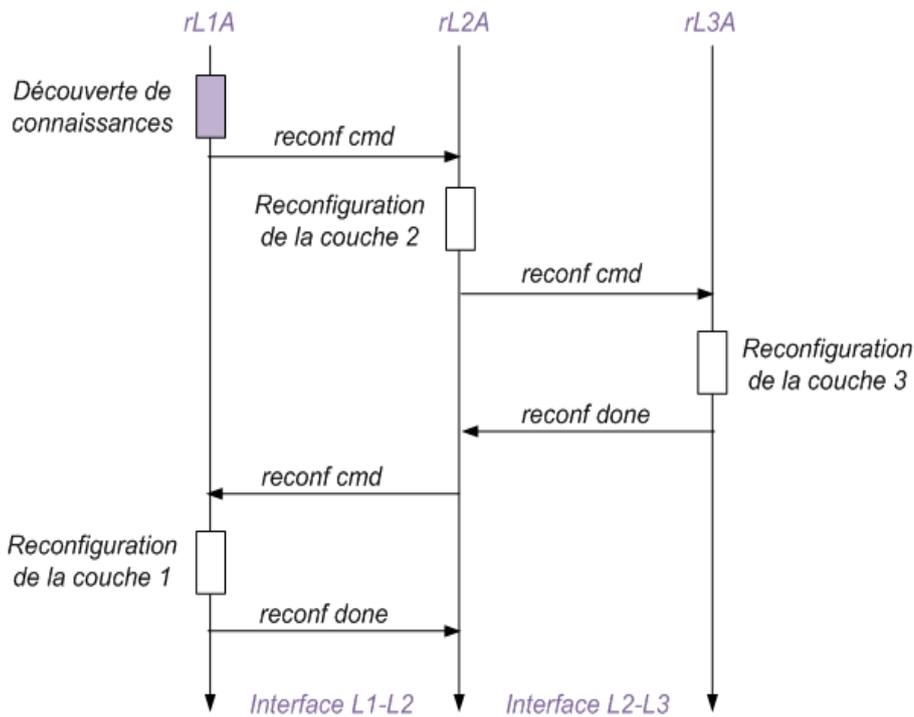


FIGURE 5.6 – Processus de reconfiguration initié à la couche 1.

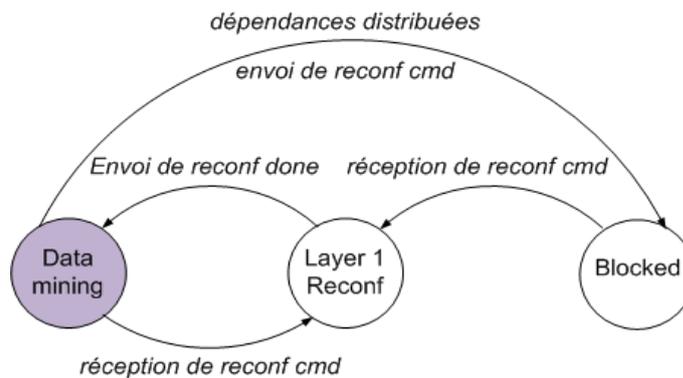


FIGURE 5.7 – Machine à états de l’agent de reconfiguration de la couche 1.

r_{L_1A} est de ce fait dans l’état *Data mining*, c’est-à-dire qu’il est attente d’une spécification d’un nouveau composant à ajouter, il cherche dans les données du réseau, de nouvelles connaissances relatives à la topologie ou à la propagation distribuée de pannes entre les composants. Lorsque le r_{L_1A} acquiert de nouvelles connaissances concernant la topologie du réseau ou la propagation distribuée de pannes sur le réseau, il ne reconfigure pas immédiatement la couche 1. Il envoie un message *reconf cmd* au r_{L_2A} et transite à l’état d’attente *Blocked*. Dans l’état *Blocked*, le r_{L_1A} reste bloqué jusqu’à ce qu’il reçoit le message *reconf done* en provenance du r_{L_2A} lui notifiant la terminaison des processus de reconfiguration des couches supérieures. Le r_{L_1A} transite ensuite à l’état *Layer 1 Reconf* dans lequel il reconfigure la couche 1. Dans le cas où le r_{L_2A} apprend une nouvelle dépendance distribuée, il demande la reconfiguration de la couche 1 en envoyant le message *reconf cmd* au r_{L_1A} qui transite de l’état *Data mining* à l’état *Layer 1 Reconf*, puis retransite à l’état initial *Data mining* lorsqu’il termine la reconfiguration de la couche 1. Le modèle retrouve donc un nouvel état d’équilibre.

Supposons par exemple qu'un nouveau composant spécifié doit être ajouté dans le modèle du réseau. La spécification de ce nouveau composant donne les dépendances locales à ce composant et les dépendances distribuées entre ce composant et les composants déjà existants dans le modèle auxquels il est connecté. La reconfiguration du modèle est dans ce cas initiée par le rL_1A qui envoie un message *reconfcmd* au rL_2A , attend que les couches 2 et 3 soient reconfigurées et reconfigure la couche 1. La reconfiguration de la couche 1 consistera à ajouter un nouveau noeud à la couche 1 qui symbolise le nouveau composant. Ce nouveau noeud est un DAG qui contient les dépendances distribuées avec les voisins du nouveau composant. Ensuite le rL_1A doit mettre à jour les noeuds de la couche 1 (représentés par des DAGs) qui symbolisent les composants voisins. Cette mise à jour consiste essentiellement à ajouter dans chacun de ces DAGs, les dépendances distribuées entre le nouveau composant et le composant symbolisé par le DAG courant. La reconfiguration de la couche 2 consistera à construire le DAG de la couche 2 qui contient les dépendances locales au nouveau composant. Ensuite le rL_2A notifiera le cL_2A de calculer les lois conditionnelles des nouvelles variables de la couche 2 (si ces lois n'ont pas été données dans la spécification du nouveau composant) et recalculer celles des anciens noeuds de la couche 2 dont l'ensemble des prédécesseurs a été modifié. La reconfiguration de la couche 3 consistera à reconstruire l'arbre de jonction et recalculer les potentiels initiaux des cliques.

Il faut remarquer que seul le rL_2A prend une décision de reconfiguration après l'apprentissage d'une nouvelle connaissance. Les autres agents de reconfiguration (rL_1A et rL_3A) attendent une commande *reconfcmd* en provenance du rL_2A avant de reconfigurer leur couche respective. Par conséquent, le rL_2A est le coordonnateur du système multi agents ainsi conçu.

5.7 Conclusion et perspectives

Nous avons présenté dans ce chapitre un modèle générique en trois couches basé sur les réseaux bayésiens pour le diagnostic automatique des réseaux de télécommunications. Ce modèle propose une séparation entre la propagation locale de pannes sur un composant du réseau et la propagation distribuée de pannes entre les composants connectés. Cette séparation permet également de dissocier la modélisation de la topologie du réseau de celle de son comportement ou propagation de pannes apportant ainsi au modèle les propriétés de modularité et d'extensibilité qui facilitent la reconfiguration. Le modèle générique a la capacité de prendre des décisions de diagnostic basées sur un raisonnement probabiliste et de se reconfigurer automatiquement lorsque de nouvelles connaissances relatives à la topologie et au comportement du réseau de télécommunication sont acquises. La reconfiguration du modèle est basée sur un schéma de communication entre les agents de reconfiguration des différentes couches.

Deux principales perspectives peuvent être envisagées. Comme les couches 1 et 2, il est envisageable de décomposer également l'arbre de jonction qui constitue la couche 3. On pourra ainsi avoir, pour chaque composant du réseau de télécommunication, un arbre de jonction équivalent au graphe de dépendances de la couche 2 qui modélise la propagation locale de pannes sur ce composant. Cette conception permettra de transformer le modèle générique en un système distribué avec par conséquent la possibilité de paralléliser et de distribuer l'inférence ou les calculs de diagnostic. Une autre perspective pourrait être de donner plus d'intelligence à l'agent de reconfiguration de la couche 3 en

développant un algorithme permettant de prendre en compte de nouvelles dépendances apprises au niveau des couches inférieures sans reconstruire entièrement la couche 3. Ceci permettra notamment d'augmenter le passage à l'échelle et l'efficacité de reconfiguration du modèle générique.

Dans le chapitre suivant, nous étudierons le réseau d'accès optique GPON-FTTH (Gigabit Passive Optical Network-Fiber To The Home) et nous montrerons comment le modèle générique peut être appliqué pour modéliser la topologie et le comportement de ce réseau d'accès pour des raisons de diagnostic.

Chapitre 6

Modélisation probabiliste du réseau FTTH (Fiber To The Home) de type GPON (Gigabit Passive Optical Network)

« Probability is not really about number; it is about the structure of reasoning. »

Glenn Shafer

Sommaire

6.1 Introduction	94
6.2 Modèle du réseau GPON-FTTH	94
6.2.1 Le réseau de transport	95
6.2.2 Le réseau de branchement	98
6.2.3 Le modèle initial d'un PON	99
6.3 Paramètres experts du modèle initial d'un PON	100
6.4 Validation du modèle initial d'un PON	104
6.4.1 Validation sur une plateforme expérimentale	104
6.4.2 Validation sur un réseau GPON-FTTH opérationnel	109
6.5 Comparaison entre système expert et modèle probabiliste de diagnostic	113
6.6 Discussion et conclusion	116

6.1 Introduction

Le développement d'une méthode de diagnostic d'un réseau de télécommunication à base d'un modèle de ce réseau se fait en deux étapes. La première étape construit une représentation explicite de la topologie et du comportement du réseau. La seconde étape consiste à développer ou appliquer un algorithme de raisonnement sur le modèle précédemment construit. Nous proposons dans la Section 2 de ce chapitre, un modèle probabiliste de la topologie et de la propagation de pannes et d'alarmes sur le réseau d'accès optique GPON-FTTH. Ce modèle, basé sur le formalisme d'un réseau bayésien, est une application de la méthode générique de modélisation d'un réseau de télécommunication développée dans le Chapitre 5. La Section 3 détermine approximativement les paramètres du modèle du réseau GPON-FTTH proposé, sur la base des connaissances métiers dites "expertes" acquises à partir de la norme ITU-T G.984 de ce réseau. Le modèle probabiliste de diagnostic ainsi construit est doublement validé dans la Section 4. Une première validation expérimentale avec des cas simples de diagnostic provoqués sur un réseau en laboratoire suivie d'une seconde validation avec des cas réels issus du réseau GPON-FTTH opérationnel de l'opérateur Orange est faite. Dans la Section 5, les résultats de diagnostic effectués avec le modèle probabiliste sont comparés avec ceux du système de diagnostic à base de règles "expertes" actuellement en exploitation dans le système de gestion du réseau GPON-FTTH d'Orange. Ce chapitre se termine par une discussion et une conclusion faites dans la Section 6.

6.2 Modèle du réseau GPON-FTTH

Le réseau GPON-FTTH est constitué de trois blocs fonctionnels : le composant OLT (Optical Line Termination), le composant ONT (Optical Network Terminal) et le réseau de distribution optique ODN (Optical Distribution Network). L'OLT est un équipement actif situé au central de l'opérateur. L'ONT est un équipement actif placé chez l'abonné, sur lequel se connecte une passerelle résidentielle RG (Residential Gateway). Un OLT est un châssis équipé d'un maximum de 16 cartes électroniques. Chaque carte pouvant disposer de 8 ou 16 ports selon le fabricant et la génération de l'OLT. Un port d'un OLT est connecté à un PON (Passive Optical Network), voir la Figure 6.1. Il s'agit d'un réseau arborescent permettant de connecter un maximum de 64 ou 128 ONTs sur un port d'un OLT.

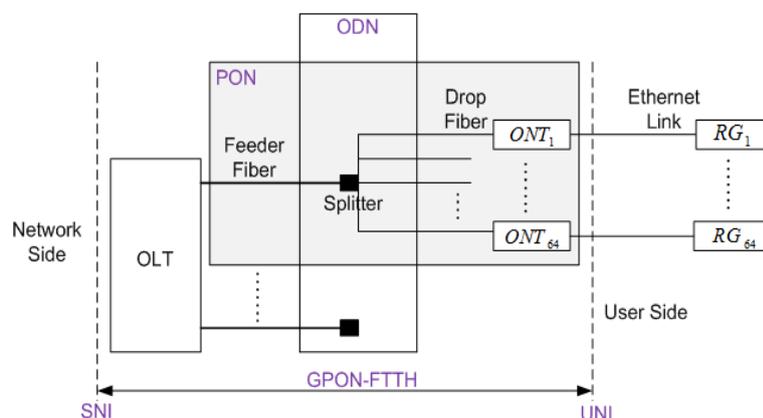


FIGURE 6.1 – Architecture du réseau GPON-FTTH.

La connexion sur un port d'un OLT, de plusieurs ONTs situés chez les abonnés, se fait au travers de l'architecture en arbre du réseau de distribution ODN (Optical Distribution

Network) qui constitue le troisième bloc fonctionnel. Puisque tous les PONs d'un réseau GPON-FTTH ont le même comportement et qu'il n'y a pas d'interaction entre les PONs, nous avons fait le choix de modéliser un seul PON sachant que ce modèle peut être répliqué à tous les PONs d'un réseau GPON-FTTH. Par ailleurs, compte tenu de la diversité des ingénieries du réseau de distribution qui complexifie la modélisation du composant ODN, nous nous sommes proposé de modéliser dans un premier temps le composant OLT et les ONTs d'un même arbre PON. Le modèle de ces deux composants est appelé le modèle initial du réseau GPON-FTTH. Le modèle de l'ODN sera développé plus tard dans cette thèse (dans le Chapitre 8) et rajouté facilement au modèle initial grâce aux propriétés de modularité et d'extensibilité de la méthode générique de modélisation d'un réseau que nous appliquons.

6.2.1 Le réseau de transport

Nous présentons dans cette sous-section un modèle de propagation locale de pannes et d'alarmes sur le composant OLT, ainsi que la propagation distribuée de pannes entre l'OLT et les ONTs du même arbre PON qui lui sont connectés. Conformément au modèle générique, le composant OLT est modélisé par deux graphes de dépendances acycliques. La Figure 6.2.a présente ces deux graphes de dépendances. Il s'agit d'un graphe de la couche 2 qui modélise la propagation locale de pannes et d'alarmes sur le composant OLT, un graphe de la couche 1 qui modélise la propagation distribuée de pannes et d'alarmes entre les composants OLT et ONT. Ces graphes présentent trois types de variables : les pannes ou causes racines qui sont surlignées, les causes intermédiaires et les faits observés. Une cause est dite intermédiaire si la variable qui lui est associée admet au moins un prédécesseur et au moins un successeur dans le modèle de propagation de pannes et d'alarmes. Il faut noter que certaines variables du modèle sont des vecteurs afin de prendre en compte la topologie en arbre d'un PON. En effet, chacune de ces variables est un regroupement de plusieurs variables de même nom, mais propre à des ONTs différents. Par exemple, la variable $R_{x_{OLT}}[]$ est un regroupement des variables désignant les puissances reçues à l'OLT, des ONTs du même arbre PON. Ainsi la variable $R_{x_{OLT}}[i]$ désigne la puissance reçue à l'OLT de l'ONT numéro i sur l'arbre PON.

Les causes racines : elles ont été considérées selon les besoins de diagnostic de l'opérateur de télécommunications Orange. Nous les énumérons ci-dessous :

- La fibre optique de transport du PON représentée par la variable $Fiber_T$. Cette variable peut prendre trois états possibles. L'état *OK* signifie qu'il n'y a aucune anomalie de transmission sur la fibre optique. Les états *AT* et *BR* signifient respectivement que l'atténuation de la fibre est trop importante ou que la fibre est rompue.
- La température de l'OLT notée T_{OLT}^c , est une variable continue que nous avons discretisée en trois états T_0 , T_n et T_h , correspondants respectivement à un intervalle de très basses températures, de températures normales ou de températures très élevées du composant OLT.
- L'alimentation de l'OLT dénotée par la variable binaire Alt_{OLT} . L'alimentation peut être bonne, c'est-à-dire qu'elle délivre un courant de polarisation et une tension suffisants pour faire fonctionner l'OLT. Un courant de polarisation faible ou très important peut être dû à une alimentation défectueuse ou un vieillissement du laser.

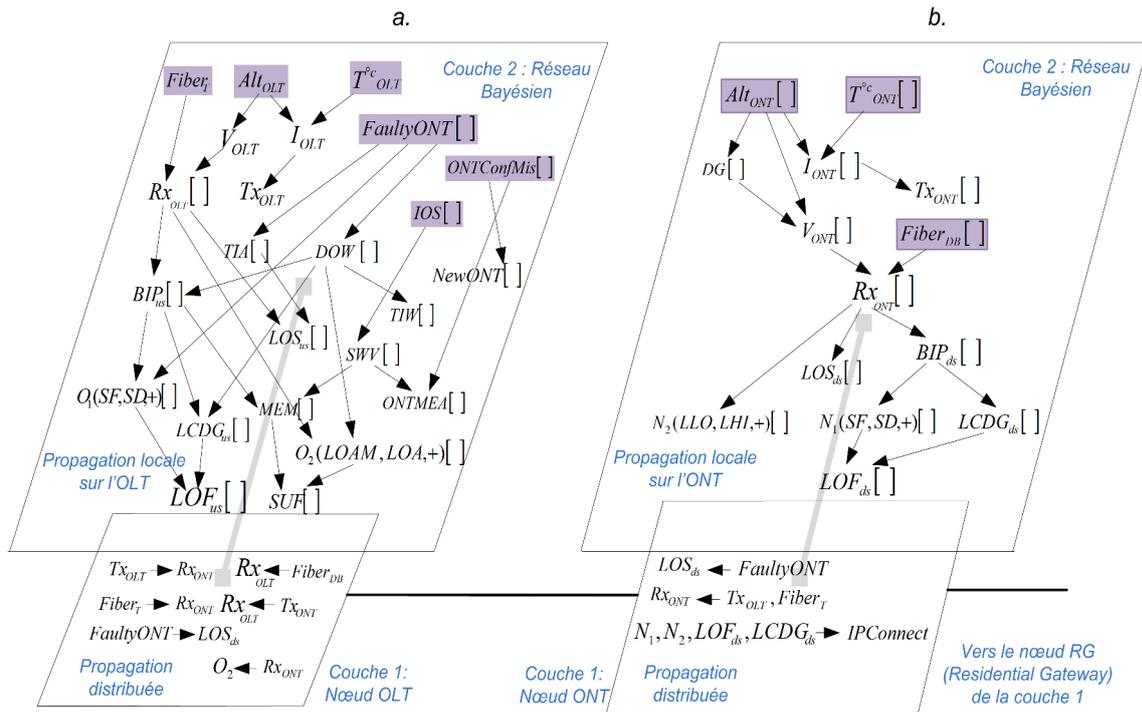


FIGURE 6.2 – Modèle initial d'un PON du réseau d'accès GPON-FTTH.

- La variable binaire $FaultyONT[i]$, avec $i \in \{1, \dots, 64\}$ pour un PON de 64 ONTs, symbolise l'état global de l'ONT numéro i sur ce PON, tel qu'il est connu de l'OLT. Un ONT est dit "rogue" ou défectueux lorsqu'il émet un signal dans un intervalle de temps qui ne lui a pas été alloué par l'OLT auquel il est connecté. Ce comportement hors du standard de fonctionnement nominal d'un ONT peut engendrer des conflits avec les signaux de données transmis par les autres ONTs sur le PON causant ainsi des interférences entre les signaux de données d'un quelconque ensemble d'ONTs, rendant donc le PON inutilisable. Nous notons qu'il s'agit bien d'une panne relative à un ONT, mais les alarmes causées par un ONT "rogue" sont déclenchées par l'OLT. De façon plus générale, un ONT est dit "faulty" lorsqu'il est défaillant pour une raison qui n'est pas forcément identifiée.
- Une incompatibilité entre le logiciel IOS (Image Operating System) d'un ONT et celui de l'OLT auquel il est connecté est un dysfonctionnement qui cause des échecs de (re)synchronisation entre les deux composants. Cette panne déclenche très souvent une alarme SWV[i] (SoftWare Version) pour l'ONT numéro i sur l'arbre PON.
- La variable binaire $ONTConfMis[i]$ (ONT Configuration Mismatch) dénote une erreur faite lors de la configuration nécessaire pour connecter logiquement l'OLT et l'ONT numéro i sur l'arbre PON.

Les causes intermédiaires et les alarmes : une cause est dite intermédiaire si la variable qui la représente dans le graphe de dépendances qui modélise la propagation de pannes admet au moins un prédécesseur et un successeur. Quant aux alarmes observées, elles peuvent être déclenchées aussi bien par les causes racines que par les causes intermédiaires. Nous énumérons ci-dessous, les causes intermédiaires et les alarmes tout en justifiant les dépendances entre les noeuds correspondants dans la Figure 6.2.a.

-
- Le courant de polarisation de l'OLT, noté I_{OLT} , est influencé par l'alimentation et la température de l'OLT d'où les dépendances locales $Alt_{OLT} \rightarrow I_{OLT}$ et $T_{OLT}^c \rightarrow I_{OLT}$. Nous avons discrétisé la variable I_{OLT} en trois états I_0 , I_l ou I_n correspondants respectivement à des intervalles dans lesquels le courant est très faible ou quasiment nul, faible ou nominal. La tension d'alimentation V_{OLT} de l'OLT dépend de l'état de son alimentation. On peut donc aussi considérer la dépendance $Alt_{OLT} \rightarrow V_{OLT}$.

 - La puissance Tx_{OLT} transmise par l'OLT est régulée par son courant de polarisation I_{OLT} . Cette corrélation est représentée par la dépendance $I_{OLT} \rightarrow Tx_{OLT}$ locale au composant OLT. La variable Tx_{OLT} peut prendre les états Tx_0 , Tx_l et Tx_n correspondants respectivement à des intervalles dans lesquels la puissance émise par l'OLT est très faible ou tend vers $-\infty$ en décibel-milliwatt (dBm), faible ou nominale.

 - Le niveau de puissance $Rx_{OLT}[i]$ de l'ONT numéro i reçu à l'OLT est une variable pouvant prendre les états Rx_0 , Rx_l et Rx_n correspondants respectivement à des intervalles dans lesquels la puissance reçue est très faible ou tend vers $-\infty$ en dBm, faible ou nominale. $Rx_{OLT}[i]$ dépend de la tension V_{OLT} et de l'état de la fibre de transport (feeder fiber en anglais) $Fiber_T$. Le niveau de puissance $Rx_{OLT}[i]$ dépend aussi de l'état de la fibre de branchement $Fiber_{DB}[i]$ de l'ONT numéro i et de la puissance $Tx_{ONT}[i]$ transmise par cet ONT. Ces corrélations sont modélisées par deux dépendances ($Fiber_T \rightarrow Rx_{OLT}$, $V_{OLT} \rightarrow Rx_{OLT}$) locales au composant OLT et deux autres dépendances ($Fiber_{DB} \rightarrow Rx_{OLT}$, $Tx_{ONT} \rightarrow Rx_{OLT}$) distribuées entre les composants OLT et ONT. Il faut remarquer que, conformément à la modélisation générique développée dans le Chapitre 5, les dépendances distribuées font partie des arêtes du graphe de dépendances de la couche 1 qui représente le composant OLT dans la topologie physique du réseau PON.

 - Le compteur $BIP_{us}[i]$ (Bit Interleaving Parity upstream) est un calcul de parité sur les trames (montantes dans ce cas, c'est-à-dire les trames émises par un ONT). L'évolution de ce compteur est corrélée avec la valeur du BER (Bit Error Rate) d'une transmission de données sur le canal montant, c'est-à-dire de l'ONT numéro i vers l'OLT. Une mauvaise réception à l'OLT, du signal optique transmit par l'ONT numéro i , peut causer des erreurs sur des bits de données d'où la dépendance locale $Rx_{OLT} \rightarrow BIP_{us}$. La variable $BIP_{us}[i]$ peut prendre trois états correspondants à des intervalles dans lesquels le nombre d'erreurs sur les bits de données est très faible, grand ou très important. Les erreurs de transmission sur le canal montant d'un PON impactent la qualité du signal reçu à l'OLT qui peut déclencher les alarmes relatives à la qualité du signal telle que SD (Signal Degraded), SF (Signal Fail), LCDG (Loss of GEM Channel Delineation) et MEM (Message Error Message).

 - Les alarmes SD et SF ne pouvant être observées simultanément, pour des raisons de simplicités, nous les avons considérées comme faisant partie des états d'une variable ternaire notée O_1 . Le troisième état noté + de cette variable correspond à une situation pour laquelle aucune des alarmes SD ou SF n'est observée. La relation entre ces alarmes et les erreurs de transmission est représentée par la dépendance $BIP_{us} \rightarrow O_1$ locale au composant OLT.

 - Une alarme $MEM[i]$ indique que l'OLT reçoit des messages de synchronisation in-

connus en provenance de l'ONT numéro i sur le PON. L'alarme $MEM[i]$ peut être due à des erreurs de transmission qui ont corrompu les entêtes des messages de synchronisation, d'où la dépendance $BIP_{us}[i] \rightarrow MEM[i]$. Une incompatibilité des versions des logiciels "firmware" de l'ONT numéro i et l'OLT peut également engendrer l'alarme $MEM[i]$. De ce fait, nous considérons aussi la dépendance locale $SWV[i] \rightarrow MEM[i]$. Notons aussi qu'une dégradation très importante du signal montant cause des erreurs de transmission et donc des trames de données erronées reçues à l'OLT qu'il supprime s'il ne parvient pas à corriger les erreurs et déclenche l'alarme $LOF[i]$. Cette relation est modélisée par la dépendance locale $O_1 \rightarrow LOF$.

- L'alarme $LCDG_{us}[i]$ (Loss of upstream GEM Channel Delineation) est déclenchée par l'OLT lorsqu'il ne parvient pas à séparer ou délimiter les trames de plusieurs services différents dans un flux multiplexé en provenance de l'ONT numéro i . Cet échec de "délinéation" peut causer la perte d'une ou de plusieurs trames dans le flux, d'où la dépendance $LCDG_{us} \rightarrow LOF_{us}$.
- L'alarme $LOS_{us}[i]$ (Loss Of upstream Signal) est déclenchée par l'OLT lorsqu'il ne reçoit aucun signal optique valide attendu, envoyé par l'ONT numéro i après quatre intervalles de temps non consécutifs qui lui ont été alloués par l'OLT. Ceci peut être dû à une mauvaise réception du signal optique par l'OLT, d'où la dépendance $Rx_{OLT}[i] \rightarrow LOS_{us}[i]$. L'alarme $LOS_{us}[i]$ peut aussi être due au fait que l'ONT numéro i n'a tout simplement pas transmis un signal parce qu'un ONT numéro $j \neq i$ a utilisé l'intervalle de temps alloué à l'ONT numéro i . Dans ce cas, l'OLT déclenche l'alarme $TIA[j]$ (Transmission Interference Alarm) pour signaler que l'ONT numéro j a utilisé un "time slot" qui ne lui était pas alloué. L'ONT numéro j est dit "rogue ONT" ou "faulty ONT". Un ONT "rogue" peut aussi provoquer le déclenchement de l'alarme $DOW[j]$ (Drift Of Window) au lieu de $TIA[j]$ lorsque l'ONT numéro j émet au delà du "time slot" qui lui est alloué. L'alarme LOS peut non seulement révéler une dégradation de la qualité du support (fibre atténuée ou rompue) mais aussi un dysfonctionnement majeur de l'ONT. Nous modélisons ces comportements par les dépendances $FaultyONT \rightarrow DOW$, $FaultyONT \rightarrow TIA$ et $TIA \rightarrow LOS_{us}$.
- Une alarme $DOW[i]$ affecte la qualité du signal reçu et transmit par l'ONT numéro i ainsi que celle de l'ONT numéro $j \neq i$ donc une partie du "time slot" a été utilisé par l'ONT numéro i . Ceci qui signifie qu'une alarme DOW peut aussi déclencher les alarmes relatives à la qualité du signal que sont SD et SF ($FaultyONT \rightarrow O_1$), causer un échec d'un processus de délinéation ($DOW \rightarrow LCDG_{us}$), déclencher les alarmes LOA (Loss Of Acknowledge) ou $LOAM$ (Loss Of Physical Layer OAM). Puisque les alarmes LOA et $LOAM$ ne peuvent pas être déclenchées simultanément, nous les avons considérées comme faisant partie des états d'une variable ternaire appelée O_2 (voir la Figure 6.2.a). Le troisième état noté $+$ de la variable O_2 correspond à une situation pour laquelle ni l'alarme LOA , ni l'alarme $LOAM$ n'est déclenchée. Enfin, la perte des messages $PLOAM$ entre l'OLT et un ONT peut entraîner l'échec de (re)synchronisation de cet ONT, d'où la dépendance $O_2 \rightarrow SUF$ (StartUp Failure).

6.2.2 Le réseau de branchement

La Figure 6.2.b présente le graphe de dépendances qui modélise la propagation locale de pannes et d'alarmes sur le composant ONT ainsi que le graphe de dépendances qui

modélise la propagation distribuée de pannes entre les composants OLT et ONT.

Les causes racines : l'alimentation $Alt_{\text{ONT}}[i]$, la température $T_{\text{ONT}}^c[i]$ et la fibre de branchement $Fiber_{\text{DB}}[i]$ de l'ONT numéro i sur le PON sont les trois causes racines du réseau de branchement des abonnés que nous avons considérés selon les besoins de diagnostic de l'opérateur Orange.

Les causes intermédiaires et les alarmes :

- La puissance $Rx_{\text{ONT}}[i]$ reçue à l'ONT numéro i dont le niveau peut provoquer le déclenchement des alarmes LLO (Level LOw) ou LHI (Level HIGH) selon qu'il soit trop faible ou trop élevé. Puisque les alarmes LLO et LHI ne peuvent pas être déclenchées simultanément, nous les avons considérées comme faisant partie des états d'une variable ternaire appelée N_2 (voir la Figure 6.2.b). Le troisième état noté + de la variable N_2 correspond à une situation pour laquelle ni l'alarme LLO, ni l'alarme LHI ne sont déclenchées. Le niveau de puissance $Rx_{\text{ONT}}[i]$ dépend de la tension d'alimentation $V_{\text{ONT}}[i]$, de l'état de la fibre de branchement $Fiber_{\text{DB}}[i]$ de l'ONT numéro i , de l'état de la fibre de transport $Fiber_{\text{T}}$ du PON et de la puissance Tx_{OLT} transmise par l'OLT. Nous notons deux dépendances locales au composant ONT que sont $V_{\text{ONT}} \rightarrow Rx_{\text{ONT}}$ et $Fiber_{\text{DB}} \rightarrow Rx_{\text{ONT}}$ ainsi que deux dépendances distribuées entre les composants OLT et ONT que sont $Tx_{\text{OLT}} \rightarrow Rx_{\text{ONT}}$ et $Fiber_{\text{T}} \rightarrow Rx_{\text{ONT}}$.
- Le compteur d'erreurs de transmission $BIP_{ds}[i]$ sur le canal descendant, les alarmes $LOS_{ds}[i]$, $SD_{ds}[i]$, $SF_{ds}[i]$, $LCGD_{ds}[i]$ et $LOF_{ds}[i]$ ont chacune la même signification que leur homologue du canal montant. Les alarmes relatives à la qualité du signal sur le canal descendant peuvent engendrer une dégradation de la qualité des services fournis aux abonnés. En effet, supposons par exemple que nous souhaitons étendre le modèle initial du réseau GPON-FTTH en ajoutant le composant RG (Résidentiel Gateway) (voir la Figure 6.1). La passerelle résidentielle RG est un composant du réseau domestique sur lequel sont déployés les services fournis aux abonnés. Ce n'est donc pas un composant du réseau GPON-FTTH, mais l'ajouter au modèle initial du réseau GPON-FTTH, permettrait de corrélérer les alarmes de l'infrastructure GPON-FTTH aux dysfonctionnements des services fournis aux abonnés. Par exemple, sur la Figure 6.2.b, une telle corrélation peut être faite avec la dépendance $N_1, N_2, LOF_{ds}, LCGD_{ds} \rightarrow IPConnect$ distribuée entre les composants ONT et RG. La variable binaire $IPConnect$ désignerait dans ce cas l'état du service d'accès à Internet fourni par le composant RG aux abonnés.

6.2.3 Le modèle initial d'un PON

Le modèle initial d'un PON du réseau GPON-FTTH est présenté par la Figure 6.2. La propagation locale de pannes et d'alarmes sur chacun des composants OLT et ONT est modélisée ainsi que la propagation distribuée de pannes entre ces deux composants. Conformément à la méthode générique présentée dans le Chapitre 5, chacun de ces deux composants est modélisé par deux graphes de dépendances interconnectés. Cette méthode de modélisation du réseau GPON-FTTH permet de séparer la propagation locale de pannes et d'alarmes sur chacun des composants OLT et ONT de la propagation distribuée de pannes et d'alarmes entre ces deux composants. Cette séparation implémentée par les couches 1 et 2 du modèle de la Figure 6.2 apporte les propriétés de modularité et

d'extensibilité. Ainsi le modèle initial du réseau d'accès GPON-FTTH peut être facilement étendu pour inclure des composants du réseau de collecte en amont du composant OLT modélisé par la Figure 6.2.a, ou pour inclure des composants du réseau domestique des abonnés en aval du composant ONT modélisé par la Figure 6.2.b.

6.3 Paramètres experts du modèle initial d'un PON

Dans la section précédente, nous avons montré comment utiliser les connaissances métiers dites "expertes" du réseau GPON-FTTH pour construire un modèle de la topologie et de la propagation de pannes et d'alarmes sur ce réseau. Dans cette section, nous montrons comment ces connaissances "expertes" peuvent être utilisées pour déterminer approximativement les paramètres du modèle, c'est-à-dire les distributions de probabilités conditionnelles qui permettent de quantifier les incertitudes sur certaines relations de causes à effets du phénomène de propagation de pannes et d'alarmes. Ce choix est motivé par le fait que l'apprentissage automatique des paramètres d'un modèle statistique basé sur un réseau bayésien, à partir d'un échantillon de données très souvent incomplet comme c'est notre cas, avec un algorithme basé sur le principe du maximum de vraisemblance des données, nécessite une valeur initiale du vecteur des paramètres assez proche de la réalité afin d'espérer contourner les problèmes liés aux optimaux locaux de la fonction de vraisemblance des données. Les paramètres que nous déterminons approximativement dans cette section pourront servir plus tard de point d'initialisation à un algorithme tel que l'EM (Expectation Maximization) qui pourra donc les ajuster automatiquement.

Considérons une variable quelconque X_i du modèle d'un PON présenté par la Figure 6.2. Soit $\theta_{i,j,k} = \mathbb{P}(X_i = k | pa(X_i) = j)$ la probabilité que la variable X_i prenne la valeur k lorsque ses parents sont dans la configuration j . Une configuration d'un ensemble de variables est une combinaison de valeurs prises par ces variables. On se propose de déterminer approximativement pour chaque variable X_i du modèle d'un PON, la distribution de probabilités conditionnelles $\theta_i = \mathbb{P}(X_i | pa(X_i))$, sur la base des connaissances dites "expertes" du réseau GPON-FTTH. L'idée n'est pas de déterminer précisément les quantités $\theta_{i,j,k}$ mais plutôt de trouver :

- un ordre de grandeur entre les proportions du vecteur $\theta_{i,j,\bullet}$, pour toutes les valeurs possibles de k , avec les contraintes $\sum_k \theta_{i,j,k} = 1$ et $\theta_{i,j,k} \in [0, 1]$,
- un ordre de grandeur entre les probabilités du vecteur $\theta_{i,\bullet,k}$, pour toutes les valeurs possibles de j .

En bref, nous cherchons à déterminer un ordre de grandeur entre les éléments de chaque ligne et entre les éléments de chaque colonne de la matrice à deux dimensions $\theta_i = \mathbb{P}(X_i | pa(X_i))$. La distribution θ_i n'est donc pas précise, mais elle doit être cohérente avec le comportement d'un PON du réseau GPON-FTTH. Nous illustrons avec quelques exemples ci-dessous, cette approche d'estimation paramétrique d'un modèle à base des connaissances métiers dans le domaine modélisé, c'est-à-dire la propagation de pannes et d'alarmes sur un PON d'un réseau GPON-FTTH.

Distribution de probabilités conditionnelles $\mathbb{P}(\text{LOF}_{\text{us}} | \mathbf{O}_1, \text{LCDG}_{\text{us}})$: la Table 6.1 présente cette distribution. Notons $+lof$ l'évènement de perte d'une trame de données qui se manifeste par l'observation de l'alarme LOF (Loss Of Frame). $\neg lof$ est le complémentaire de l'évènement $+lof$. Sur le canal montant entre un ONT et l'OLT, il est plus

probable que la perte d'une trame de données (+*lof*) se produise lorsqu'une erreur de délimitation des trames transmises par cet ONT se produit (+*lcd*, loss of gem channel delineation), que lorsque l'OLT parvient à délimiter les trames sans erreurs (\neg *lcd*). Cette connaissance métier nous permet de considérer une contrainte telle que celle de l'Inéquation (6.1), dans laquelle l'évènement +*e* correspond à l'évènement +*lcd* et le complémentaire $\neg e$ de +*e* correspond à \neg *lcd* (voir la Table 6.1).

$$\theta_{p,q}^{\neg e} \leq \theta_{p,q}^{+e} \quad \text{pour } p = 1, q \in \{1, 2, 3\} \quad (6.1)$$

Par ailleurs, puisque le taux d'erreurs de transmission est plus important lorsqu'une alarme SF (Signal Fail) est observée que lorsqu'une alarme SD (Signal Degraded) l'est, il est plus probable que la perte d'une trame se produise lorsque le niveau de dégradation du signal est important (SF) que lorsqu'il est moyen (SD). Nous pouvons donc aussi considérer une seconde contrainte telle que celle donnée par l'Inéquation (6.2).

$$\begin{aligned} \theta_{p,q}^{+e} &\leq \theta_{p,q+1}^{+e} \\ \theta_{p,q}^{\neg e} &\leq \theta_{p,q+1}^{\neg e} \quad \text{pour } p = 1, q \in \{1, 2\} \end{aligned} \quad (6.2)$$

Enfin, nous pouvons dire qu'il est plus probable qu'une perte de trames survienne sur le canal montant lorsque ses deux causes potentielles se sont produites, c'est-à-dire la dégradation du signal optique observée au travers de l'alarme SD_{us} ou SF_{us} et une erreur de délimitation de trames observée via l'alarme $LCDG_{us}$, que lorsqu'une seule des deux causes potentielles s'est produite. On peut donc aussi considérer une troisième contrainte telle que l'Inéquation (6.3) (voir la Table 6.1).

$$\theta_{1,2}^{\neg e} \leq \theta_{1,3}^{+e} \quad \text{et} \quad \theta_{1,3}^{\neg e} \leq \theta_{1,2}^{+e} \quad (6.3)$$

TABLE 6.1 – Distribution de probabilités conditionnelles $\mathbb{P}(\text{LOF}_{us} | O_1, \text{LCDG}_{us})$.

O_1	LCDG_{us}	+ <i>lof</i>	\neg <i>lof</i>
+	+ <i>lcd</i>	$\theta_{1,1}^{+lcd} \approx 0.600$	$1 - \theta_{1,1}^{+lcd}$
SD		$\theta_{1,2}^{+lcd} \approx 0.800$	$1 - \theta_{1,2}^{+lcd}$
SF		$\theta_{1,3}^{+lcd} \approx 0.800$	$1 - \theta_{1,3}^{+lcd}$
+	\neg <i>lcd</i>	$\theta_{1,1}^{\neg lcd} \approx 0.100$	$1 - \theta_{1,1}^{\neg lcd}$
SD		$\theta_{1,2}^{\neg lcd} \approx 0.700$	$1 - \theta_{1,2}^{\neg lcd}$
SF		$\theta_{1,3}^{\neg lcd} \approx 0.700$	$1 - \theta_{1,3}^{\neg lcd}$

Les probabilités conditionnelles de la distribution $\mathbb{P}(\text{LOF}_{us} | O_1, \text{LCDG}_{us})$ doivent donc satisfaire les contraintes formalisées par les inéquations (6.1), (6.2) et (6.3). En d'autres termes, ces trois contraintes formalisent les connaissances dites "expertes" encodées dans la distribution $\mathbb{P}(\text{LOF}_{us} | O_1, \text{LCDG}_{us})$. La Table 6.1 donne également une valeur approximative de chaque probabilité conditionnelle cohérente avec l'ordre de grandeur défini par les trois contraintes ci-dessus. Les paragraphes ci-dessous présentent d'autres exemples illustrant également la méthode utilisée pour déterminer approximativement les lois conditionnelles du modèle présenté à la Figure 6.2.

Distribution de probabilités conditionnelles $\mathbb{P}(\text{LCDG}_{us}|\text{BIP}_{us}, \text{DOW})$: La Table 6.2 présente la loi conditionnelle de la variable LCDG_{us} déterminée approximativement sur la base des contraintes (6.1), (6.2) et (6.3), dans lesquelles l'évènement $+e$ correspond à l'évènement $+dow$ et le complémentaire $\neg e$ de $+e$ correspond à $\neg dow$. La contrainte (6.1) exprime que, pour un quelconque taux d'erreurs de transmission BIP_{us} , il est plus probable qu'une erreur de délimitation de trames (évènement $+lcd$) se produise lorsqu'un saut de fenêtre de temps est observé (évènement $+dow$), que lorsqu'il ne l'est pas. La contrainte (6.2) formalise qu'indépendamment de l'évènement $+dow$, la probabilité qu'une erreur de délimitation de trames se produise, augmente avec le taux d'erreurs BIP_{us} . D'après la troisième contrainte (6.3), il est plus probable qu'une erreur de délimitation de trames survienne lorsque ses deux causes potentielles sont considérées que lorsqu'une seule d'entre elles est considérée.

TABLE 6.2 – Distribution de probabilités conditionnelles $\mathbb{P}(\text{LCDG}_{us}|\text{BIP}_{us}, \text{DOW})$.

BIP_{us}	DOW	$+lcd$	$\neg lcd$
$+$	$+dow$	$\theta_{1,1}^{+dow} \approx 0.100$	$1 - \theta_{1,1}^{+dow}$
low		$\theta_{1,2}^{+dow} \approx 0.800$	$1 - \theta_{1,2}^{+dow}$
$high$		$\theta_{1,3}^{+dow} \approx 0.900$	$1 - \theta_{1,3}^{+dow}$
$+$	$\neg dow$	$\theta_{1,1}^{-dow} \approx 0.001$	$1 - \theta_{1,1}^{-dow}$
low		$\theta_{1,2}^{-dow} \approx 0.700$	$1 - \theta_{1,2}^{-dow}$
$high$		$\theta_{1,3}^{-dow} \approx 0.700$	$1 - \theta_{1,3}^{-dow}$

Distribution de probabilités conditionnelles $\mathbb{P}(\text{LOS}_{us}|\text{Rx}_{OLT}, \text{TIA})$: La Table 6.3 présente la loi conditionnelle de la variable LOS_{us} déterminée approximativement sur la base des contraintes (6.1) et (6.2). La contrainte (6.1) exprime que, pour une quelconque valeur du niveau de puissance Rx_{OLT} , il est plus probable qu'une rupture de communication entre l'OLT et un ONT (évènement $+los$) se produise lorsqu'une interférence de transmission est observée (évènement $+tia$) sur le PON, que lorsqu'il ne l'est pas. La contrainte (6.2) formalise qu'indépendamment de l'évènement $+tia$, la probabilité d'une rupture de communication entre l'OLT et un ONT augmente lorsque la puissance de cet ONT reçue à l'OLT diminue. Nous noterons que la contrainte (6.3) n'est pas vérifiée pour cette distribution. En effet, la perte de communication LOS_{us} entre l'OLT et un ONT est très influencée par le niveau de puissance Rx_{OLT} de cet ONT reçu à l'OLT, que par une interférence TIA.

TABLE 6.3 – Distribution de probabilités conditionnelles $\mathbb{P}(\text{LOS}_{us}|\text{Rx}_{OLT}, \text{TIA})$.

Rx_{OLT}	TIA	$+los$	$\neg los$
Rx_n	$+tia$	$\theta_{1,1}^{+tia} \approx 0.500$	$1 - \theta_{1,1}^{+tia}$
Rx_l		$\theta_{1,2}^{+tia} \approx 0.890$	$1 - \theta_{1,2}^{+tia}$
Rx_0		$\theta_{1,3}^{+tia} \approx 0.990$	$1 - \theta_{1,3}^{+tia}$
Rx_n	$\neg tia$	$\theta_{1,1}^{-tia} \approx 0.010$	$1 - \theta_{1,1}^{-tia}$
Rx_l		$\theta_{1,2}^{-tia} \approx 0.800$	$1 - \theta_{1,2}^{-tia}$
Rx_0		$\theta_{1,3}^{-tia} \approx 0.990$	$1 - \theta_{1,3}^{-tia}$

Intéressons nous à présent à un exemple plus dense : l'approximation de la loi conditionnelle du niveau de puissance Rx_{OLT} . Cette variable est influencée par ses quatre pré-

décèsseurs dans le modèle présenté par la Figure 6.2, d’où la difficulté a priori introduite par le nombre de combinaisons des états de ces prédécesseurs. Cependant, nous verrons ci-dessous comment les connaissances métiers relatives à la manière dont les prédécesseurs de la variable $R_{x_{OLT}}$ l’influencent, simplifie l’approximation de sa loi conditionnelle.

Distribution de probabilités conditionnelles $\mathbb{P}(R_{x_{OLT}}|Fiber_T, V_{OLT}, Fiber_{DB}, Tx_{ONT})$:

- Indépendamment des états de la fibre de transport $Fiber_T$, de la fibre de branchement $Fiber_{DB}$ et de la puissance Tx_{ONT} transmise par un ONT, le niveau de puissance $R_{x_{OLT}}$ reçu à l’OLT de cet ONT est très faible ou tend vers $-\infty$ en dBm ($R_{x_{OLT}} = R_{x_0}$) lorsque la tension d’alimentation de l’OLT est très faible voir quasiment nulle ($V_{OLT} = V_0$). La Table 6.4 représente cette connaissance. Le nombre ϵ est un réel positif et très proche de zéro. Il permet d’éviter de rencontrer des problèmes numériques tels que la division par zéro.

TABLE 6.4 – Distribution de probabilités conditionnelles $\mathbb{P}(R_{x_{OLT}}|\bullet, V_0, \bullet, Tx_{ONT})$.

$Fiber_T$	V_{OLT}	$Fiber_{DB}$	Tx_{ONT}	R_{x_0}	R_{x_l}	R_{x_n}
•	V_0	•	Tx_0	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$
			Tx_l	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$
			Tx_n	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$

- Lorsque la fibre optique de transport est rompue ($Fiber_T = BR$, avec "BR" signifiant "BBroken") ou la fibre de branchement d’un ONT ($Fiber_{DB} = BR$), on est quasiment certain que le niveau de puissance reçue à l’OLT de cet ONT est très faible (tend vers $-\infty$ en dBm). La Table 6.5 encode cette connaissance.

TABLE 6.5 – Distribution de probabilités conditionnelles $\mathbb{P}(R_{x_{OLT}}|\bullet, \bullet, BR, Tx_{ONT})$.

$Fiber_T$	V_{OLT}	$Fiber_{DB}$	Tx_{ONT}	R_{x_0}	R_{x_l}	R_{x_n}
•	•	BR	Tx_0	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$
			Tx_l	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$
			Tx_n	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$

- Lorsque la tension d’alimentation de l’OLT est nominale ($V_{OLT} = V_1$) et que l’une des deux fibres est en bon état tandis que l’autre souffre d’atténuation, le niveau de puissance $R_{x_{OLT}}$ est très influencé par la puissance Tx_{ONT} transmise par l’ONT. La Table 6.6 représente cette connaissance.

TABLE 6.6 – Distribution de probabilités conditionnelles $\mathbb{P}(R_{x_{OLT}}|OK, V_1, BR, Tx_{ONT})$.

$Fiber_T$	V_{OLT}	$Fiber_{DB}$	Tx_{ONT}	R_{x_0}	R_{x_l}	R_{x_n}
OK	V_1	AT	Tx_0	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$
			Tx_l	$0.8 - \epsilon$	0.2	ϵ
			Tx_n	0.6	$0.4 - \epsilon$	ϵ

- Lorsque la tension d'alimentation de l'OLT est nominale ($V_{OLT} = V_1$) et que les deux fibres introduisent de l'atténuation, l'atténuation conjointe des deux fibres est probablement plus importante que celle d'une seule fibre. C'est pourquoi les probabilités des évènements $Rx_{OLT} = Rx_0$ sont un peu plus grandes dans la Table 6.7 que dans la Table 6.6.

TABLE 6.7 – Distribution de probabilités conditionnelles $\mathbb{P}(Rx_{OLT}|AT, V_1, AT, Tx_{ONT})$.

$Fiber_T$	V_{OLT}	$Fiber_{DB}$	Tx_{ONT}	Rx_0	Rx_l	Rx_n
AT	V_1	AT	Tx_0	$1.0 - \epsilon$	$\epsilon/2$	$\epsilon/2$
			Tx_l	$0.9 - \epsilon$	0.1	ϵ
			Tx_n	0.7	$0.3 - \epsilon$	ϵ

Les Tables 6.4, 6.5, 6.6 et 6.7 permettent de prendre en considération toutes les combinaisons possibles des états de la fibre de transport du PON, de la tension d'alimentation de l'OLT, de la fibre de branchement d'un ONT et de la puissance transmise par cet ONT. On parvient donc à approximer la loi conditionnelle de la variable Rx_{OLT} .

Un raisonnement similaire à celui développé dans cette section est appliqué pour déterminer approximativement la loi de chaque variable conditionnellement à ses prédécesseurs dans le modèle de la Figure 6.2. Nous rappelons que l'approximation de ces distributions de probabilités conditionnelles ne consiste pas à trouver des nombres comme cela peut le sembler, mais plutôt à structurer un raisonnement cohérent avec le comportement d'un PON du réseau d'accès GPON-FTTH. Nous suivons ainsi la logique de Glenn Shafer : "Probability is not really about numbers; it is about the structure of reasoning".

6.4 Validation du modèle initial d'un PON

Nous présentons dans cette section, la méthodologie de validation du modèle probabiliste de propagation de pannes et d'alarmes sur le réseau GPON-FTTH. La validation consiste à vérifier que les résultats de diagnostic effectués avec ce modèle probabiliste sont conformes avec les connaissances métiers relatives au fonctionnement du réseau GPON-FTTH. Le modèle probabiliste est doublement validé. Une première validation est faite sur la plateforme expérimentale du réseau GPON-FTTH d'Orange Labs. Une seconde validation est faite sur le réseau GPON-FTTH opérationnel de l'opérateur Orange.

6.4.1 Validation sur une plateforme expérimentale

Les expériences sont faites sur la plateforme expérimentale du réseau GPON-FTTH d'Orange Labs. Nous avons considéré un PON de deux ONTs nommés ONT_1 et ONT_2 . La plateforme nous permet d'émuler des pannes et de collecter les alarmes déclenchées. Il est aussi possible de lire, s'ils sont disponibles, les valeurs des compteurs d'erreurs BIP_{us} , BIP_{ds} , des paramètres scalaires Rx_{OLT} , Rx_{ONT} , Tx_{OLT} , Tx_{ONT} , des courants de polarisation I_{OLT} , I_{ONT} , des tensions V_{OLT} , V_{ONT} et des températures T_{OLT}^c , T_{ONT}^c de l'OLT et des deux ONTs connectés sur le PON considéré. Chaque table de la Table 6.8 à la Table 6.14 montre les croyances ou probabilités a posteriori des variables représentant les causes racines, calculées par notre implémentation en Python du modèle du réseau GPON-FTTH à la Figure 6.2, sur la base des évidences observées. Le titre de chacune de ces tables est un fait

résumant les évidences observées et décrites en détail en dessous de chaque table.

Nous débutons l'évaluation du modèle probabiliste pour le diagnostic automatique du réseau d'accès GPON-FTTH, en vérifiant que l'inférence sur ce modèle ne retourne aucune panne lorsqu' aucune alarme n'est observée sur le PON et les valeurs des compteurs et paramètres scalaires sont nominaux. La Table 6.8 présente le résultat de ce test. On y observe que l'état positif ou nominal en gras de chaque cause racine est l'état le plus probable avec une très forte croyance.

TABLE 6.8 – État nominal de l'arbre PON.

Causes Racines	États	Croyances
$Fiber_T$	[OK , AT, BR]	[0.99 , 6.e-03, 7.e-12]
Alt_{OLT}	[OK , \neg OK]	[0.99 , 4.e-07]
FaultyONT[1]	[+ f_{to} , \neg f_{to}]	[0.007, 0.993]
FaultyONT[2]	[+ f_{to} , \neg f_{to}]	[0.014, 0.986]
IOS[1]	[OK , \neg OK]	[0.980 , 0.020]
IOS[2]	[OK , \neg OK]	[0.980 , 0.020]
ONTConfMis[1]	[+ ocm , \neg ocm]	[0.002, 0.998]
ONTConfMis[2]	[+ ocm , \neg ocm]	[0.002, 0.998]
$Fiber_{DB}[1]$	[OK , AT, BR]	[0.91 , 8.e-02, 9.e-07]
$Fiber_{DB}[2]$	[OK , AT, BR]	[0.91 , 8.2-02, 9.e-07]
$Alt_{ONT}[1]$	[OK , \neg OK]	[0.99 , 1.e-07]
$Alt_{ONT}[2]$	[OK , \neg OK]	[0.99 , 1.e-07]

Aucune alarme n'est observée sur le PON, les compteurs et les paramètres scalaires sont nominaux pour l'OLT et pour chacun des deux ONTs, ONT_1 et ONT_2 connectés au PON.

La Table 6.9 montre que l' ONT_1 est diagnostiqué comme étant en panne ou défectueux lorsqu'il perd la communication bidirectionnelle avec l'OLT, bien qu'aucune alarme relative à la qualité du signal n'est observée. Il faut noter l'absence des valeurs des compteurs et des paramètres scalaires de l' ONT_1 puisque le système de gestion NMS (Network Management System) du réseau GPON-FTTH ne dispose d'aucun moyen pour récupérer ces valeurs dans la MIB (Management Information Base) de l' ONT_1 . Ces valeurs manquantes, pourtant utiles pour calculer le diagnostic le plus probable, sont inférées par le modèle probabiliste sur la base de ses distributions de probabilités conditionnelles déterminées approximativement à la Section 6.3.

Cependant, la perte de la communication entre l'OLT et l' ONT_1 peut être aussi due à la coupure de la fibre de branchement de l' ONT_1 puisque cette fibre optique se trouve chez l'abonné et par conséquent, elle n'est pas protégée par les infrastructures de l'opérateur de télécommunications. En plus, d'après le fonctionnement du réseau GPON-FTTH, l'alarme LOS[1] observée toute seule ne permet pas, même à un expert avéré de ce réseau, de discriminer entre ces deux causes racines potentielles. Nous verrons un peu plus tard dans la Table 6.10 que la présence d'autres alarmes en plus telles que celles relatives à la qualité du signal optique permettent de lever cette incertitude. Sur la Table 6.10, la présence des alarmes indiquant une dégradation du signal optique entre l'OLT et l' ONT_1 , apporte des données additionnelles au modèle probabiliste et change les probabilités a pos-

tériori précédemment obtenues dans la Table 6.9. l'ONT₁ n'est plus diagnostiqué comme étant défectueux, mais c'est plutôt sa fibre de branchement qui souffre d'une atténuation trop importante provoquant la rupture de communication avec l'OLT. Il faut remarquer sur la Table 6.10 que, contrairement à la Table 6.9, la probabilité que l'ONT₁ soit défectueux diminue en faveur de l'augmentation de la croyance que la fibre de branchement de l'ONT₁ atténue trop.

TABLE 6.9 – Rupture soudaine de la communication entre l'OLT et l'ONT₁.

Causes racines	États	Croyances
<i>Fiber_T</i>	[OK, AT, BR]	[0.99, 6.e-03, 1.e-11]
FaultyONT[1]	[+ <i>fto</i> , ¬ <i>fto</i>]	[0.998 , 0.002]
FaultyONT[2]	[+ <i>fto</i> , ¬ <i>fto</i>]	[0.014, 0.986]
<i>Fiber_{DB}</i> [1]	[OK, AT, BR]	[0.91, 8.e-02, 1.e-06]
<i>Fiber_{DB}</i> [2]	[OK, AT, BR]	[0.91, 8.e-02, 9.e-07]

Les alarmes LOS_{us}[1] et LOS_{ds}[1] sont respectivement observées sur le canal montant et sur le canal descendant entre l'OLT et l'ONT₁, les compteurs et les paramètres scalaires de l'ONT₁ sont inconnus tandis que ceux de l'ONT₂ sont connus et nominaux.

TABLE 6.10 – Rupture progressive de la communication entre l'OLT et l'ONT₁.

Causes racines	États	Croyances
<i>Fiber_T</i>	[OK, AT, BR]	[0.94, 5.e-02, 1.e-7]
FaultyONT[1]	[+ <i>fto</i> , ¬ <i>fto</i>]	[0.422, 0.578]
FaultyONT[2]	[+ <i>fto</i> , - <i>fto</i>]	[0.014, 0.986]
<i>Fiber_{DB}</i> [1]	[OK, AT , BR]	[0.409, 0.577 , 0.014]
<i>Fiber_{DB}</i> [2]	[OK, AT, BR]	[0.92, 7.e-02, 1.e-06]

Les alarmes LOS_{us}[1], LOS_{ds}[1], ainsi que les alarmes relatives à la qualité du signal LCGD_{us}[1], LCGD_{ds}[1], SD_{us}[1], SD_{ds}[1] sont observées. Les compteurs et les paramètres scalaires de l'ONT₁ sont inconnus tandis que ceux de l'ONT₂ sont connus et nominaux.

La Table 6.11 présente une situation pour laquelle le diagnostic le plus probable est l'atténuation importante de la fibre de branchement de l'ONT₁, lorsque l'alarme SD[1] (Signal Degraded) est observée sur le canal montant et sur le canal descendant, suite à des erreurs de transmission et des niveaux de puissances faibles observées. Si pour une quelconque raison, le NMS (Network Management System) du réseau GPON-FTTH ne parvient pas à récupérer les valeurs des compteurs d'erreurs et des niveaux de puissances, le modèle probabiliste est toujours capable de calculer le bon diagnostic bien que des données très importantes soient manquantes. Dans ce dernier cas, la croyance en l'atténuation importante de la fibre de branchement de l'ONT₁ diminue de 0.92 sur la Table 6.11 à 0.76 sur la Table 6.12. Mais la décision finale reste la même sur les deux Tables. Le scénario décrit par la Table 6.12 met donc bien en évidence la capacité du modèle et du raisonnement probabiliste à faire face à certaines données manquantes pourtant utiles pour effectuer un diagnostic efficace.

TABLE 6.11 – Dégradation complètement observée du signal entre l'OLT et l'ONT₁.

Causes racines	États	Croyances
$Fiber_T$	[OK, AT, BR]	[0.91, 8.e-02, 1.e-11]
FaultyONT[1]	[+ f_{to} , $\neg f_{to}$]	[1.e-06, 0.999]
FaultyONT[2]	[+ f_{to} , $\neg f_{to}$]	[0.014, 0.986]
$Fiber_{DB}[1]$	[OK, AT , BR]	[7.e-02, 0.92 , 2.e-06]
$Fiber_{DB}[2]$	[OK, AT, BR]	[0.92, 7.e-02, 1.e-06]

Les alarmes $SD_{us}[1]$ et $SD_{ds}[1]$ sont observées, une variation de chacun des compteurs $BIP_{us}[1]$ et $BIP_{ds}[1]$ est observée, les niveaux de puissances $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ sont faibles. Les compteurs et les paramètres scalaires de l'ONT₂ sont nominaux.

TABLE 6.12 – Dégradation partiellement observée du signal entre l'OLT et l'ONT₁.

Causes racines	États	Croyances
$Fiber_T$	[OK, AT, BR]	[0.93, 6.e-02, 1.e-09]
FaultyONT[1]	[+ f_{to} , $\neg f_{to}$]	[0.082, 0.918]
FaultyONT[2]	[+ f_{to} , $\neg f_{to}$]	[0.014, 0.986]
$Fiber_{DB}[1]$	[OK, AT , BR]	[2.e-01, 0.76 , 1.e-04]
$Fiber_{DB}[2]$	[OK, AT, BR]	[0.92, 7.e-02, 1.e-06]

Les alarmes $SD_{us}[1]$ et $SD_{ds}[1]$ sont observées, les compteurs et les paramètres scalaires de l'ONT₁ sont inconnus tandis que ceux de l'ONT₂ sont nominaux.

Maintenant, si l'on observe une dégradation du signal entre l'OLT et l'ONT₁ d'une part et entre l'OLT et l'ONT₂ d'autre part, telle que décrit par la Table 6.13, le modèle probabiliste diagnostique l'atténuation importante de la fibre de transport du PON partagée par les deux ONTs et non pas l'atténuation de chacune des fibres de branchement des deux ONTs. Il faut remarquer sur la Table 6.13 que la probabilité d'atténuation de la fibre de transport est 0.58 tandis que celle de chacune des deux fibres de branchement est 0.50. La différence entre ces probabilités semble ne pas être très significative. Par conséquent, on peut avoir tendance à croire que le modèle a du mal à prendre la meilleure décision entre l'atténuation de la fibre de transport ou celle de chacune des fibres de branchement. Mais, si on calcule la probabilité jointe de l'atténuation des deux fibres de branchement ($0.5 \times 0.5 = 0.25$), on remarque que la différence entre cette probabilité jointe et celle de l'atténuation de la fibre de transport est très significative. Ce dernier résultat renforce la croyance que c'est bien la fibre de transport qui atténue trop. Il faut noter que la probabilité jointe d'atténuation des deux fibres de branchement est le produit des probabilités d'atténuation de chacune des deux fibres. En effet, la couverture de Markov $MB(Fiber_{DB}[i])$ de la variable $Fiber_{DB}[i]$ est définie par l'union des ensembles suivants : l'ensemble PA_i des parents de $Fiber_{DB}[i]$ dans le modèle, l'ensemble CH_i de ses enfants et l'ensemble OP_i des autres parents de ses enfants.

$$MB_i = PA_i \cup CH_i \cup OP_i \quad (6.4)$$

l'ensemble PA_i est vide puisque $Fiber_{DB}[i]$ est une cause racine, l'ensemble CH_i est égal à $\{Rx_{OLT}[i], Rx_{ONT}[i]\}$ et l'ensemble $OP_i = \{Tx_{ONT}[i], V_{ONT}[i], V_{OLT}, Tx_{OLT}, Fiber_T\}$. Les fibres

TABLE 6.13 – Dégradation du signal entre l’OLT et chacun des deux ONTs.

Causes racines	États	Croyances
$Fiber_T$	[OK, AT , BR]	[0.41, 0.58 , 1.e-09]
FaultyONT[1]	[+ f_{to} , $\neg f_{to}$]	[1.e-04, 0.999]
FaultyONT[2]	[+ f_{to} , $\neg f_{to}$]	[5.e-04, 0.999]
$Fiber_{DB}[1]$	[OK, AT , BR]	[0.49, 0.50 , 1.e-05]
$Fiber_{DB}[2]$	[OK, AT , BR]	[0.49, 0.50 , 1.e-05]

Les alarmes $SD_{us}[1]$, $SD_{ds}[1]$, $SD_{us}[2]$ et $SD_{ds}[2]$ sont observées, une variation de chacun des compteurs d’erreurs $BIP_{us}[1]$, $BIP_{ds}[1]$, $BIP_{us}[2]$ et $BIP_{ds}[2]$ est observée, les niveaux de puissances reçues sont faibles pour les deux ONTs.

de branchement $Fiber_{DB}[1]$ et $Fiber_{DB}[2]$ sont indépendantes puisque la variable aléatoire $Fiber_{DB}[1] \notin MB(Fiber_{DB}[2])$ et $Fiber_{DB}[2] \notin MB(Fiber_{DB}[1])$.

TABLE 6.14 – Perte de la communication sur le canal descendant entre l’OLT et l’ONT₁

Causes racines	États	Croyances
$Fiber_T$	[OK, AT, BR]	[0.92, 7.e-02, 3.e-11]
FaultyONT[1]	[+ f_{to} , $\neg f_{to}$]	[0.094, 0.906]
FaultyONT[2]	[+ f_{to} , $\neg f_{to}$]	[0.014, 0.986]
$Fiber_{DB}[1]$	[OK, AT , BR]	[8.e-02, 0.91 , 4.e-06]
$Fiber_{DB}[2]$	[OK, AT, BR]	[0.92, 7.e-02, 1.e-06]

L’alarme $LOS_{ds}[1]$ est observée, les compteurs et les paramètres scalaires de l’ONT₁ sont inconnus tandis que ceux de l’ONT₂ sont connus et nominaux.

La Table 6.14 illustre une situation pour laquelle on observe une perte de communication sur le canal descendant entre l’OLT et l’ONT₁. Il s’agit donc d’une rupture unidirectionnelle de la communication. Contrairement à une rupture bidirectionnelle de la communication telle qu’illustrée par la Table 6.9, le modèle probabiliste ne diagnostique plus l’ONT₁ comme étant défectueux. Le diagnostic est plutôt l’atténuation de la fibre de branchement de l’ONT₁. Dans ce cas l’atténuation de cette fibre n’est pas suffisamment importante pour impacter également le canal montant entre l’OLT et l’ONT₁. Le canal montant est justement moins sensible à l’atténuation de la fibre de branchement que le canal descendant.

Les Tables ci-dessus montrent que les résultats de diagnostic obtenus avec le modèle probabiliste du réseau GPON-FTTH sont cohérents avec les expériences faites sur la plateforme de test du réseau GPON-FTTH d’Orange Labs. En effet, ces résultats ont été inspectés manuellement afin de vérifier leur fiabilité. Ceci démontre bien que le diagnostic automatique d’un réseau de télécommunication avec un modèle à base de réseaux bayésiens est une méthode pertinente et très prometteuse.

6.4.2 Validation sur un réseau GPON-FTTH opérationnel

Les résultats de l'outil de diagnostic PANDA (Probabilistic tool for GPON-FTTH Access Network self-Diagnosis) qui implémente le modèle probabiliste, sont présentés et analysés dans cette sous-section. Il s'agit de confronter le modèle probabiliste à la réalité en s'intéressant aux anomalies réelles observées sur le réseau GPON-FTTH opérationnel de l'opérateur Orange. Plusieurs arbres PON de tailles différentes (nombre d'ONTs connectés) sont considérés. Pour chaque PON, nous appelons "ONT testé" noté ONT_1 , l'ONT de l'abonné qui sollicite un diagnostic de sa ligne FTTH. Les autres ONTs du PON sont appelés les voisins de l'ONT testé. Nous collectons les alarmes observées, les valeurs éventuelles des compteurs BIP_{us} , BIP_{ds} , des paramètres scalaires Rx_{OLT} , Rx_{ONT} , Tx_{OLT} , Tx_{ONT} , des courants de polarisation I_{OLT} , I_{ONT} , des tensions d'alimentation V_{OLT} , V_{ONT} et des températures T_{OLT}^c , T_{ONT}^c de l'OLT et des ONTs connectés sur le PON considéré. Une base de données de 10611 cas réels de diagnostic est construite pour confronter l'outil PANDA à la réalité. Un cas est un ensemble de données réelles collectées sur les équipements du réseau GPON-FTTH opérationnel d'Orange. Ces données réelles constituent les observations ou évidences en entrée du modèle probabiliste. Chaque Table de la Table 6.15 à la Table 6.25, excepté les Tables 6.16, 6.22 et 6.25 présente une instance de cas de diagnostic basé sur des observations réelles générées par le réseau GPON-FTTH opérationnel d'Orange. Le titre de chaque table est une brève description des observations du cas considéré. Ces observations sont détaillées en dessous de chaque table.

TABLE 6.15 – Dégradation du signal entre l'OLT et chacun des ONTs connectés sur un PON de grande dimension.

Causes racines	États	Croyances
$Fiber_T$	[OK, AT , BR]	[2.e-02, 0.97 , 9.e-06]

Le PON contient quarante ONTs. L'alarme LEVELLO est observée pour tous les ONTs, c'est-à-dire que $N_2[i \in \{1, \dots, 40\}] = LLO$ (voir la Figure 6.2). Le niveau de puissance de l'ONT₁ reçue à l'OLT est faible ainsi que le niveau de puissance reçue à l'ONT₁, c'est-à-dire que les variables $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ ont des valeurs non nominales. Les compteurs et les paramètres scalaires des ONTs voisins sont inconnus.

TABLE 6.16 – États inférés des variables manquantes du cas de la Table 6.15.

Variables manquantes	États	Croyances
$Rx_{OLT}[i \neq 1]$	[Rx₀ , Rx_l , Rx_n]	[0.88 , 0.10, 0.02]
$Rx_{ONT}[i \neq 1]$	[Rx₀ , Rx_l , Rx_n]	[0.82 , 0.17, 0.01]
$BIP_{us}[i]$	[+, low , high]	[0.01, 0.49 , 0.50]
$BIP_{ds}[i]$	[+, low , <i>high</i>]	[9.e-04, 0.98 , 1.e-03]
$O_1[i]$	[+, SD , SF]	[0.01, 0.49 , 0.50]
$N_1[i]$	[+, SD , SF]	[0.01, 0.98 , 0.01]

Les états les plus probables des variables non observées pourtant utiles pour détecter une atténuation de la fibre de transport du PON sur la Table 6.15.

Sur la Table 6.15, le diagnostic le plus probable calculé avec PANDA est l'atténuation importante de la fibre de transport du PON, puisque tous les ONTs connectés sur ce PON

ont de faibles niveaux de puissances sur les deux canaux de communications. La Table 6.16 montre les probabilités a posteriori des variables manquantes pourtant très utiles pour trouver le diagnostic de la Table 6.15. On peut remarquer que les états les plus probables de ces variables non observées, inférés par le modèle probabiliste, sont consistants avec le diagnostic de la Table 6.15. En effet, l'état le plus probable des puissances reçues $Rx_{OLT}[i \neq 1]$ sur le canal montant et celles reçues $Rx_{ONT}[i \neq 1]$ sur le canal descendant de chaque ONT voisin de l'ONT₁ est Rx_0 , dénotant un intervalle de valeurs de puissances considérées comme très faibles. On note aussi une variation très probable du compteur d'erreurs BIP sur les deux canaux de chaque ONT voisin, conduisant à une forte probabilité qu'une alarme SD ou SF indiquant une dégradation du signal, aurait été déclenchée pour chaque ONT sur le PON. Il faut bien remarquer que la probabilité d'observer l'alarme SD_{as} sur le canal descendant ($\mathbb{P}(N_1[i] = SD) = 0.98$) est significativement supérieure à celle d'observer SD_{us} sur le canal montant ($\mathbb{P}(O_1[i] = SD) = 0.49$). Ce résultat montre que le modèle probabiliste embarque suffisamment d'intelligence pour comprendre que le canal descendant d'un PON est plus sensible à l'atténuation de la fibre que le canal montant. Notons au passage qu'une règle experte aurait été incapable de produire un diagnostic pour une pareille situation dans laquelle plusieurs variables utiles pour calculer le diagnostic ne sont pas observées.

TABLE 6.17 – Dégradation du signal entre l'OLT et chacun des voisins de l'ONT₁ sur un PON de grande dimension.

Causes racines	États	Croyances
$Fiber_T$	[OK, AT , BR]	[0.12, 0.87 , 0.01]

Le PON contient quarante ONTs. Aucune alarme n'est observée pour l'ONT₁. Les compteurs et les paramètres scalaires de l'ONT₁ sont nominaux. L'alarme LEVELLO est observée pour chaque voisin de l'ONT₁, c'est-à-dire $N_2[i \in \{2, \dots, 40\}] = LLO$. Les compteurs et les paramètres scalaires des voisins sont inconnus.

TABLE 6.18 – Dégradation du signal entre l'OLT et l'ONT₁ connecté sur un PON de grande dimension.

Causes racines	États	Croyances
$Fiber_{DB}[1]$	[OK, AT , BR]	[1.e-03, 0.94 , 5.e-02]
$Fiber_{DB}[i \neq 1]$	[OK , AT, BR]	[0.91 , 8.e-02, 2.e-06]
$Fiber_T$	[OK , AT, BR]	[0.99 , 2.e-09, 1.e-48]

Le PON contient quarante ONTs. L'alarme LEVELLO est observée pour l'ONT₁. Les puissances $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ sont faibles. Aucune alarme n'est observée pour chaque voisin de l'ONT₁. Les niveaux de puissances de chaque ONT voisin sont nominaux.

Contrairement à la Table 6.15, la Table 6.17 illustre une situation pour laquelle l'alarme LEVELLO est observée seulement pour les voisins de l'ONT₁. Aucun symptôme relatif à l'atténuation du signal n'est observé pour l'ONT₁. Un humain ou une règle experte de diagnostic qui se focalise seulement sur l'ONT₁ diagnostiquera qu'il n'a aucun problème concernant l'ONT₁ puisqu' aucune alarme n'est observée pour cet ONT en plus du fait

que ses compteurs et paramètres scalaires sont tous nominaux. Mais, le modèle probabiliste, avec son analyse globale ou généralisée de l'arbre PON, nous indique que la fibre de transport du PON atténue. En effet, l'atténuation de la fibre de transport n'a pas affecté l'ONT₁ puisque les niveaux de puissances $R_{xOLT}[1]$ et $R_{xONT}[1]$ étaient très élevés avant le début de l'atténuation de la fibre de transport. Par contre les ONTs voisins ont été affectés puisque leurs niveaux de puissances reçues sur au moins un des canaux étaient nominaux avant le début de l'atténuation, mais très proches de la borne inférieure de la plage de puissances nominales. La Table 6.18 présente un cas symétriquement similaire au cas de la Table 6.17, c'est-à-dire que les symptômes relatifs à l'atténuation du signal sont plutôt observés pour l'ONT₁ et absents pour les voisins. Dans ce dernier cas, le modèle probabiliste raisonne très bien : parmi tous les ONTs connectés au même PON, si un seul ONT manifeste les symptômes d'atténuation du signal et que les autres ONTs ne le manifestent pas, alors ce n'est pas la fibre de transport partagée par tous les ONTs qui souffre d'atténuation. C'est plutôt la fibre de branchement de l'ONT manifestant les symptômes d'atténuation qui dégrade la qualité du signal entre l'OLT et cet ONT.

TABLE 6.19 – Rupture soudaine de communication entre l'OLT et l'ONT₁ sur un PON de grande dimension.

Causes racines	États	Croyances
FaultyONT[1] (1)	[+fto, $\neg fto$]	[0.59 , 0.41]
Fiber _{DB} [1] (2)	[OK, AT, BR]	[0.13, 0.36, 0.49]
Fiber _{DB} [<i>i</i>]	[OK , AT, BR]	[0.91 , 8.e-02, 1.e-06]

Le PON contient vingt ONTs, les alarmes $LOS_{us}[1]$ et $LOS_{ds}[1]$ sont observées pour l'ONT₁. Les compteurs et les paramètres scalaires de l'ONT₁ sont inconnus tandis que ceux des ONTs voisins sont connus et nominaux.

La Table 6.19 présente une situation similaire à celle présentée par la Table 6.9 dans la sous-section 6.4.1. Les deux tables montrent le diagnostic probabiliste de l'alarme LOS bidirectionnelle. Cependant, dans la situation illustrée par la Table 6.19, le PON a vingt ONTs contre seulement deux dans la Table 6.9. Ce qui signifie que dans la Table 6.19, on a beaucoup d'observations relatives aux dix-neuf voisins de l'ONT₁. Ce flux de données aide le modèle probabiliste à prendre une décision plus raisonnable que dans la Table 6.9. Le modèle probabiliste calcule un podium de causes racines probables. La première est que l'ONT₁ est défectueux et la seconde est la rupture de la fibre de branchement de cet ONT. Nous avons néanmoins vu avec la situation décrite par la Table 6.10 que la présence d'autres alarmes relatives à la qualité du signal change ce résultat de diagnostic. La Table 6.20 décrit une situation dans laquelle les alarmes LOS_{us} et LOS_{ds} ne sont plus seulement observées pour l'ONT₁ comme dans la Table 6.19, mais sur tous les ONTs connectés au PON. Comme on pouvait s'y attendre, l'outil PANDA calcule que la cause racine la plus probable est la rupture de la fibre de transport partagée par tous les ONTs.

L'observation d'une alarme DG (Dying Gasp) pour un ONT, indique que l'alimentation de cet ONT est éteinte ou défectueuse. Le modèle probabiliste détecte correctement cette situation telle que décrite par la Table 6.21. On peut en plus remarquer sur la Table 6.22 que les valeurs les plus probables des paramètres scalaires de l'ONT en question, inférées par le modèle, renforcent la croyance que l'alimentation de cet ONT est éteinte ou défectueuse. En effet, l'état le plus probable du courant de polarisation est I_0 , c'est-à-dire un

courant quasiment nul. La tension d'alimentation est également quasiment nulle avec une croyance de 0.98. La puissance transmise par l'ONT, influencée par son courant de polarisation, est très faible (ou tend vers $-\infty$ en dBm) avec une probabilité de 0.99 ainsi que les puissances reçues qui dépendent de la tension d'alimentation.

TABLE 6.20 – Rupture soudaine de communication entre l'OLT et chacun des ONTs connectés sur un PON de grande dimension.

Causes racines	États	Croyances
$Fiber_T$	[OK, AT, BR]	[0.02, 0.25, 0.73]

Le PON a vingt ONTs. Les alarmes LOS_{us} et LOS_{ds} sont observées pour tous les ONTs, c'est-à-dire que les variables $LOS_{us}[i]$ et $LOS_{ds}[i]$, pour $i \in \{1, \dots, 20\}$, prennent la valeur $+los$. Les compteurs et les paramètres scalaires de tous les ONTs sont inconnus.

TABLE 6.21 – Interruption différée de l'alimentation électrique d'un ONT.

Causes racines	États	Croyances
$Alt_{ONT}[1]$	[OK, \neg OK]	[0.001, 0.999]
$Alt_{ONT}[i \neq 1]$	[OK , \neg OK]	[0.999 , 0.001]

Le PON contient quarante ONTs, l'alarme DG[1] est observée pour l'ONT₁. Les compteurs et les paramètres scalaires de l'ONT₁ sont inconnus tandis que ceux des ONTs voisins sont connus et nominaux.

TABLE 6.22 – Les états probables des variables manquantes pourtant utiles pour détecter une alimentation éteinte ou défectueuse de l'ONT₁ dans la Table 6.21.

Variables manquantes	États	Croyances
$I_{ONT}[1]$	[I₀ , I ₁]	[0.95 , 0.05]
$V_{ONT}[1]$	[V₀ , V ₁]	[0.98 , 0.02]
$Rx_{OLT}[1]$	[Rx₀ , Rx _l , Rx _n]	[0.99 , 2.e-03, 1.e-03]
$Rx_{ONT}[1]$	[Rx₀ , Rx _l , Rx _n]	[0.99 , 4.e-03, 4.e-03]
$Tx_{ONT}[1]$	[Tx₀ , Tx _l , Tx _n]	[0.99 , 3.e-02, 3.e-03]

Il existe aussi des situations sur le réseau dans lesquelles les compteurs et les paramètres scalaires d'un ONT sont nominaux bien que l'alarme DG est observée (voir la Table 6.23). Dans ce cas, le modèle comprend que l'alimentation de cet ONT n'est pas encore éteinte au moment où l'alarme DG est diagnostiquée.

La Table 6.24 présente une situation dans laquelle l'alarme SD est observée pour l'ONT₁ sur les deux canaux de communications. Le modèle probabiliste diagnostique logiquement l'ONT₁ comme étant défectueux. En effet, les puissances reçues $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ relatives à l'ONT₁ sont nominales, ce qui veut dire que la dégradation du signal observée ne peut être due qu'à un dysfonctionnement intrinsèque à l'ONT₁. La Table 6.25 confirme davantage ce résultat. On n'y observe que les alarmes DOW[1] (drift of windows) et TIA[1]

(transmission interference alarm) qui indiquent que l'ONT₁ se comporte anormalement, auraient dues être observées avec de très fortes probabilités.

TABLE 6.23 – Interruption instantanée de l'alimentation électrique d'un ONT.

Causes racines	États	Croyances
$Alt_{ONT}[1]$	[OK , \neg OK]	[0.910 , 0.090]
$Alt_{ONT}[i \neq 1]$	[OK , \neg OK]	[0.999 , 0.001]

Le PON a quarante ONTs, l'alarme DG[1] est observée pour l'ONT₁. Les compteurs et les paramètres scalaires de l'ONT₁ sont nominaux ainsi que ceux des ONTs voisins.

TABLE 6.24 – Dégradation du signal entre l'OLT et un ONT avec des paramètres nominaux.

Causes racines	États	Croyances
FaultyONT[1]	[+fto , \neg fto]	[0.992 , 0.008]
$Fiber_{DB}[i]$	[OK , AT, BR]	[0.967 , 8.e-02, 9.e-07]

Le PON a quarante ONTs, les alarmes $SD_{us}[1]$ et $SD_{ds}[1]$ sont observées pour l'ONT₁. Les compteurs et les paramètres scalaires relatifs à l'ONT₁ sont connus et nominaux, ainsi que ceux des ONTs voisins.

Il faut noter que toutes les instances de cas de diagnostic présentées par chacune des tables ci-dessus ont été inspectées manuellement afin de vérifier leur fiabilité. Les résultats sont cohérents vis-à-vis de l'expertise métier du réseau GPON-FTTH. Cependant, il est difficile et très fastidieux de vérifier manuellement les résultats de tous les 10611 cas de diagnostic. En plus, nous souhaitons quantifier la plus value de l'outil PANDA qui implémente le modèle probabiliste de diagnostic par rapport à l'outil DELC qui implémente une méthode experte de diagnostic à base de règles métiers. L'outil DELC est le système expert à base de règles actuellement en exploitation dans le système de gestion du réseau d'accès GPON-FTTH de l'opérateur Orange.

TABLE 6.25 – Probabilités d'observer les alarmes DOW[1] et TIA[1] relatives à l'ONT₁ et a posteriori aux évidences du cas de la Table 6.24.

Causes racines	États	Croyances
DOW[1]	[+dow , \neg dow]	[0.999 , 0.001]
TIA[1]	[+tia , \neg tia]	[0.973 , 0.027]

6.5 Comparaison entre système expert et modèle probabiliste de diagnostic

Cette section présente analyse et quantifie les performances du modèle probabiliste de diagnostic comparativement au système expert exploité par l'opérateur Orange. Cha-

cun des 10611 cas est diagnostiqué séparément par le système expert DELC en exploitation et par l'outil probabiliste PANDA que nous avons développé. Une matrice de confusion à deux dimensions présentée par la Table 6.26 est utilisée pour comparer les deux approches de diagnostic.

TABLE 6.26 – Matrice de confusion entre le système expert à base de règles de diagnostic DELC et l'outil de diagnostic probabiliste PANDA.

Causes racines	1	2	3	4	5	6	7	8
1 No Default	7393		39			17		
2 Faulty ONT		3						
3 Attenuate drop fiber			72			18		
4 Broken drop fiber				1463				
5 Power Supply shutdown	2				780			
6 Attenuate feeder fiber						0		
7 Broken feeder fiber		1					57	
8 Unknown root cause	716	4	19		27			0

Chaque ligne de la Table 6.26 est labellisée par un résultat de diagnostic de DELC tandis que chacune colonne correspond à un diagnostic de PANDA. Bien que le modèle probabiliste effectue simultanément le diagnostic de tous les ONTs connectés sur un PON, nous avons restreint la comparaison des deux outils, au diagnostic d'un seul ONT. Il s'agit de l'ONT précédemment appelé "ONT testé" ou ONT_1 , c'est-à-dire l'ONT de la l'abonné qui signale un dysfonctionnement sur sa ligne FTTH. Nous sommes contraints de procéder ainsi par ce que le système expert DELC en exploitation, ne s'intéresse qu'au diagnostic de l'ONT dit "ONT testé". Pour chaque cas parmi les 10611, DELC et PANDA effectuent séparément le diagnostic de ce cas sur la base des mêmes observations. Sur la Table 6.26, les nombres en gras placés sur la diagonale de la matrice de confusion représentent les cas pour lesquels le système expert et le modèle probabiliste donnent le même résultat de diagnostic. Parmi les 10611 cases, DELC et PANDA prennent la même décision dans 9768 cas parmi lesquels on note 7393 cas pour lesquels le résultat de diagnostic commun aux deux méthodes est "No Default", c'est-à-dire que chacun des deux outils signale un fonctionnement nominal de l'infrastructure GPON-FTTH.

Pour analyser la comparaison nous nous focalisons sur les cas pour lesquels DELC et PANDA calculent un diagnostic différent. Il s'agit notamment des cas placés hors de la diagonale de la matrice de confusion. Nous énumérons et analysons une instance de chacun de ces cas ci-dessous. L'analyse de chaque cas consiste à inspecter manuellement les observations de ce cas afin de déterminer dans un premier temps, lequel des deux outils présente un résultat de diagnostic plus raisonnable vis-à-vis du fonctionnement du réseau GPON-FTTH. Dans un second temps, l'analyse nous permet également de comprendre pourquoi les diagnostics des deux outils sont parfois différents.

- Nous notons 39 cas pour lesquels le système expert retourne "No Default" tandis que le modèle probabiliste diagnostique une atténuation de la fibre de branchement de l' ONT_1 (ONT testé). L'exploration des observations de ces cas nous montre qu'au moins une des puissances $Rx_{OLT}[1]$ sur le canal montant entre l'OLT et ONT_1 ou $Rx_{ONT}[1]$ sur le canal descendant est faible et l'autre n'est pas observé. Par contre

les puissances des ONTs voisins sont nominales. Aucune règle du système expert DELC ne décrit clairement cette situation, tandis que le modèle probabiliste le fait naturellement. On remarque aussi que lorsque les puissances des ONTs voisins de l'ONT₁ sont aussi faibles, le modèle probabiliste analyse très bien que c'est la fibre de transport partagée par tous les ONTs du PON qui atténue. Cette dernière situation est rencontrée dans 17 cas pour lesquelles DELC ne détecte aucune panne.

- Lorsque les niveaux de puissances de tous les ONTs connectés sur le PON considéré sont tous faibles, le système expert DELC diagnostique une atténuation de la fibre de branchement de l'ONT₁ tandis que PANDA diagnostique une atténuation de la fibre de transport. Bien que ces deux résultats sont différents, ils sont tous les deux raisonnables. En effet, le système expert DELC effectue une corrélation entre les observations des ONTs de l'arbre PON pour lesquels des alarmes ont été déclenchées. L'outil probabiliste PANDA quant à lui effectue une corrélation entre les observations de tous les ONTs connectés sur l'arbre PON même si ces observations ne contiennent pas d'alarmes. Ce qui veut dire que lorsque les niveaux de puissances $R_{X_{OLT}[1]}$ et $R_{X_{ONT}[1]}$ de l'ONT₁ sont faibles et que l'alarme SD (Signal Degraded) relative à la dégradation du signal est observée pour l'ONT₁, mais n'est pas observée pour chacun des ONTs voisins bien que leurs niveaux de puissances soient également faibles, le système expert DELC ne fait pas la corrélation entre les observations relatives à l'ONT₁ et celles relatives aux ONTs voisins. De ce fait, DELC se focalise donc sur les observations de l'ONT₁ et diagnostique une atténuation de sa fibre de branchement. Par contre, le modèle probabiliste, avec sa vision globale de l'arbre PON, c'est-à-dire qu'il corrèle toutes les observations en supposant que l'absence d'une alarme est une donnée manquante et non une observation positive, diagnostique une atténuation de la fibre de transport lorsque les niveaux de puissance de tous les ONTs du PON sont faibles. même si les alarmes relatives à la qualité du signal ne sont pas tous observées.
- Nous notons 2 cas pour lesquelles DELC diagnostique une alimentation éteinte ou défectueuse de l'ONT₁ tandis que PANDA ne détecte aucun défaut. Cette divergence de diagnostic est due au fait que la règle experte de DELC déduit que l'alimentation de l'ONT₁ est éteinte ou défectueuse sur la base de la seule observation de l'alarme DG[1]. Cette règle ne considère pas les situations où l'alarme DG[1] peut être observée, mais les compteurs et les paramètres scalaires de l'ONT₁ sont encore disponibles et nominaux. Le modèle probabiliste, grâce à son analyse globale de la situation, déduit que l'alimentation de l'ONT₁ est encore en service et en bon état au moment où l'alarme DG[1] est diagnostiquée (voir la Table 6.23).
- Nous notons 1 cas pour lequel DELC diagnostique une coupure de la fibre de transport tandis que PANDA calcule un podium de deux causes racines (l'ONT₁ est défectueux ou sa fibre de branchement est rompue) comme sur la Table 6.19. En regardant de près les observations de ce cas, on se rend compte que l'arbre PON ne contient qu'un seul ONT (l'ONT₁) pour lequel une alarme LOS[1] est observée sur les deux canaux, telle que sur la Table 6.19. Le résultat de DELC est dû au fait qu'il a effectué la corrélation entre l'alarme LOS[1] et les alarmes LOS des autres ONTs n'étant pas sur le même arbre PON que l'ONT₁, mais géolocalisés au même endroit que l'ONT₁. Cette différence pour ce cas très rare est donc due au fait que l'ensemble des ONTs pour lesquels les observations sont corrélées n'est pas le même.

-
- Les cas pour lesquels la différence entre les résultats de diagnostic des deux outils est très significative sont ceux pour lesquels le système expert DELC ne parvient pas à produire un diagnostic dû au fait que certaines données ne sont pas observées ou que la situation rencontrée est nouvelle, c'est-à-dire qu'elle n'est pas encore couverte par les règles existantes du système expert. Pour chacun de ces cas, DELC retourne "unknown root cause", c'est-à-dire qu'il est incapable de prendre une décision, tandis que l'outil probabiliste PANDA calcule naturellement le diagnostic le plus probable basé sur les données disponibles même si certaines d'entre elles sont parfois manquantes. Nous notons $716 + 4 + 19 + 27$ cas pour lesquels le système expert DELC échoue par ce que les données observées ne sont pas suffisantes pour déclencher une de ses règles expertes, tandis que le modèle probabiliste PANDA utilise le peu d'observations disponibles pour compléter les données en inférant l'état le plus probable des variables manquantes. Le diagnostic est alors calculé en se basant sur les données complétées. Par exemple, il existe 19 cas pour lesquels DELC ne sait pas diagnostiquer une alarme LEVELLO[1] lorsque les puissances reçues $R_{X_{OLT}[1]}$ et $R_{X_{ONT}[1]}$ ne sont pas mesurées. Pour chacun de ces cas, le modèle probabiliste infère que ces niveaux de puissances sont très probablement faibles et en déduit que l'explication la plus probable de l'alarme LEVELLO[1] est l'atténuation de la fibre de branchement de l'ONT₁.
 - On note aussi 716 cas pour lesquels DELC ne produit aucun diagnostic tandis que PANDA retourne qu'il ne détecte aucune panne sur le réseau GPON-FTTH. Ce résultat peut sembler anodin pour certains alors qu'il présente un intérêt capital pour un opérateur de télécommunications. En effet, ce résultat indique un fonctionnement nominal de l'infrastructure GPON-FTTH et permet d'en déduire que la panne se trouve très certainement dans le réseau domestique de l'abonné qui se plaint. Ceci permet à l'opérateur d'éviter de déclencher des interventions inutiles sur l'infrastructure GPON-FTTH mais d'intervenir éventuellement plutôt chez l'abonné. Ce résultat de PANDA permet ainsi de réduire le nombre d'interventions et de gagner du temps et de l'argent au service après-vente de l'opérateur.

En somme, nous pouvons dire que les résultats de diagnostic du réseau d'accès GPON-FTTH du modèle probabiliste PANDA sont plus cohérents avec l'expertise métier, et plus raisonnables sur un échantillon de cas représentatifs, que le système expert à base de règles DELC. Nous notons que pour 7% des cas, le système expert est incapable de calculer un diagnostic dû à l'absence de certaines données, tandis que le modèle probabiliste calcule toujours le diagnostic le plus probable consistant avec les observations disponibles. La principale explication de ces différences est due au fait que, contrairement au système expert, le modèle probabiliste intègre une capacité de raisonnement très étendu et diversifié qui lui permet de faire des analyses globales ou généralisées des situations.

6.6 Discussion et conclusion

Nous avons construit et présenté dans ce chapitre un modèle de la topologie et de la propagation de pannes et d'alarmes sur un PON du réseau d'accès optique GPON-FTTH. Le modèle dit "initial" inclut deux des trois blocs fonctionnels d'un PON, à savoir l'OLT et les ONTs. Le modèle est basé sur le formalisme des réseaux bayésiens, et est construit selon la méthode générique développée dans le Chapitre 5. Par conséquent, il hérite des

propriétés de modularité et d'extensibilité qui nous permettront un peu plus tard dans cette thèse d'y ajouter le modèle du troisième bloc fonctionnel, le bloc fonctionnel ODN (Optical Distribution Network). Les paramètres du modèle initial ainsi construit ont été approximativement déterminés en se basant sur les connaissances métiers relatives au comportement d'un PON du réseau GPON-FTTH. Un raisonnement structuré et cohérent avec ce comportement est présenté dans ce chapitre.

Le modèle initial est ensuite doublement vérifié afin de s'assurer qu'il est bien cohérent avec l'expertise métier du réseau GPON-FTTH. Dans un premier temps, la vérification est faite avec des cas de diagnostic simples émulsés sur la plateforme expérimentale du réseau GPON-FTTH d'Orange Labs. Ceci nous a permis de vérifier l'adéquation de notre approche à base d'un modèle probabiliste avec le diagnostic automatique du réseau GPON-FTTH. Dans un second temps, le modèle probabiliste a été confronté à la réalité. Des cas réels de diagnostic sur le réseau GPON-FTTH opérationnel de l'opérateur Orange ont été traités. Les résultats de diagnostic du modèle probabiliste ont été comparés avec ceux du système à base de règles expertes de diagnostic actuellement en exploitation dans le NMS (Network Management System) d'Orange. L'analyse comparative entre les résultats de diagnostic de l'outil probabiliste PANDA et ceux du système expert DELC montre que le modèle probabiliste est plus cohérent avec l'expertise métier, et plus raisonnable sur un échantillon de cas de diagnostic représentatifs que le système expert DELC.

Chapitre 7

Ajustement automatique des paramètres du modèle GPON-FTTH : application de l'algorithme EM

*« I am enough of an artist to draw
freely upon my imagination.
Imagination is more important than
knowledge. Knowledge is limited.
Imagination encircles the world. »*

Albert Einstein, 1929

Sommaire

7.1 Introduction	119
7.2 Contexte applicatif de l'algorithme EM	119
7.2.1 Contexte relatif au réseau GPON-FTTH	119
7.2.2 L'échantillon de données réelles	120
7.3 L'EM avec les données réelles d'apprentissage	120
7.3.1 Vérification de la cohérence des données réelles	121
7.3.2 Implémentation de l'EM : difficultés rencontrées et solutions	121
7.3.3 Convergence de l'algorithme EM	123
7.4 Valeur ajoutée de l'EM pour le diagnostic du réseau d'accès GPON-FTTH	123
7.5 Discussion et conclusion	127

7.1 Introduction

La modélisation du comportement d'un réseau est un problème central pour le développement d'une méthode de diagnostic à base d'un modèle de ce réseau. Il existe deux approches principales pour concevoir un tel modèle. Le modèle peut être construit à partir des connaissances "expertes" acquises par exemple des standards de l'ITU-T de ce réseau. Le modèle peut être automatiquement appris en développant ou en appliquant un algorithme de découverte et d'extraction de connaissances, à partir des données générées par les équipements du réseau. Dans le Chapitre 6, nous avons construit un modèle de la topologie et de la propagation de pannes et d'alarmes sur le réseau d'accès GPON-FTTH. Le développement de ce modèle s'est fait essentiellement sur la base des connaissances métiers du réseau GPON-FTTH. Ce modèle est basé sur le formalisme d'un réseau bayésien qui encode les connaissances métiers. Il s'agit notamment des dépendances qui encodent la propagation de pannes et d'alarmes ainsi que des distributions de probabilités conditionnelles qui quantifient la force des dépendances. Les résultats de diagnostic d'un réseau GPON-FTTH opérationnel effectué avec ce modèle sont très satisfaisants. Nous les avons présentés et analysés dans le Chapitre 6. Néanmoins, nous pensons que ces résultats peuvent être améliorés si les paramètres du modèle, c'est-à-dire les distributions de probabilités conditionnelles, sont automatiquement ajustés ou affinés par un algorithme d'apprentissage artificiel, à partir des données générées par les équipements du réseau GPON-FTTH. Ceci est un problème d'estimation paramétrique d'un modèle statistique à partir d'un échantillon de données incomplètes, que nous pouvons résoudre avec l'algorithme EM (Expectation Maximization). L'algorithme EM a été étudié dans le Chapitre 4. Le chapitre courant se focalise donc essentiellement sur son application avec un échantillon de données réelles d'un réseau GPON-FTTH opérationnel. La Section 2 de ce chapitre précise justement le contexte applicatif de l'algorithme EM. Son implémentation est présentée dans la Section 3 tout en discutant des difficultés rencontrées relatives à la manipulation des données réelles et des solutions proposées. Dans la Section 4 nous évaluons l'apport de l'EM pour le diagnostic du réseau GPON-FTTH. La Section 5 conclut ce chapitre et ouvre plusieurs perspectives.

7.2 Contexte applicatif de l'algorithme EM

Cette section décrit les conditions dans lesquelles l'algorithme EM est appliqué pour ajuster les paramètres du modèle du réseau GPON-FTTH présenté par la Figure 6.2.

7.2.1 Contexte relatif au réseau GPON-FTTH

Comme nous l'avons dit dans le Chapitre 6, la Figure 6.2 présente le modèle initial d'un PON du réseau GPON-FTTH. Un PON peut contenir jusqu'à 64 ONTs dans notre cas. Les PONs d'un réseau GPON n'interagissent pas et tous les PONs ont le même comportement, c'est-à-dire que les pannes et les alarmes sont identiques et se propagent de façon identique, quel que soit le PON considéré. Par ailleurs, même si les fabricants des ONTs d'un même PON peuvent être différents, tous ces ONTs ont le même fonctionnement puisqu'ils implémentent chacun les mêmes standards de l'ITU-T [TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU \[2008\]](#) [TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU \[2003\]](#). Par conséquent, nous avons choisi d'appliquer l'EM sur le modèle probabiliste d'un PON contenant un seul ONT. Les paramètres de cet ONT peuvent être généralisés à tous les autres ONTs. Par exemple, pour deux quelconques ONTs i et $j \neq i$

connectés à un PON, la probabilité $\mathbb{P}(\text{LOS}_{ds}[i]|\text{Rx}_{\text{ONT}}[i])$ d'observer une perte de communication entre l'OLT et l'ONT_{*i*} sachant le niveau de puissance reçue à l'ONT_{*i*} est la même pour l'ONT_{*j*}. Dans le Chapitre 4, nous avons expliqué que l'algorithme EM doit être initialisé avec une valeur θ^0 du vecteur de paramètres, choisie de façon non aléatoire pour espérer éviter que l'algorithme reste coincé sur un maximum local de la fonction de vraisemblance des observations. Pour notre problème, θ^0 est la valeur du vecteur de paramètres déterminée approximativement dans le Chapitre 6, sur la base des connaissances métiers relatives à la propagation de pannes et d'alarmes sur un PON du réseau GPON-FTTH. En procédant ainsi, nous sommes certains d'améliorer ces paramètres "experts" du modèle. En effet, la propriété de stabilité numérique de l'EM formalisée par l'Équation (4.14) garantit que l'algorithme converge vers une valeur du vecteur de paramètres meilleur que θ^0 sous le critère de la vraisemblance de l'échantillon de données d'apprentissage. Il convient donc de vérifier la pertinence et la cohérence de cet échantillon pour s'assurer que les paramètres affinés par apprentissage artificiel ne s'écartent pas de la réalité au lieu de s'en approcher comme nous le souhaitons.

7.2.2 L'échantillon de données réelles

L'algorithme EM est appliqué sur un échantillon de données réelles générées par les équipements du réseau GPON-FTTH opérationnel de l'opérateur Orange. L'échantillon est une table de 10611 lignes de données. Chaque ligne correspond à un cas à diagnostiquer, c'est-à-dire une configuration ou une combinaison de valeurs prises par certaines variables du modèle du réseau GPON-FTTH présenté à la Figure 6.2. L'échantillon contient les alarmes, les compteurs d'erreurs de transmission, les puissances transmises et reçues, les courants de polarisation, les tensions d'alimentation, les températures des OLTs et des ONTs du réseau. Ces données correspondent aux mesures effectuées sur le réseau GPON-FTTH opérationnel pendant les mois de juillet et août 2015. Cependant, toutes les variables du modèle représenté par la Figure 6.2 ne sont pas mesurées. Il existe des situations où le système de gestion du réseau ne parvient pas récupérer certaines mesures sur certains équipements. Ces situations sont dues par exemple à la politique de filtrage des données appliquée par l'opérateur, la rupture momentanée de la communication entre le système de gestion du réseau et un ou plusieurs équipements, ou dues à certains vieux équipements qui ne génèrent pas certains faits. Ces situations conduisent à des variables manquantes ou non observées. C'est pourquoi l'algorithme EM est utilisé pour faire face à ces données manquantes.

Nous avons divisé les données mesurées en deux sous-ensembles. Le premier sous-ensemble correspond à 5121 cas de diagnostic ou de mesures effectuées sur le réseau GPON-FTTH opérationnel durant le mois de juillet 2015. Le second sous-ensemble correspond à 5490 cas de diagnostic collectés durant le mois d'août 2015. Le premier sous-ensemble est l'échantillon de données d'apprentissage utilisé pour ajuster artificiellement les paramètres "experts" du modèle initial du réseau GPON-FTTH. Le second sous-ensemble correspond aux données de test que nous utilisons pour évaluer les performances de diagnostic du modèle ajusté.

7.3 L'EM avec les données réelles d'apprentissage

Cette section présente l'application de l'EM selon le contexte défini ci-dessus. Nous commençons tout d'abord par vérifier la cohérence de l'échantillon de données réelles.

7.3.1 Vérification de la cohérence des données réelles

Soit D l'échantillon de données d'apprentissage et $d \in D$ une ligne ou un cas de diagnostic de la table D . Nous voulons vérifier que chaque cas d n'est pas très invraisemblable sous le modèle du réseau GPON-FTTH de paramètres θ^0 . En effet, la présence de nombreux cas aberrants dans l'échantillon de données d'apprentissage peut faire dériver le modèle, c'est-à-dire que les paramètres ajustés automatiquement ne refléteront pas le comportement de la propagation de pannes et d'alarmes sur le réseau GPON-FTTH. Le modèle s'écartera dans ce cas de la réalité puisque ses paramètres auront été ajustés à partir d'un échantillon de données aberrant.

D'après l'Équation (4.23), la vraisemblance $\mathbb{P}(d; \theta^0)$ d'une configuration d sous le modèle de paramètres θ^0 se calcule par la formule :

$$\mathbb{P}(d; \theta^0) = \sum_C \phi_C^*(\theta^0),$$

ou $\phi_C^*(\theta^0)$ est le potentiel actualisé d'une quelconque clique C , par l'algorithme de propagation des évidences d sur un arbre de jonction équivalent au modèle de paramètres θ^0 . Cet algorithme a été étudié dans le Chapitre 3. La Figure 7.1 présente l'histogramme des logarithmes des vraisemblances des cas de l'échantillon de données d'apprentissage. On peut y remarquer qu'il n'existe aucun cas pour lequel la log-vraisemblance s'écarte considérablement de la médiane de la série statistique des log-vraisemblances. Le minimum et le maximum de la série sont presque équidistants de la médiane. Ce qui signifie que l'échantillon d'apprentissage ne contient pas de données aberrantes sous le modèle de paramètres θ^0 . Nous pouvons donc utiliser cet échantillon pour ajuster les paramètres du modèle initial sans risque d'apprendre des paramètres complètement dénués de sens vis-à-vis du fonctionnement du réseau GPON-FTTH.

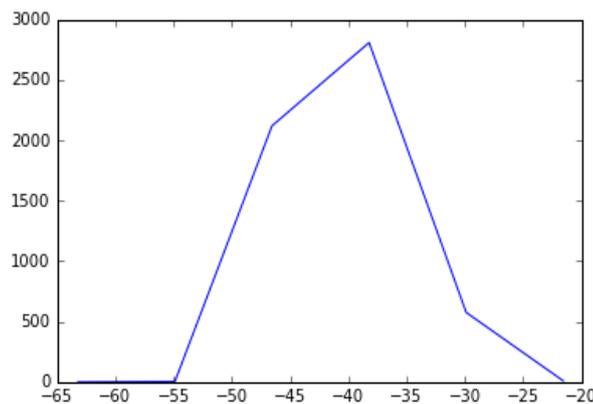


FIGURE 7.1 – Histogramme des log-vraisemblances des cas de l'échantillon de données.

7.3.2 Implémentation de l'EM : difficultés rencontrées et solutions

Nous avons codé une implémentation en Python de l'algorithme EM. Les calculs nécessaires pour évaluer les matrices appelées "expected counts" dans l'étape E de l'EM, sont basés sur l'algorithme d'inférence sur un arbre de jonction équivalent au modèle GPON-FTTH, étudié dans le Chapitre 3. D'un point de vue pratique, nous mentionnons

ci-dessous quelques difficultés rencontrées durant la mise en oeuvre de l'EM avec les données réelles d'apprentissage générées par le réseau GPON-FTTH opérationnel d'Orange :

- Des problèmes numériques dans l'étape E : l'Équation (3.23) qui calcule le potentiel initial d'une clique de l'arbre de jonction est un produit de distributions de probabilités conditionnelles, c'est-à-dire un produit de petites quantités comprises dans l'intervalle]0, 1[. Ce qui signifie que les potentiels initiaux sont des matrices de plusieurs dimensions qui contiennent des éléments de très petites valeurs. D'autre part, le calcul des "expected counts" est basé sur l'Équation (3.24) qui multiplie et inverse plusieurs fois les potentiels des cliques adjacentes pour les actualiser. Durant l'inférence sur l'arbre de jonction, on se retrouve parfois avec des potentiels contenant des nombres à virgule flottante tellement petits ou tellement grands qu'ils ne peuvent plus être représentés sous le format double précision à 64 bits de la norme IEEE 754. On obtient dans ce cas des NaN (Not a Number) pour les nombres très petits ou des Inf (Infini) pour les nombres très grands, ce qui interrompt les calculs dans l'étape E de l'itération courante. Pour contourner ce problème, à chaque fois que la valeur d'un élément d'un potentiel est NaN, nous la remplaçons par le plus petit nombre positif normalisé différent de zéro, c'est-à-dire le nombre $2^{-1022} \approx +2.2250738585072014e - 309$. De même chaque valeur Inf est remplacée par le plus grand nombre positif fini représentable sur 64 bits, c'est-à-dire $(2^{1024} - 2^{971}) \approx 1.7976931348623157e + 308$.
- Des problèmes numériques dans l'étape M : l'Équation (4.27) qui donne la solution du problème de maximisation sous contrainte (4.26) est une fraction. Le numérateur est la matrice des "expected counts" $\hat{N}_{i,j,k}$ qui estime le nombre d'occurrences de la combinaison des événements $X_i = k$ et $pa(X_i) = j$ dans l'échantillon de données d'apprentissage, c'est-à-dire le nombre d'occurrences qu'une variable X_i prenne la valeur k lorsque ses parents $pa(X_i)$ sont dans la configuration j . Le dénominateur est la quantité $\hat{N}_{i,j}$ qui donne une estimation du nombre d'occurrences de l'évènement $pa(X_i) = j$ dans l'échantillon de données. Lors de la mise à jour de la loi conditionnelle de la variable X_i , on se retrouve parfois avec une valeur nulle de $\hat{N}_{i,j,k}$ lorsque $\hat{N}_{i,j} \neq 0$. Ce qui veut dire que la combinaison des événements $X_i = k$ et $pa(X_i) = j$ n'est probablement jamais observée dans les données tandis que l'évènement $pa(X_i) = j$ l'est. Dans ce cas la probabilité conditionnelle $\mathbb{P}(X_i = k | pa(X_i) = j)$ est nulle. Ce résultat engendre, dans l'étape E de la prochaine itération, certaines valeurs nulles dans les potentiels initiaux des cliques contenant la variable X_i , causant ainsi des problèmes de division par zéro lorsque certains potentiels sont inversés lors de leur mise à jour (Équation 3.24). L'inférence dans l'étape E s'interrompt dans ce cas. Le problème numérique de division par zéro est aussi souvent rencontré dans l'étape M de l'itération courante, lorsque le dénominateur $\hat{N}_{i,j}$ est nul.
- Pour contourner les problèmes numériques de division par zéro, la première solution que nous avons testée est d'éviter de mettre à jour la probabilité conditionnelle $\mathbb{P}(X_i = k | pa(X_i) = j)$ lorsque $\hat{N}_{i,j,k} = 0$ ou $\hat{N}_{i,j} = 0$ à l'itération courante, et de renormaliser ensuite le vecteur $\mathbb{P}(X_i | pa(X_i) = j)$. Mais cette approche ne fonctionne pas bien qu'elle permet d'éviter les divisions par zéro. En effet, les valeurs des autres probabilités conditionnelles $\mathbb{P}(X_i = h | pa(X_i) = j)$ telles que $h \neq k$, sont aussi modifiées. Ceci fausse complètement les résultats de l'algorithme EM. La seconde solution testée est d'éviter de mettre à jour le vecteur $\mathbb{P}(X_i | pa(X_i) = j)$ à l'itération

courante. Les proportions $\mathbb{P}(X_i = k | pa(X_i) = j, \forall k)$, à l'itération courante sont donc identiques à celles de l'itération précédente. Cette seconde approche fausse également les résultats de l'EM puisque la loi conditionnelle $\mathbb{P}(X_i | pa(X_i))$ est partiellement actualisée à l'itération courante. La troisième approche qui est la bonne est de tout simplement considérer que la loi conditionnelle $\mathbb{P}(X_i | pa(X_i))$ ne fait pas partir des paramètres à actualiser à l'itération courante. La valeur de ce paramètre à l'itération courante est la même qu'à l'itération précédente, mais ce paramètre est normalement actualisé à la prochaine itération pour laquelle les conditions $\hat{N}_{i,j,k} \neq 0$ et $\hat{N}_{i,j} \neq 0$ sont vérifiées.

7.3.3 Convergence de l'algorithme EM

La Figure 7.2 affiche l'évolution de la log-vraisemblance des données réelles d'apprentissage, en fonction des itérations de l'algorithme EM. Comme définit par l'Équation (4.14), on peut remarquer que la log-vraisemblance des données mesurées sur le réseau GPON-FTTH augmente après chaque itération et se stabilise au bout de 7 ou 8 itérations. L'algorithme EM converge donc bien avec les solutions proposées pour contourner les problèmes numériques dans la sous-section précédente.

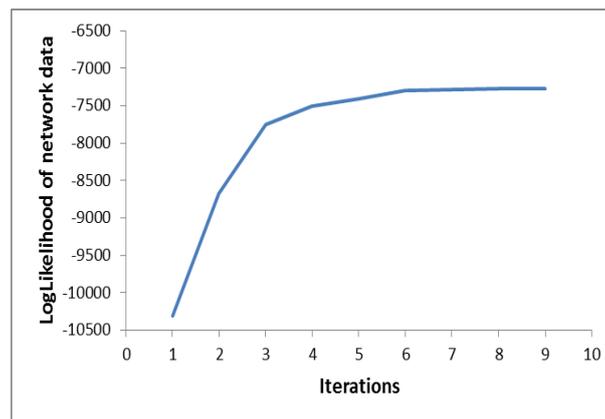


FIGURE 7.2 – L'évolution de la log-vraisemblance des données mesurées sur un réseau GPON-FTTH opérationnel avec les itérations.

La propriété de stabilité numérique formalisée par l'Équation (4.14), nous garantit que les paramètres θ^c obtenus après la convergence de l'EM sont meilleurs que les paramètres "experts" initiaux θ^0 sous le critère de la vraisemblance aux données. A ce stade de nos travaux, on se retrouve donc avec deux modèles du réseau GPON-FTTH. Le modèle initial "expert" M_0 de paramètres θ^0 et le modèle M_c de paramètres θ^c . Il est question de vérifier dans la section suivante que le modèle M_c est plus performant que le modèle M_0 pour le diagnostic du réseau GPON-FTTH. En effet, on s'attend à ce que le modèle M_c soit plus performant que M_0 puisque les paramètres θ^c reflètent mieux le comportement du réseau GPON-FTTH que les paramètres θ^0 .

7.4 Valeur ajoutée de l'EM pour le diagnostic du réseau d'accès GPON-FTTH

Nous utilisons les données de test contenant 5490 cas de diagnostic pour évaluer la plus-value de l'algorithme EM pour le diagnostic du réseau GPON-FTTH. Pour cela, cha-

un de ces cas est diagnostiqué séparément par chacun des modèles M_0 et M_c . La Table 7.1 présente une matrice de confusion à deux dimensions qui compare les résultats de diagnostic des deux modèles. Les lignes et les colonnes de cette matrice représentent respectivement les diagnostics effectués avec le modèle initial M_0 et avec le modèle M_c . On note que l'ajustement automatique des paramètres θ^0 pour obtenir les paramètres θ^c du modèle M_c , change les résultats de diagnostic de 185 cas parmi les 5490 cas de test, soit environ 3.37%. Ce qui signifie que la détermination approximative des paramètres initiaux θ^0 faite dans le Chapitre 6 est raisonnablement correcte. Cependant, il convient d'analyser dans les détails les cas pour lesquels les diagnostics des deux modèles sont différents. Il s'agit notamment des cas correspondant aux valeurs non nulles placées hors de la diagonale de la matrice de confusion. Chaque table de 7.2 à 7.7 compare les deux modèles pour une instance de chacun de ces cas. Le titre de ces tables est une brève description des observations collectées sur le réseau GPON-FTTH opérationnel pour le cas considéré. Ces observations sont détaillées en dessous de chaque table.

TABLE 7.1 – Matrice de confusion à deux dimensions pour la comparaison des résultats de diagnostic des deux modèles.

Causes racines	1	2	3	4	5	6	7
1 No Default	4030			7	6		9
2 Configuration Mistake		183					
3 Faulty ONT			0				
4 ONT power supply				402			
5 Drop fiber attenuated					56		
6 Drop fiber broken					14	602	1
7 Feeder fiber attenuated			148				32

TABLE 7.2 – Affaiblissement du signal sur le canal montant entre l'ONT₁ et l'OLT.

Modèle	Causes racines	États	Croyances
Expert M_0	$Fiber_{DB_1}$	[OK, AT, BR]	[0.9 , 8.e-02, 3.e-06]
	$Fiber_{DB_{i \neq 1}}$	[OK, AT, BR]	[0.9 , 8.e-02, 3.e-06]
Appris M_c	$Fiber_{DB_1}$	[OK, AT, BR]	[9.e-02, 0.9 , 2.e-06]
	$Fiber_{DB_{i \neq 1}}$	[OK, AT, BR]	[0.9 , 8.e-02, 3.e-06]

Le PON contient 40 ONTs et aucune alarme n'est observée. La puissance reçue $Rx_{OLT}[1]$ de l'ONT₁ à l'OLT est faible. La puissance $Rx_{ONT}[1]$ reçue à l'ONT₁ est nominale. Les compteurs et les paramètres scalaires des ONTs voisins sont nominaux.

La Table 7.2 montre que, sur un PON de 40 ONTs, lorsque la puissance reçue à l'OLT de l'ONT numéro 1 (ONT₁), sur le canal montant est faible, tandis que la puissance reçue à l'ONT₁ sur le canal descendant est nominale, le modèle expert M_0 ne détecte aucune panne. Ceci est un diagnostic incorrect puisque le canal montant entre l'OLT et l'ONT₁ introduit une atténuation trop importante du signal optique. Par contre, le modèle appris artificiellement M_c calcule le diagnostic correct, c'est-à-dire l'atténuation trop importante de la fibre de branchement $Fiber_{DB_1}$ de l'ONT₁. Nous rappelons que les notations OK, AT et BR désignent respectivement l'état d'une fibre optique en bon état, qui

TABLE 7.3 – Affaiblissement du signal entre l’OLT et chacun des ONTs voisins de l’ONT₁.

Modèle	Causes racines	États	Croyances
Expert M ₀	<i>Fiber</i> _{DB₁}	[OK, AT, BR]	[0.9 , 8.e-02, 3.e-06]
	<i>Fiber</i> _{DB_{i≠1}}	[OK, AT , BR]	[8.e-02, 0.9 , 3.e-06]
Appris M _c	<i>Fiber</i> _{DB₁}	[OK, AT, BR]	[0.9 , 9.e-02, 2.e-06]
	<i>Fiber</i> _T	[OK, AT , BR]	[5.e-03, 0.99 , 5.e-39]
	<i>Fiber</i> _{DB_{i≠1}}	[OK , AT, BR]	[0.68 , 0.31, 0.001]

Le PON contient 40 ONTs et aucune alarme n’est observée. Les puissances reçues $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ sont nominales. Les puissances reçues $Rx_{OLT}[i]$ et $Rx_{ONT}[i]$, pour $i \in \{2, \dots, 40\}$, des ONTs voisins sont faibles.

atténuée trop ou alors qui est rompue. Cette situation se produit 6 fois dans l’échantillon de test. La Table 7.3 présente une situation dans laquelle les niveaux de puissances reçues de l’ONT₁ sont nominaux tandis que ceux des ONTs voisins de l’ONT₁ sont faibles. Dans cette situation, le modèle expert M₀ diagnostique que la fibre de branchement de chaque ONT voisin souffre d’atténuation. Le modèle M₀ assume donc que lorsque les niveaux de puissances d’au moins un ONT sur un PON sont nominaux, alors la fibre de transport *Fiber*_T partagée par tous les ONTs connectés sur ce PON ne peut pas souffrir d’atténuation, bien que les puissances reçues des autres ONTs sont faibles. Ce raisonnement n’est pas absolument vrai puisque pour ce cas de diagnostic, c’est justement la fibre de transport qui atténue. Mais cette atténuation n’a pas affectée l’ONT₁ puisque ses niveaux de puissances reçues étaient très élevés avant le début de l’atténuation de la fibre de transport. Par contre les niveaux de puissances des ONTs voisins de l’ONT₁ étaient nominaux, mais très proches de la borne inférieure de la plage des puissances nominales. Nous avons rencontré 9 occurrences de ce cas dans les données de test.

TABLE 7.4 – Transmission à très faible puissance de l’ONT₁.

Modèle	Causes racines	États	Croyances
Expert M ₀	<i>Alt</i> ONT ₁	[OK, ¬OK]	[0.90 , 0.10]
Appris M _c	<i>Alt</i> ONT ₁	[OK, ¬ OK]	[0.05, 0.95]

Le PON contient 20 ONTs et aucune alarme n’est observée. Les puissances reçues $Rx_{OLT}[i]$ et $Rx_{ONT}[i]$, pour $i \in \{1, \dots, 20\}$, sont nominales. La puissance transmise $Tx_{ONT}[1]$ de l’ONT₁ est très faible.

La Table 7.4 illustre un cas pour lequel le modèle appris M_c diagnostique que l’alimentation de l’ONT₁ désignée par la variable binaire *Alt*ONT₁, est défectueuse puisque la puissance transmise par cet ONT est très faible tandis que ses niveaux de puissances reçues sont nominales. Le modèle expert quant à lui ne comprend pas cette situation étrange et diagnostique qu’il n’a aucun défaut.

La Table 7.5 illustre une situation dans laquelle l’alarme LOF[1] (Loss of Frame) est observée pour l’ONT₁ et les puissances reçues de cet ONT n’ont pas pu être mesurées par le système de gestion du réseau GPON-FTTH. Le modèle expert diagnostique que la fibre de branchement de l’ONT₁ est rompue. Ceci est un diagnostic incorrect puisque la perte

TABLE 7.5 – Perte de trames de données entre l’OLT et l’ONT₁.

Modèle	Causes racines	États	Croyances
Expert M ₀	$Fiber_{DB_1}$	[OK, AT, BR]	[0.14, 0.34, 0.52]
	$Fiber_{DB_{i \neq 1}}$	[OK , AT, BR]	[0.9 , 8.e-02, 2.e-06]
Appris M _c	$Fiber_{DB_1}$	[OK, AT , BR]	[0.08, 0.86 , 0.06]
	$Fiber_{DB_{i \neq 1}}$	[OK , AT, BR]	[0.9 , 8.e-02, 3.e-06]

Le PON contient 40 ONTs. Les alarmes $LOF_{us}[1]$ et $LOF_{ds}[1]$ sont observées pour l’ONT₁. Les puissances $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ de l’ONT₁ ne sont pas mesurées tandis que celles des ONTs voisins sont nominales.

de trames entre l’OLT et l’ONT₁ n’est pas due à une coupure de la fibre, mais plutôt à la mauvaise qualité du signal transmis sur cette fibre, c’est-à-dire l’atténuation importante de cette fibre. Le modèle appris calcule le diagnostic approprié. Cette situation apparaît 14 fois dans l’échantillon de données de test.

TABLE 7.6 – Rupture soudaine de la communication sur un PON à un seul ONT.

Modèle	Causes racines	États	Croyances
Expert M ₀	$Faulty_{ONT_1}$	[+fto , $\neg fto$]	[0.532 , 0.468]
	$Fiber_T$	[OK, AT , BR]	[0.28, 0.43 , 0.29]
	$Fiber_{DB_1}$	[OK, AT, BR]	[0.24, 0.35, 0.41]
Appris M _c	$Faulty_{ONT_1}$	[+fto , $\neg fto$]	[0.960 , 0.040]
	$Fiber_{DB_1}$	[OK, AT, BR]	[0.33, 0.36, 0.31]
	$Fiber_T$	[OK, AT, BR]	[0.33, 0.34, 0.33]

Le PON contient un seul ONT. Les alarmes $LOS_{us}[1]$ et $LOS_{ds}[1]$ sont observées pour l’ONT₁. Les puissances reçues $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ ne sont pas mesurées.

La Table 7.6 présente une situation de perte de communication entre l’OLT et l’unique ONT connecté sur le PON. Dans cette situation, aucune donnée relative aux ONTs voisins de l’ONT₁ n’est disponible pour aider le modèle expert à discriminer entre trois causes racines potentielles, l’atténuation de la fibre de transport du PON, la coupure de la fibre de branchement de l’ONT₁ ou un dysfonctionnement intrinsèque à l’ONT₁. Le modèle appris diagnostique que l’ONT₁ est défectueux, mais en pratique c’est impossible de discriminer entre les trois causes racines potentielles même pour un expert avéré du réseau GPON-FTTH. Il faut très souvent une intervention d’un technicien qui effectuera des tests sur l’infrastructure GPON-FTTH afin d’identifier la cause racine. Cette situation est observée 148 fois dans l’échantillon de données de test. Dans la Table 7.7, l’ONT₁ n’est plus seul sur le PON. Trente-neuf autres ONTs sont également connectés sur le PON et l’alarme SD relative à la dégradation du signal est observée pour chacun des ces trente-neuf ONTs. Dans ce cas, le modèle expert M₀ diagnostique une coupure de la fibre de transport du PON tandis que le modèle appris M_c diagnostique plutôt une atténuation importante de cette fibre. Le diagnostic du modèle M_c est plus approprié aux observations, puisque les alarmes de perte de signal LOS et de dégradation de signal SD peuvent être dues à une atténuation trop importante de la fibre de transport. Par contre, une coupure de cette fibre

TABLE 7.7 – Rupture soudaine de la communication entre l’OLT et l’ONT₁ et dégradation du signal entre l’OLT et chacun des ONTs voisins.

Modèle	Causes racines	États	Croyances
Expert M ₀	<i>Fiber</i> _T	[OK, AT, BR]	[0.01, 0.18, 0.81]
	<i>Faulty</i> ONT ₁	[+ <i>f to</i> , ¬ <i>f to</i>]	[0.46, 0.54]
Appris M _c	<i>Fiber</i> _T	[OK, AT , BR]	[3.e-42, 0.998 , 1.e-03]
	<i>Faulty</i> ONT ₁	[+f to , ¬ <i>f to</i>]	[0.932 , 0.062]

Le PON contient quarante ONTs. Les alarmes LOS_{us}[1] et LOS_{ds}[1] sont observées pour l’ONT₁. Les puissances reçues Rx_{OLT}[1] et Rx_{ONT}[1] ne sont pas mesurées. Les alarmes SD_{us}[*i*] et SD_{ds}[*i*], pour $i \in \{2, \dots, 40\}$, sont observées pour les ONTs voisins.

aurait engendré une alarme LOS pour l’ONT₁ ainsi que pour tous ses ONTs voisins. Une seule occurrence de ce cas est observée dans l’échantillon de données de test.

7.5 Discussion et conclusion

Ce chapitre présente une mise en oeuvre de l’algorithme EM (Expectation Maximization) étudié dans le Chapitre 4. Une implémentation en Python de cet algorithme est développée et appliquée pour ajuster automatiquement les paramètres du modèle initial du réseau GPON-FTTH proposé dans le Chapitre 6. En effet, les paramètres de ce modèle ont été approximativement déterminés dans le Chapitre 6, à partir des connaissances métiers dites "expertes" relatives au fonctionnement du réseau GPON-FTTH. Notre implémentation de l’EM utilise ces paramètres "experts" comme point d’initialisation et les ajuste artificiellement en recherchant de façon itérative d’autres paramètres qui sont plus vraisemblables aux données réelles d’un réseau GPON-FTTH opérationnel que les paramètres "experts" initiaux. Les données réelles mesurées sur le réseau GPON-FTTH ont été divisées en deux parties. Un premier échantillon de 5121 cas de diagnostic utilisé pour ajuster automatiquement les paramètres "experts" initiaux et obtenir de nouveaux paramètres dits "appris", et un second échantillon de 5490 cas utilisé pour évaluer les performances de diagnostic du modèle appris comparativement au modèle expert. Une analyse détaillée des cas pour lesquels les diagnostics des deux modèles divergent est faite. Comme l’on pouvait s’y attendre, les résultats de diagnostic du modèle appris sont raisonnablement meilleurs que ceux du modèle expert.

Une perspective intéressante de ce travail serait d’ajuster automatiquement le modèle expert dans sa totalité (dépendances et paramètres) avec pour objectif d’améliorer davantage les résultats de diagnostic. Un algorithme tel que SEM (Structural EM) pourrait être appliqué dans ce cas. Une application concrète des travaux de ce chapitre pour un opérateur de télécommunications serait par exemple de mettre en oeuvre un système de diagnostic complètement autonome dans lequel une boucle fermée serait créée entre le modèle probabiliste de diagnostic intégrant des capacités d’apprentissage artificiel, les diagnostics donnant lieu à des interventions sur le réseau et les données remontées par les techniciens d’interventions qui serviraient à ajuster automatiquement le modèle pour le rendre plus performant. Cependant, la fiabilité de ces données doit être garantie pour éviter de faire plutôt dériver le modèle au lieu de l’améliorer.

Chapitre 8

Extension du modèle GPON-FTTH : prise en compte du réseau de distribution optique

« The optical distribution network in the PON context, is a tree of optical fibers in the access network, supplemented with power or wavelength splitters, filters or other passive optical devices. »

ITU-T G.984.2, 2003

Sommaire

8.1 Introduction	129
8.2 Diversité des ingénieries de l'ODN (Optical Distribution Network)	129
8.3 Modélisation automatique de l'ODN	130
8.3.1 Modèle générique de l'ODN et algorithme d'instanciation	131
8.3.2 Le premier niveau de couplage	132
8.3.3 Les niveaux de couplage intermédiaires	132
8.3.4 Le dernier niveau de couplage	133
8.4 Modèle complet du réseau GPON-FTTH	133
8.5 Valeur ajoutée du modèle de l'ODN pour le diagnostic GPON-FTTH . . .	134
8.6 Discussion et conclusion	137

8.1 Introduction

Le réseau d'accès GPON-FTTH est composé de trois blocs fonctionnels. Dans le Chapitre 6 nous avons construit un modèle de propagation de pannes et d'alarmes sur un PON de ce réseau. Ce modèle dit "initial" que nous désignons M_0 , n'est constitué que de deux blocs fonctionnels : le composant OLT et le composant ONT. La structure du réseau de distribution optique (ODN) qui constitue le troisième bloc fonctionnel physiquement placé entre les composants OLT et ONT, n'est pas pris en compte dans le modèle initial M_0 . Le modèle M_0 effectue donc un diagnostic de bout en bout puisqu'il considère l'ODN comme transparent, car essentiellement constitué d'équipements passifs ne générant pas d'alarmes. Cependant, une panne d'un équipement passif dans le réseau de distribution peut être la cause racine d'une ou de plusieurs anomalies observées au niveau du composant OLT ou ONT. Il convient donc de corréliser les trois composants d'un PON pour effectuer un diagnostic plus précis des anomalies observées sur un arbre PON, d'où l'intérêt de développer un modèle qui intègre les trois composants d'un arbre PON. Dans ce chapitre nous proposons une extension du modèle initial M_0 pour prendre en compte le réseau de distribution. Nous proposons un modèle de l'ODN qui s'adapte à toutes les ingénieries de distribution optique. L'ajout du modèle de l'ODN dans le modèle M_0 , conçu selon la méthode générique développée dans le Chapitre 5, est facilité par les propriétés de modularité et d'extensibilité du modèle générique. La Section 8.2 présente les diverses ingénieries de l'ODN qui rendent complexe sa modélisation. Un algorithme de construction automatique d'un modèle de l'ODN est présenté et formalisé dans la Section 8.3. Le modèle complet de propagation de pannes et d'alarmes sur un arbre PON du réseau GPON-FTTH est présenté dans la Section 8.4. La valeur ajoutée de la modélisation de l'ODN pour le diagnostic automatique d'un PON est présentée dans la Section 8.5. La Section 8.6 synthétise et conclut ce chapitre.

8.2 Diversité des ingénieries de l'ODN (Optical Distribution Network)

Comme le montre la Figure 6.1, un PON est une liaison optique entre un port du composant OLT et plusieurs instances du composant ONT. Cette liaison est établie au travers du réseau de distribution optique (ODN). L'ingénierie d'un PON ou l'ingénierie de son ODN désigne la topologie physique des coupleurs utilisés pour répartir la puissance optique transmise par l'OLT, entre tous les ONTs connectés sur ce PON. Un coupleur est un équipement passif qui possède la propriété de répartir équitablement la puissance optique reçue via sa fibre entrante ou fibre tronc, entre ses fibres sortantes. La Figure 8.1 présente le schéma synoptique d'un coupleur avec un taux de 1/4, c'est-à-dire un coupleur avec une fibre tronc et quatre fibres sortantes.

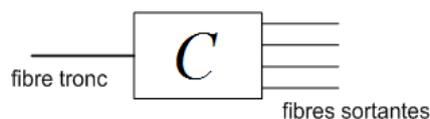


FIGURE 8.1 – Schéma synoptique d'un coupleur de taux 1/4.

L'ODN est constitué d'un ou de plusieurs niveaux de couplage et chaque niveau de couplage est constitué d'un ou de plusieurs coupleurs. La Figure 6.1 présente l'ingénierie la plus simple d'un ODN avec un seul niveau de couplage et un seul coupleur de taux 1/64.

La Figure 8.2 présente une ingénierie un peu plus dense avec trois niveaux de couplage. Le niveau 0 contient un seul coupleur de taux $1/2$, le niveau 1 contient deux coupleurs chacun de taux $1/2$ et le niveau 2 contient quatre coupleurs chacun de taux $1/16$. Le PON présenté par la Figure 8.2 permet donc de desservir un maximum de $16 \times 4 = 64$ ONTs.

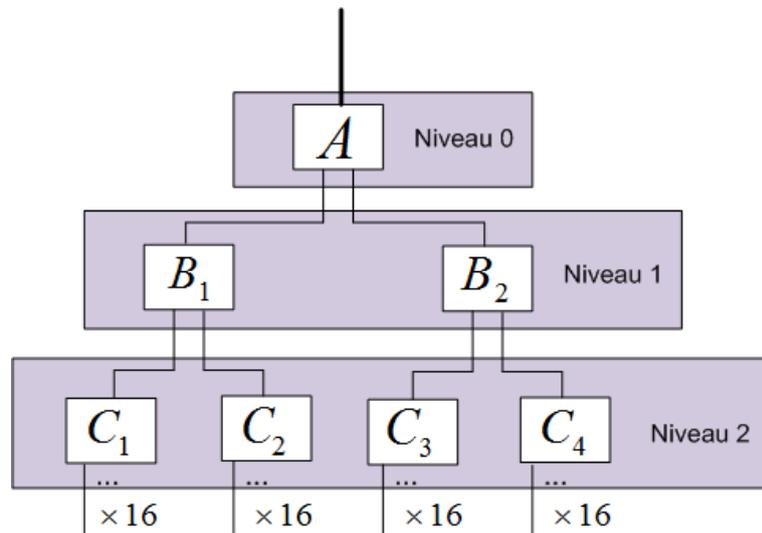


FIGURE 8.2 – Schéma synoptique de l'ingénierie ODN [(2,), (2, 2), (16, 16, 16, 16)].

Sur un PON, le segment entre l'OLT et le premier niveau de couplage est appelé réseau de transport. Le segment entre le dernier niveau de couplage et les ONTs est appelé réseau de connexion ou de branchement des abonnés. L'ingénierie ODN utilisée pour connecter le réseau de transport et le réseau de branchement des abonnés varie d'un PON à l'autre. Cette diversité d'ingénieries dans un réseau GPON-FTTH complexifie la modélisation du réseau de distribution. En effet, le modèle de l'ODN doit rester valable, quel que soit le PON sur lequel une ou plusieurs anomalies doivent être diagnostiquées.

8.3 Modélisation automatique de l'ODN

Pour faire face à la diversité des ingénieries du réseau de distribution optique, deux approches peuvent être utilisées pour sa modélisation. La première consiste à développer un modèle générique très abstrait de l'ODN de telle sorte que toutes ou la plupart des ingénieries s'y retrouvent. Dans ce cas, le modèle peut être bien adapté pour certaines ingénieries, mais aussi très imprécis pour d'autres. Par exemple, le modèle initial M_0 d'un PON développé dans le Chapitre 6 convient parfaitement à un PON dont le réseau de distribution se réduit à un seul niveau de couplage avec un seul coupleur de taux $1/64$. Par contre, le modèle M_0 est imprécis pour toutes les autres ingénieries.

La seconde approche consiste à développer également un modèle générique de l'ODN, mais en lui donnant cette fois-ci de l'intelligence pour générer une instance du modèle parfaitement adaptée à l'ingénierie du PON considéré. Si l'on fait la similitude avec la programmation orientée objet, le modèle générique est vu comme une classe et l'algorithme qui lui est associé peut être assimilé au constructeur de cette classe. Le constructeur est donc un algorithme qui reçoit en entrée les paramètres propres à une ingénierie de l'ODN, et génère automatiquement une instance de la classe ou du modèle générique. Le principal avantage de cette approche est que chaque instance est très précise puisque

construite avec les paramètres du PON pour lequel elle est générée. L'inconvénient est le coût de calcul nécessaire pour générer les instances et le temps d'acquisition des paramètres de l'ingénierie du PON.

Nous avons choisi la seconde approche puisque notre objectif principal est de faire un diagnostic efficace des anomalies qui surviennent sur un PON, c'est-à-dire localiser avec précision les pannes. Ceci n'est possible que si le modèle de propagation de pannes et d'alarmes sur ce PON est lui-même déjà précis.

8.3.1 Modèle générique de l'ODN et algorithme d'instanciation

Nous rappelons que pour être rajouté au modèle initial M_0 , le modèle du composant ODN doit être conçu conformément à la méthodologie de modélisation développée dans le Chapitre 5. Sur un PON, le composant ODN est physiquement placé entre les composants OLT et ONT auxquels il est connecté. Comme le modèle des composants OLT et ONT déjà présent dans le modèle initial M_0 d'un PON, le modèle de l'ODN est constitué de deux graphes de dépendances acycliques (DAGs). Un graphe qui modélise la propagation locale de pannes dans le réseau de distribution et un autre graphe qui modélise la propagation distribuée de pannes entre l'ODN et les composants OLT et ONT qu'il relie.

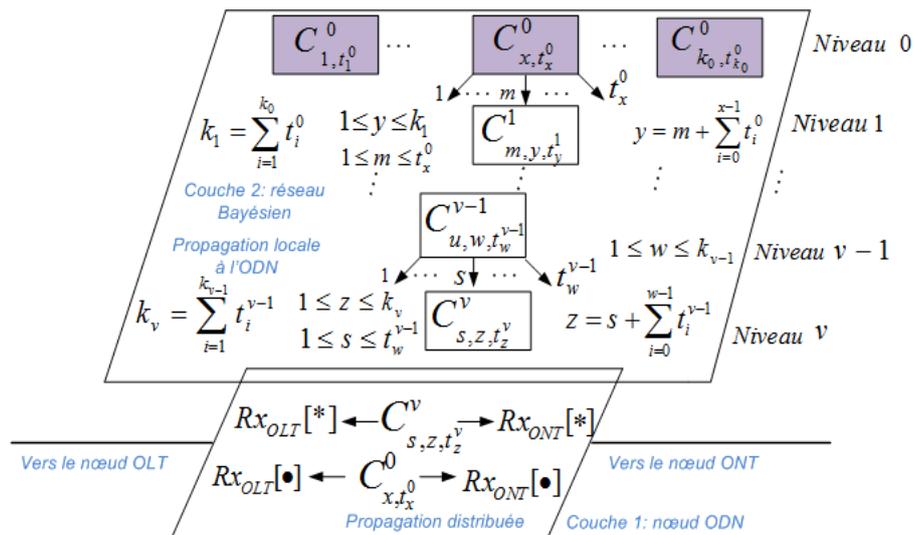


FIGURE 8.3 – Modèle du réseau de distribution optique d'un PON du réseau GPON-FTTH.

La Figure 8.3 présente le modèle générique de l'ODN d'un PON ainsi que l'algorithme permettant de construire automatiquement une instance de ce modèle pour une ingénierie donnée. Les noeuds et les dépendances du modèle de l'ODN sont générés automatiquement par l'algorithme formalisé par la Figure 8.3. Les noeuds représentent les coupleurs et les dépendances représentent les liaisons optiques entre eux. Cet algorithme requiert deux paramètres pour générer une instance du modèle de l'ODN et ajouter cette instance au modèle initial M_0 . Le premier paramètre désigne l'ingénierie de l'ODN, c'est-à-dire la topologie physique des coupleurs. Par exemple, [(2,), (2,2), (16,16,16,16)] désigne le paramètre qui identifie l'ingénierie présentée par la Figure 8.2. Le second paramètre permettant de rajouter l'instance du modèle ODN générée au modèle initial M_0 , désigne la configuration de la liaison entre le dernier niveau de couplage de l'instance générée et le réseau de branchement des ONTs. Ce second paramètre permet précisément de connaître les fibres de branchement qui sortent d'un quelconque coupleur du

dernier niveau de couplage. En d'autres mots, le second paramètre permet d'identifier pour chaque ONT, le coupleur du dernier niveau de couplage sur lequel il est connecté. Par exemple, pour un PON de quarante ONTs, $[10, 10, 10, 10]$ indique que les dix premiers ONTs sont connectés sur le premier coupleur du dernier niveau de couplage de l'ODN représenté par la Figure 8.2, les dix deuxièmes, les dix troisièmes et les dix quatrièmes ONTs sont respectivement connectés sur le deuxième, troisième et quatrième coupleur du dernier niveau de couplage.

Le réseau de distribution optique est constitué de plusieurs niveaux de couplage. Le DAG de la couche 2 qui modélise la propagation locale de pannes dans l'ODN est également constitué de plusieurs niveaux de couplage.

8.3.2 Le premier niveau de couplage

Le premier niveau de couplage noté *Niveau 0* sur la Figure 8.3 contient k_0 coupleurs. Un coupleur de ce niveau noté $C_{x,t_x^0}^0$ désigne le coupleur numéro x qui distribue la puissance optique reçue en entrée sur sa fibre tronc, à t_x^0 fibres sortantes numérotées de 1 à t_x^0 . Évidemment qu'en pratique, le premier niveau de couplage d'un PON du réseau GPON-FTTH comporte un seul coupleur dont la fibre tronc est la fibre de transport du PON notée $Fiber_T$ sur le modèle initial M_0 et partagée par tous les ONTs connectés sur ce PON. Cependant, le modèle de l'ODN proposé suppose qu'il existera peut-être dans le futur, certaines configurations avec plusieurs coupleurs au premier niveau de couplage, même si cela semble impossible aujourd'hui. Pour l'instant, la variable x sera donc considérée comme une constante ($x = 1$) lors de la génération d'une instance du modèle de l'ODN.

Il faut bien noter que le premier niveau de couplage est le point de démarcation entre le composant OLT et le composant ODN. Ces deux composants sont connectés au moyen des dépendances distribuées $C_{x,t_x^0}^0 \rightarrow Rx_{OLT}[\bullet]$ et $C_{\bullet,z,t_z^v}^v \rightarrow Rx_{OLT}[\star]$ ou z est le numéro d'un coupleur du dernier niveau de couplage v , comme le montre la couche 1 de la Figure 8.3. La première dépendance distribuée indique que l'état de l'unique coupleur $C_{1,t_1^0}^0$ du niveau 0 influence la puissance reçue à l'OLT de chaque ONT connecté sur le PON considéré ($Rx_{OLT}[\bullet]$). La seconde dépendance distribuée indique que l'état d'un coupleur de numéro z dans le dernier niveau de couplage influence uniquement la puissance reçue à l'OLT de chaque ONT qui lui est connecté ($Rx_{OLT}[\star]$). Les autres ONTs du PON qui sont connectés à d'autres coupleurs du dernier niveau de couplage ne sont pas influencés par le coupleur numéro z .

8.3.3 Les niveaux de couplage intermédiaires

Chaque coupleur noté $C_{p,q,t_q^l}^l$ est un coupleur du niveau $l \neq 0$ tel que $0 < l \leq v$ ou v représente le numéro du dernier niveau de couplage. Nous notons p le numéro de la fibre tronc de ce coupleur, c'est-à-dire la fibre qui connecte ce coupleur du niveau l avec son coupleur parent du niveau $l-1$. Les notations q et t_q^l désignent respectivement le numéro du coupleur dans le niveau de couplage l et le taux de ce coupleur. Par exemple, sur la Figure 8.3, une fibre numérotée m sortant du coupleur $C_{x,t_x^0}^0$ du premier niveau est reliée au coupleur $C_{m,y,t_y^1}^1$ du second niveau noté *Niveau 1*. Cette liaison optique est modélisée par une dépendance locale entre les deux coupleurs. L'indice $y \in \{1, \dots, k_1\}$ représente le numéro de ce coupleur dans la liste des coupleurs du niveau 1 et k_1 est le nombre total

de coupleurs du niveau 1. k_1 est déterminé en sommant les taux t_x^0 pour $x \in \{1, \dots, k_0\}$, des coupleurs du niveau 0.

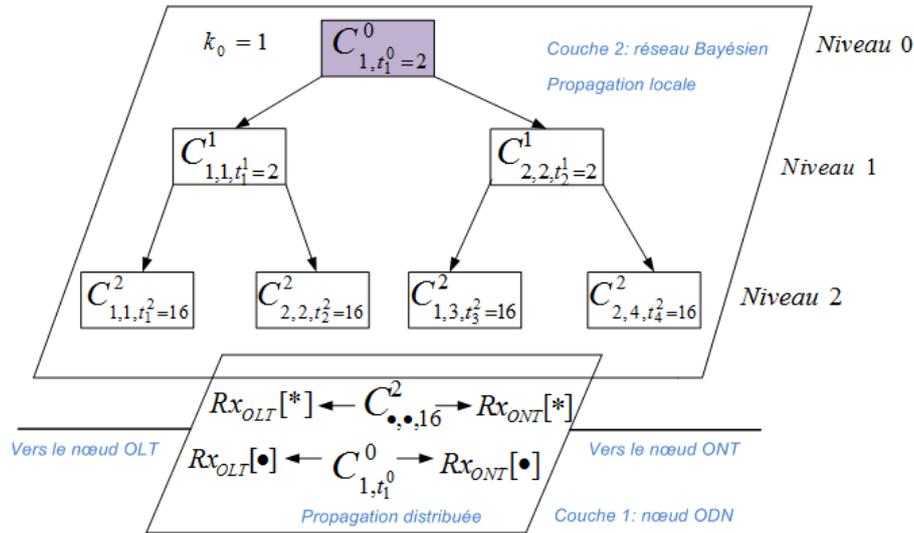


FIGURE 8.4 – Modèle de l'ODN pour l'ingénierie : [(2,), (2, 2), (16, 16, 16, 16)]

Un autre exemple, le coupleur $C_{u,w,t_w^{v-1}}^{v-1}$ de l'avant-dernier niveau de couplage noté *Niveau* $v - 1$, distribue la puissance optique qu'il a reçue sur sa fibre tronc à t_w^{v-1} fibres sortantes. Chacune de ces fibres sortantes numérotées $s \in \{1, \dots, t_w^{v-1}\}$ est reliée au coupleur $C_{s,z,t_z^v}^v$ du dernier niveau de couplage noté *Niveau* v . La Figure 8.4 représente l'instance du modèle de l'ODN générée automatiquement pour l'ingénierie [(2,), (2, 2), (16, 16, 16, 16)].

8.3.4 Le dernier niveau de couplage

Le dernier niveau de couplage est le point de démarcation entre le réseau de distribution et le réseau de branchement des abonnés ou des ONTs. C'est le point de jonction entre les composants ODN et ONT. En effet, chaque fibre sortante d'un coupleur $C_{•,z,t_z^v}^v$ du dernier niveau de couplage v , est la fibre de branchement d'un ONT du PON considéré. C'est pourquoi l'état de ce coupleur influence les niveaux de puissances Rx_{OLT} et Rx_{ONT} des ONTs qui lui sont connectés au moyen des fibres de branchement.

Les deux composants ODN et ONT sont connectés au moyen des dépendances distribuées $C_{x,t_x^0}^0 \rightarrow Rx_{ONT}[\bullet]$ et $C_{•,z,t_z^v}^v \rightarrow Rx_{ONT}[\star]$ comme le montre la couche 1 de la Figure 8.3. La première dépendance distribuée indique que l'état de l'unique coupleur $C_{1,t_1^0}^0$ du niveau 0 influence la puissance reçue à chaque ONT connecté sur le PON considéré ($Rx_{ONT}[\bullet]$). La seconde dépendance distribuée indique que l'état d'un coupleur de numéro z dans le dernier niveau de couplage influence uniquement la puissance reçue à chaque ONT qui lui est connecté ($Rx_{ONT}[\star]$). Les autres ONTs du PON qui sont connectés à d'autres coupleurs du dernier niveau de couplage ne sont pas influencés par le coupleur numéro z .

8.4 Modèle complet du réseau GPON-FTTH

La combinaison de la Figure 6.2 qui représente le modèle initial M_0 d'un PON développé dans le Chapitre 6 et de la Figure 8.3 qui représente le modèle générique de l'ODN,

forme le modèle complet M_1 de propagation de pannes et d'alarmes de n'importe quel PON, quel que soit l'ingénierie de son réseau de distribution. Le modèle M_1 intègre les trois composants OLT, ONT et ODN d'un PON. Nous rappelons que chaque composant i est modélisé par deux DAGs selon l'architecture en couche proposée dans le Chapitre 5, un graphe L_2N_i pour la propagation locale de pannes et d'alarmes sur le composant i et un graphe L_1N_i pour la propagation distribuée entre ce composant et ses voisins auxquels il est connecté. Le modèle complet M_1 est utilisé pour effectuer le diagnostic du réseau GPON-FTTH. Nous rappelons aussi que les calculs de diagnostic sont basés sur l'algorithme d'inférence sur la représentation en arbre de jonction du large réseau bayésien $G = \bigcup_{i=1}^3 L_2N_i \cup L_1N_i$. Cet algorithme d'inférence a été étudié dans le Chapitre 3.

8.5 Valeur ajoutée du modèle de l'ODN pour le diagnostic GPON-FTTH

Dans cette section, nous effectuons le diagnostic d'un réseau GPON-FTTH opérationnel. Afin d'évaluer le modèle de l'ODN, les résultats de diagnostic du modèle complet M_1 et ceux du modèle initial M_0 sont évalués par rapport au fonctionnement du réseau GPON-FTTH. Les expériences consistent à considérer un PON opérationnel et de générer automatiquement une instance du modèle de son ODN en fonction de l'ingénierie lue dans une base de données du système de gestion du réseau GPON-FTTH de l'opérateur Orange. Nous lisons également, le second paramètre qui nous permet de connaître la répartition des ONTs de ce PON sur les coupleurs du dernier niveau de couplage du réseau de distribution du PON. L'instance générée de l'ODN de ce PON est ensuite rajoutée au modèle initial M_0 pour obtenir le modèle complet M_1 du PON en question. Nous effectuons ensuite le diagnostic de plusieurs cas réels séparément avec les deux modèles. Un cas de diagnostic est un ensemble d'évidences ou d'observations collectées sur le réseau GPON-FTTH opérationnel d'Orange. Pour bien évaluer la plus-value du modèle de l'ODN, les cas de diagnostic sont choisis de telle sorte que la répartition des ONTs incriminés sur les coupleurs du dernier niveau de couplage varie d'un cas à l'autre. Un ONT est dit incriminé si certaines observations le concernant sont négatives.

Nous présentons et analysons ci-dessous les résultats de diagnostic effectués sur les PONs dont l'instance de l'ODN générée est celle de l'ingénierie [(2,), (2, 2), (16, 16, 16, 16)] présentée par la Figure 8.4. Le second paramètre d'un de ces PONs contenant quarante ONTs nous indique que les dix premiers, les dix deuxièmes, les dix troisièmes et les dix quatrièmes ONTs sont respectivement connectés aux coupleurs $C_{1,1,16}^2$, $C_{2,2,16}^2$, $C_{1,3,16}^2$ et $C_{2,4,16}^2$ du dernier niveau de couplage *Niveau 2*. Chaque Table de 8.1 à 8.5 décrit un scénario de diagnostic et montre les probabilités marginales des causes racines calculées par notre implémentation en Python des deux modèles, conditionnelles aux évidences brièvement décrites dans le titre de chaque table, et détaillées en dessous de la table.

La Table 8.1 décrit un scénario pour lequel le modèle initial M_0 et le modèle complet M_1 calculent le même diagnostic, c'est-à-dire l'atténuation de la fibre de branchement $Fiber_{DB_1}$ de l'ONT₁. Nous rappelons que les états OK, AT et BR dénotent respectivement une fibre qui n'atténue pas le signal qu'elle transporte, une fibre qui atténue ou une fibre rompue. L'état du réseau de distribution (ODN) est nominal puisque l'état de chaque fibre et chaque coupleur sur la Figure 8.4 est nominal, c'est-à-dire qu'il distribue la puissance optique qu'il reçoit sans perte considérable. L'état d'un coupleur est dit non nominal si la

TABLE 8.1 – Affaiblissement du signal entre l'OLT et l'ONT₁ et prise en compte de l'ODN.

Modèle	Causes racines	États	Croyances
M ₀	$Fiber_T$	[OK, AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_1}$	[OK, AT , BR]	[1.e-03, 0.99 , 3.e-06]
	$Fiber_{DB_{i \neq 1}}$	[OK, AT, BR]	[0.99 , 8.e-02, 2.e-06]
M ₁	$Fiber_T$	[OK, AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_1}$	[OK, AT , BR]	[1.e-03, 0.99 , 3.e-06]
	$Fiber_{DB_{i \neq 1}}$	[OK, AT, BR]	[0.99 , 8.e-02, 3.e-06]
	ODN	[OK, \neg OK]	[0.99 , 1.e-08]

Le PON contient 40 ONTs et aucune alarme n'est observée. Les puissances reçues $Rx_{OLT}[1]$ et $Rx_{ONT}[1]$ de l'ONT₁ sont faibles. Les puissances reçues $Rx_{OLT}[i]$ et $Rx_{ONT}[i]$, avec $i \neq 1$, des voisins de l'ONT₁ sont nominales.

répartition de la puissance se fait avec des pertes importantes ou si sa fibre tronc atténué ou est rompue. Dans le scénario de la Table 8.1, l'ODN semble transparent comme le suppose le modèle M₀ puisque les observations négatives collectées ne concernent qu'un seul ONT parmi quarante.

TABLE 8.2 – Connexion de toutes les sorties d'un coupleur à des ONTs à faibles puissances.

Modèle	Causes racines	États	Croyances
M ₀	$Fiber_T$	[OK, AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[1.e-03, 0.99 , 3.e-06]
	$Fiber_{DB_j}$	[OK, AT, BR]	[0.99 , 8.e-02, 2.e-06]
M ₁	$C_{1,1,16}^2$	[OK, \neg OK]	[0.273, 0.727]
	$Fiber_T$	[OK, AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[0.24, 0.52 , 0.24]
	$Fiber_{DB_j}$	[OK, AT, BR]	[0.91 , 8.e-02, 2.e-06]

Le PON contient 40 ONTs. Les puissances reçues $Rx_{OLT}[i]$ et $Rx_{ONT}[i]$ de l'ONT_{*i*}, pour $i \in \{1, \dots, 10\}$ sont faibles. Les alarmes SD[*i*] sont observées. Les puissances $Rx_{OLT}[j]$ et $Rx_{ONT}[j]$, pour $j \in \{11, \dots, 40\}$ des ONT_{*j*} sont nominales.

Dans la Table 8.2, les puissances reçues relatives aux dix ONTs connectés sur le PON par les fibres de branchement sortantes du coupleur $C_{1,1,16}^2$ sont faibles, tandis que les puissances reçues relatives aux trente autres ONTs connectés sur ce même PON par les fibres de branchement sortantes des autres coupleurs, sont nominales. Le modèle M₀ diagnostique que chaque fibre de branchement sortante du coupleur $C_{1,1,16}^2$ atténué, ignorant ainsi que l'observation de faibles puissances peut aussi être due à une perte de puissance par le coupleur ou par la fibre "tronc" qui l'alimente. Le modèle M₁ comprend bien que, la première cause racine la plus probable de faibles niveaux de puissances relatifs à tous ONTs connectés sur un même coupleur, est un dysfonctionnement de ce coupleur ou alors l'atténuation de sa fibre tronc. En effet, le modèle M₁ calcule un podium de deux causes racines probables. La seconde cause est l'atténuation de chaque fibre de branchement. Le diagnostic du modèle M₁ est très intéressant puisqu'il isole dans le réseau de distribution, le segment le plus probablement incriminé ou défectueux, c'est-à-dire le

lien optique entre le coupleur $C_{1,1,16}^2$ du dernier niveau de couplage et le coupleur $C_{1,1,2}^1$ de l'avant-dernier niveau. On remarque donc très bien pour ce scénario que le diagnostic du modèle complet M_1 est plus précis et donc meilleur que ce lui du modèle initial M_0 .

TABLE 8.3 – Connexion des ONTs à faibles puissances à toutes les sorties de deux coupleurs précédés par un coupleur commun dans l'arborescence du réseau de distribution.

Modèle	Causes racines	États	Croyances
M_0	$Fiber_T$	[OK , AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[1.e-03, 0.99 , 3.e-06]
	$Fiber_{DB_j}$	[OK , AT, BR]	[0.99 , 8.e-02, 2.e-06]
M_1	$C_{1,1,16}^2$	[OK, \neg OK]	[0.004, 0.996]
	$C_{2,2,16}^2$	[OK, \neg OK]	[0.004, 0.996]
	$C_{1,1,2}^1$	[OK, \neg OK]	[0.010, 0.980]
	$Fiber_T$	[OK , AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[0.24, 0.52 , 0.24]
	$Fiber_{DB_j}$	[OK , AT, BR]	[0.91 , 8.e-02, 2.e-06]

Le PON contient 40 ONTs. Les puissances $Rx_{OLT}[i]$ et $Rx_{ONT}[i]$ des ONT_i , pour $i \in \{1, \dots, 20\}$ sont faibles. Les alarmes $SD[i]$ sont observées. Les puissances $Rx_{OLT}[j]$ et $Rx_{ONT}[j]$, pour $j \in \{21, \dots, 40\}$ des ONT_j sont nominales.

La Table 8.3 décrit une situation pour laquelle le modèle M_1 isole trois segments incriminés dans le réseau de distribution du PON. Une situation similaire est présentée par la Table 8.4 pour laquelle deux segments défectueux dans l'ODN sont identifiés.

TABLE 8.4 – Connexion des ONTs à faibles puissances à toutes les sorties de deux coupleurs n'étant pas précédés par un coupleur commun dans l'arborescence de l'ODN.

Modèles	Causes racines	États	Croyances
M_0	$Fiber_T$	[OK , AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[1.e-03, 0.99 , 3.e-06]
	$Fiber_{DB_j}$	[OK , AT, BR]	[0.99 , 8.e-02, 2.e-06]
M_1	$C_{1,1,16}^2$	[OK, \neg OK]	[0.004, 0.996]
	$C_{2,4,16}^2$	[OK, \neg OK]	[0.004, 0.996]
	$Fiber_T$	[OK , AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[0.24, 0.52 , 0.24]
	$Fiber_{DB_j}$	[OK , AT, BR]	[0.91 , 8.e-02, 2.e-06]

Le PON contient 40 ONTs. Les puissances $Rx_{OLT}[i]$ et $Rx_{ONT}[i]$ des ONT_i , pour $i \in \{1, \dots, 10, 31, \dots, 40\}$ sont faibles. Les alarmes $SD[i]$ sont observées. Les puissances $Rx_{OLT}[j]$ et $Rx_{ONT}[j]$, pour $j \in \{11, \dots, 30\}$ des ONT_j sont nominales.

La Table 8.5 présente une situation pour laquelle il y a vingt ONTs sur le PON avec de faibles niveaux de puissances reçues sur le canal montant et descendant du PON. Ces vingt ONTs sont équitablement répartis sur les quatre coupleurs du dernier niveau de couplage. Le modèle M_1 ne diagnostique aucun coupleur comme étant défectueux. Ce qui est tout à fait logique puisque chaque coupleur du dernier niveau de couplage est

TABLE 8.5 – Connexion des sorties d’un coupleur à certains ONTs à faibles niveaux de puissances et d’autres ONTs à puissances nominales.

Modèles	Causes racines	États	Croyances
M_0	$Fiber_T$	[OK , AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[1.e-03, 0.99 , 3.e-06]
	$Fiber_{DB_j}$	[OK , AT, BR]	[0.99 , 8.e-02, 2.e-06]
M_1	C_{\dots}^2	[OK , \neg OK]	[0.955 , 0.045]
	$Fiber_T$	[OK , AT, BR]	[0.99 , 4.e-20, 3.e-22]
	$Fiber_{DB_i}$	[OK, AT , BR]	[1.e-03, 0.99 , 9.e-05]
	$Fiber_{DB_j}$	[OK , AT, BR]	[0.91 , 8.e-02, 2.e-06]

Le PON contient 40 ONTs. Aucune alarme n’est observée. Les puissances $Rx_{OLT}[i]$ et $Rx_{ONT}[i]$ des ONT $_i$, pour $i \in \{1, \dots, 5, 11, \dots, 15, 21, \dots, 25, 31, \dots, 35\}$ sont faibles. Les puissances $Rx_{OLT}[j]$ et $Rx_{ONT}[j]$, pour $j \in \{6, \dots, 10, 16, \dots, 20, 26, \dots, 30, 36, \dots, 40\}$ des ONT $_j$ sont nominales.

connecté à certains ONTs avec des puissances reçues faibles et d’autres ONTs avec des puissances reçues nominales. Par conséquent, le réseau de distribution n’est pas incriminé dans ce cas, c’est plutôt l’état du réseau de branchement qui n’est pas nominal puisque c’est l’atténuation introduite par les fibres de branchement des vingt ONTs incriminés qui cause l’observation de niveaux de puissances faibles.

8.6 Discussion et conclusion

Ce chapitre propose une extension du modèle initial d’un PON du réseau GPON-FTTH développé dans le Chapitre 6. En effet, ce modèle initial ne comportait que deux des trois composants d’un PON, à savoir l’OLT et l’ONT. Nous avons proposé dans ce chapitre un modèle du réseau de distribution optique (ODN) qui constitue le troisième composant d’un PON. Ce modèle de l’ODN, basé sur un réseau bayésien, s’adapte à toutes les ingénieries de PONs. En effet, ce modèle est vu comme une classe qui est automatiquement instanciée en fonction de l’ingénierie du PON considéré. L’algorithme qui génère automatiquement une instance et qui la rajoute au modèle initial est présenté dans ce chapitre. Le nouveau modèle obtenu est dit complet puisqu’il intègre les trois composants OLT, ONT et ODN, et modélise la chaîne complète des dépendances entre les pannes ou causes racines, les causes intermédiaires et les alarmes observées sur un PON.

Afin d’évaluer la plus-value du modèle de l’ODN pour le diagnostic d’un PON, nous avons analysé et comparé les résultats de diagnostic du modèle initial et du modèle complet. Il existe des cas pour lesquels le diagnostic des deux modèles n’est pas exactement le même, ceci dû au fait que, contrairement au modèle initial, le modèle complet est capable d’isoler un ou plusieurs segments optiques défectueux dans le réseau de distribution. Cette différence s’explique par le fait que le modèle initial effectue un diagnostic de bout en bout d’un PON en ne considérant que les deux composants aux extrémités du PON. Il ne s’intéresse pas à ce qui passe entre les extrémités OLT et ONT, c’est-à-dire dans le réseau de distribution, raison pour laquelle le diagnostic du modèle complet est plus précis. Cependant, lorsque le modèle complet diagnostique un ou plusieurs segments incriminés dans le réseau de distribution comme dans les situations présentées par les

Tables 8.2, 8.3 et 8.4, bien que ce résultat soit en accord avec la logique "experte" attendue, il ne permet pas de savoir si une liaison incriminée est due à un dysfonctionnement intrinsèque au coupleur ou à une perte de puissance de la fibre "tronc" qui alimente ce coupleur. En effet, aucune observation sur le réseau GPON-FTTH ne permet actuellement de discriminer entre ces deux causes racines potentielles. Par contre, le diagnostic du modèle complet permet à l'opérateur d'envoyer un technicien d'intervention au bon endroit sur l'infrastructure GPON-FTTH. Le technicien pourra utiliser ses outils de tests pour lever les incertitudes résiduelles et effectuer une réparation efficace.

Chapitre 9

Conclusion

« Misunderstanding of probability may be the greatest of all impediments to scientific literacy. »

Stephen Jay Gould, 1995

Sommaire

9.1 Rappel des objectifs de la thèse	140
9.2 Synthèse des contributions	140
9.2.1 Modèle générique	140
9.2.2 Diagnostic du réseau d'accès optique GPON-FTTH	141
9.3 Perspectives envisagées	142

9.1 Rappel des objectifs de la thèse

Cette thèse s'inscrit dans le projet FABOAM (Fixed Access, Backhaul and core network OAM) du programme de recherche NOEMI (Network OpERation, administration, Maintenance and Integration) d'Orange Labs dont l'objectif est de développer des Opérations d'Administration et de Maintenance (OAM) pour la gestion flexible et automatisée des équipements et des périphériques réseau, afin de faciliter l'exploitation et la valorisation des réseaux. Les travaux de cette thèse se focalisent sur le développement d'un outil de diagnostic automatique des réseaux d'accès fixes bien qu'ils soient suffisamment génériques pour être appliqués à d'autres types de réseaux. L'objectif est de dépasser les méthodes courantes de diagnostic basées sur des règles spécialisées qui ont déjà montré leurs limites lorsque la dimension du réseau à diagnostiquer est grande et les signatures de pannes très diversifiées. L'idée sous-jacente est de développer une méthode de diagnostic qui s'appuie sur un modèle ou une représentation explicite de la structure et du fonctionnement du réseau en faisant recours aux méthodes probabilistes de gestion de l'incertitude puisque le fonctionnement d'un réseau encore appelé propagation de pannes et d'alarmes est un phénomène non déterministe en plus d'être distribué et parfois dynamique. L'objectif opérationnel pour l'opérateur de télécommunication Orange est d'améliorer les performances de localisation de pannes du système expert DELC (Diagnostic Expert de la Ligne Cliente) actuellement en exploitation dans son système de gestion des réseaux d'accès cuivre et optique, et de réduire ainsi les coûts relatifs aux interventions dans les infrastructures.

9.2 Synthèse des contributions

9.2.1 Modèle générique

Bien que l'objectif opérationnel de la thèse se focalise sur les réseaux d'accès, nous avons proposé une méthode générique qui facilite la modélisation de tout système distribué y compris les réseaux de télécommunications. Cette méthode repose sur un principe simple : modéliser le fonctionnement ou le comportement d'un réseau revient à modéliser le comportement de chacun des composants qui le constituent. Chaque composant est modélisé par deux graphes de dépendances acycliques (DAG) : un DAG qui modélise la propagation locale de pannes et d'alarmes sur le composant considéré et autre DAG qui modélise la propagation distribuée de pannes et d'alarmes entre ce composant et les composants voisins auxquels il est connecté. En effet, nous séparons la propagation locale de pannes et d'alarmes sur un composant, de la propagation distribuée entre composants connectés. Procédant de cette façon, nous modélisons également la topologie du réseau. Le modèle générique est organisé en trois couches. La couche 1 contient l'ensemble des DAGs modélisant la propagation distribuée entre composants connectés. La couche 2 contient l'ensemble des DAGs modélisant chacun la propagation locale sur un composant. Les couches 1 et 2 sont conçues de telle sorte que leur combinaison donne un large DAG qui modélise le fonctionnement du réseau tout entier. Mais la séparation entre la propagation locale et la propagation distribuée, implémentée par les couches 1 et 2, décompose ce large DAG en plusieurs parties interconnectées apportant ainsi des propriétés de modularité et d'extensibilité au modèle. La couche 3 est une représentation en arbre de jonction du réseau bayésien obtenu en probabilisant le large graphe. Notons que cette méthode générique apporte une solution permettant de réduire la complexité de modélisation des réseaux de très grande dimension. En effet, on peut modéliser le ré-

seau de façon incrémentale, c'est-à-dire composant après composant. Le modèle du réseau peut donc être modifié facilement pour s'adapter aux éventuelles évolutions de son architecture. En effet, de nouveaux composants peuvent être ajoutés dans le modèle ou être retirés du modèle lorsque l'opérateur le souhaite.

9.2.2 Diagnostic du réseau d'accès optique GPON-FTTH

La méthode générique a été appliquée pour modéliser la propagation de pannes et d'alarmes sur le réseau d'accès optique GPON-FTTH. Nous avons commencé par modéliser deux des trois composants d'un PON du réseau GPON-FTTH : les composants OLT et ONT situés aux extrémités d'un PON. Les paramètres de ce modèle initial ont été approximativement déterminés sur la base des connaissances expertes relatives au fonctionnement du réseau GPON-FTTH. Une implémentation en Python appelé PANDA 1.0 de ce modèle ainsi que de l'algorithme d'inférence sur un arbre de jonction, nous a permis de doublement valider ce modèle initial. La validation consiste à vérifier que le modèle donne des résultats tout à fait cohérents et raisonnables vis-à-vis de l'expertise métier du réseau GPON-FTTH. Dans un premier temps, la vérification est faite avec des cas de diagnostic simples générés en laboratoire sur une plateforme expérimentale. Dans un second temps, la vérification est faite avec des cas de diagnostic réels issus du réseau GPON-FTTH commercial de l'opérateur Orange. L'outil PANDA 1.0 donne des résultats de diagnostic cohérents avec l'expertise métier du réseau d'accès GPON-FTTH, et plus raisonnables sur un échantillon de cas de diagnostic représentatifs, que le système expert à base de règles DELC.

Puisqu'il est très difficile même pour des experts avérés d'un réseau de déterminer avec précision les paramètres d'un modèle statistique de ce réseau, nous avons pensé que les performances de diagnostic du modèle probabiliste seraient meilleures si les paramètres de ce modèle sont artificiellement appris à partir des données générées par les équipements du réseau. Une étude détaillée de l'algorithme EM (Expectation Maximization) a été réalisée et nous avons ajusté automatiquement les paramètres "experts" du modèle initial en appliquant l'EM sur des données réelles du réseau GPON-FTTH. Notons que l'EM a été initialisé avec les paramètres "experts" afin d'être sûr qu'il converge vers de nouveaux paramètres plus vraisemblables aux données du réseau GPON-FTTH que les paramètres "experts". Cette seconde implémentation a été appelée PANDA 2.0 et ses performances sont raisonnablement meilleures que ceux de PANDA 1.0 sans toutefois s'en écarter considérablement, ce qui veut dire que les paramètres "experts" de PANDA 1.0 ont bien été calibrés.

Cependant, les deux premières versions de l'outil PANDA sont des implémentations d'un modèle partiel ou initial du réseau GPON-FTTH. En effet, le modèle initial n'effectue que le diagnostic de bout en bout d'un PON puisqu'il ne contient que les composants aux extrémités du PON. Nous avons proposé une extension du modèle initial appelé PANDA 3.0 en y intégrant le modèle du réseau de distribution optique (ODN) qui constitue le troisième composant d'un PON. PANDA 3.0 est de ce fait capable de regarder ce qui se passe dans l'ODN puisqu'il permet de corrélérer ou déduire l'état des coupleurs et des liaisons optiques dans l'ODN sur la base des observations aux extrémités d'un PON. PANDA 3.0 reproduit les calculs de diagnostic des deux premières versions en plus d'être capable d'isoler un ou plusieurs liens incriminés dans le réseau de distribution optique.

9.3 Perspectives envisagées

Une perspective très intéressante des travaux menés dans le cadre de cette thèse serait de décomposer l'arbre de jonction qui constitue la couche 3 du modèle générique. En effet, la représentation en arbre de jonction de la combinaison des couches 1 et 2 n'est pas unique. Au lieu d'avoir une couche 3 faite d'un seul grand arbre de jonction, on peut envisager d'avoir plutôt n arbres de jonction pour un réseau de n composants tel que l'arbre de jonction i est équivalent au DAG i qui modélise la propagation locale de pannes sur le composant i . Les arbres de jonction de deux composants connectés i et $j \neq i$ seront ainsi connectés au moyen des cliques communes, c'est-à-dire les cliques contenant certaines variables appartenant au DAG i et d'autres variables appartenant au DAG j . L'intérêt de cette approche serait de distribuer et de paralléliser l'inférence sur la couche 3 du modèle entre les arbres de jonction.

Une autre perspective serait d'ajuster automatiquement les dépendances et les paramètres du modèle probabiliste du réseau GPON-FTTH avec l'objectif d'améliorer encore davantage les performances de diagnostic. Un algorithme tel que l'algorithme SEM (Structural Expectation Maximization) pourrait être étudié et appliqué dans ce cas. Notons que l'application de l'algorithme EM paramétrique ou structurel s'avère très intéressante pour un opérateur de télécommunications. En effet, une application pratique pourrait être d'implémenter un système de diagnostic complètement autonome dans lequel une boucle fermée est créée entre le modèle probabiliste de diagnostic intégrant les capacités d'apprentissage artificiel, les résultats de diagnostic conduisant à des interventions ciblées des techniciens sur l'infrastructure et les données reportées par les techniciens qui pourront être utilisées pour ajuster automatiquement le modèle probabiliste pour le rendre plus performant. Cependant la fiabilité de ces données doit être garantie pour éviter de faire dériver le modèle au lieu de l'améliorer.

Actions de valorisation

Publications

1. Serge Romaric TEMBO, Sandrine VATON, Jean-Luc COURANT, Stéphane GOSSELIN, Michel BEUVELOT. *Model-based probabilistic reasoning for self-diagnosis of telecommunication networks : application to a GPON-FTTH access network*. JNSM, Journal of Network and systems Management.
2. Serge Romaric TEMBO, Jean-Luc COURANT, Sandrine VATON , Stéphane GOSSELIN. *A highly adaptable probabilistic model for self-diagnosis of GPON-FTTH access networks*. IEEE 24 th International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2016, Croatia, Split, September 22-24.
3. Serge Romaric TEMBO, Sandrine VATON, Jean-Luc COURANT, Stéphane GOSSELIN. *A tutorial on the EM algorithm for Bayesian networks : application to self-diagnosis of GPON-FTTH networks*. IEEE Technically Sponsored Workshop TRAC 2016, Traffic analysis and Characterization, Cyprus, Paphos, September 5-9, 2016.
4. Serge Romaric TEMBO, Jean-Luc COURANT, Sandrine VATON. *A 3-Layered Self-Reconfigurable Generic Model For Self-Diagnosis of Telecommunication Networks*. IEEE Technically Sponsored SAI Intelligent Systems, IntelliSys 2015, London, November 10-11, 2015.
5. Serge Romaric TEMBO, The Tung NGUYEN, Didier ELBAZ. *Distributed Iterative Solution of Numerical Simulation Problems on Infiniband and Ethernet Clusters via the P2PSAP Self-adaptive Protocol*. 21st International Conference on Parallel, Distributed and networked-based Processing, PDP 2013, Belfast Northern Ireland, IEEE CPS, February 27 – March 1, 2013.
6. Serge Romaric TEMBO, Didier EL BAZ. *Distributed resolution of a trajectory optimization problem on a MEMS-based reconfigurable modular surface*. IEEE International Conference on Internet of Things, iThings 2013, 20 – 23 August 2013, Beijing (China), August 2013, pp.355-358.

Démonstrations

1. Serge Romaric TEMBO, Jean-Luc COURANT, Alexandre FERRIEUX, Sandrine VATON. *PANDA : Probabilistic tool for GPON-FTTH access Networks self-Diagnosis*. Demonstration at Orange Labs Research Exhibition 2015, France, Paris, December 1-3.
2. Serge Romaric TEMBO, Jean-Luc COURANT, Alexandre FERRIEUX, Stéphane GOSSELIN, Sandrine VATON. *PANDA : Probabilistic tool for GPON-FTTH access Networks self-Diagnosis*. Demonstration at Orange Labs Research Exposition 2015, France, Lannion, December 15.

Références bibliographiques

- AGHASARYAN, A., E. FABRE et A. BENVENIST. 1998, «Fault detection and diagnosis in distributed systems : An approach by partially stochastic petri nets», *Journal of Discrete Event Dynamic Systems : Theory and Applications*, vol. 8, p. 203–231.
- ANDREAS, H. 2006, «A hybrid rule-based/case-based reasoning approach for service fault diagnosis», *20th IEEE International Conference on Advanced Information Networking and Applications (AINA-2006)*, vol. 1.
- BAYES, T. 1958, «An essay towards solving a problem in the doctrine of chances», *Biometrika*, vol. 46, p. 293–298.
- BEACON, N. M. 2000, «A hierarchical network topology monitoring system based in IP multicast», *Services Management in Intelligent Networks*, p. 169–180.
- BENVENISTE, A., E. FABRE, S. HAAR et C. JARD. 2003, «Diagnosis of asynchronous discrete-event systems : a net unfolding approach», *Transactions on Automatic Control*, vol. 45, n° 5, p. 714–727.
- BERGMANN, R. et S. STAHL. 1998, «Similarity measures for object-oriented case representations», dans *B. Smyth and P. Cunningham (Eds.) Advances in Case-Based Reasoning (EWCBR'98)*.
- BERRY, A., P. HEGGERNES et G. SIMONET. 2003, «The minimum degree heuristic and the minimal triangulation process», *H. Bodlaender. WG'03 : 29th International Workshop on Graph Theoretic Concepts in Computer Science, Elspeet (The Netherlands), Lecture Notes in Computer Science lirmm-00191916*, p. 58–70.
- BRAMS, G. W. 1983, *Réseau de Petri : Théorie et pratique, tome 1*, Masson.
- CABASINO, M. P., A. GIUA, S. LAFORTUNE et C. SEATZU. 2012, «A new approach for diagnosability analysis of petri nets using verifier nets», *Transactions on Automatic Control*, vol. 57, n° 12, p. 3104—3117.
- CELEUX, G. et G. DIEBOLT. 1985, *The sem algorithm : a probabilistic teacher algorithm derived from the em algorithm for the mixture problem*, Rapport de recherche RR-1364, Inria, Institut National de Recherche en Informatique et en Automatique.
- CELEUX, G. et G. GOVAERT. 1991, «A classification em algorithm for clustering and two stochastic versions», *Computational Statistics Quarterly*, vol. 2, n° 1, p. 73–82.
- CHAO, C. S., D. L. YANG et A. C. LIU. 2001, «An automated fault diagnosis system using hierarchical reasoning and alarm correlation», *Journal of Network and Systems Management*, vol. 9, n° 2, p. 183—202.
- CORNÉJOLS, A. et P. MICLET. 2013, *Apprentissage artificiel, concepts et algorithmes*, EYROLLES.
- COX, R. T. 1946, «Probability, frequency and reasonable expectation», *American Journal of Physics*, vol. 14, n° 1, p. 1–13.
- CRONK, R. N., P. H. CALLAHAN et B. LAWRENCE. 1998, «Rule-based expert systems for network management and operations : an introduction», *IEEE Network*, vol. 2, p. 7–21.

-
- DAVID, J. M. et J. P. KRIVINE. 1987, «Three artificial intelligence issues in fault diagnosis : declarative programming, expert systems, and model-based reasoning», *Fault detection and reliability : knowledge based and other approaches*, p. 19–27.
- DEFINETTI, B. 1972, *Theory of Probability : a critical introductory treatment*, Probability and Statistics Series, John Wiley & Sons Australia, Limited, ISBN 9780471201410.
- DEMPSTER, A. P., N. N. LAIRD et D. RUBIN. 1977, «Maximum likelihood from incomplete data via the em algorithm», *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, n° 1, p. 1–38.
- EDGEWORTH et Y. FRANCIS. 1909, «On the probable errors of frequency-constants», *Journal of the Royal Statistical Society*, vol. 71, n° 3, p. 459–512.
- ESPARZA, J., S. ROMER et W. VOGLER. 2002, «An improvement of mcmillan’s unfolding algorithm», *Formal Methods in System Design*, vol. 20, n° 3, p. 285–310.
- FRIEDMAN, N. et M. GOLDSZMIDT. 1998, *Tutorial : Learning Bayesian Networks from Data*, Université de Berkeley.
- FROHLICH, P., W. NEJDL, K. JOBMANN et H. WIETGREFE. 1997, «Model-based alarm correlation in cellular phone networks», dans *5th International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, MASCOTS 1997*, p. 197–204.
- GARDNER, R. D. et D. A. HARLE. 1996, «Methods and systems for alarm correlation», dans *Global Telecommunications Conference. Communications : The Key to Global Prosperity, GLOBECOM 1996*.
- GARDNER, R. D. et D. A. HARLE. 1997, «Alarm correlation and network fault resolution using the Kohonen self-organising map», *Global Telecommunications Conference (GLOBECOM 1997)*, p. 1398–1402.
- GOOD, I. J. 1950, *Probability and the Weighing of Evidence*, Charles Griffin.
- GOPAL, R. 2000, «Layered model for supporting fault isolation and recovery», *IEEE Network Operation and Management Symposium*.
- GROSCLAUDE, I. 2008, «Une approche a base de modeles pour les services de depannage (model based problem solving for help and support services)», *French Conference on Artificial Intelligence, RFIA 2008*.
- HAMMOND, K. 1989, «Case-based planning : Viewing planning as a memory task», *Boston Academic Press*.
- HERMANN, W. 2002, «Investigation and practical assessment of alarm correlation methods for the use in gsm access networks», *IEEE/IFIP Network Operations and Management Symposium, NOMS 2002*, p. 391–403.
- HOUCK, K., S. CALO et A. FINKEL. 1995, «Towards a practical alarm correlation system», dans *Proceedings of the fourth international symposium on Integrated network management*, p. 226–237.
- HOUNKONNOU, C. 2013, *Active self-diagnosis in telecommunication networks*, PhD Thesis, European University of Brittany, University of Rennes 1, INRIA, ISTIC, France.

-
- TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU. 2003, *G.777.1 Recommendation*, ITU-T.
- TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU. 2008, *G.984.3 Recommendation*, ITU-T.
- JAKOBSON, G. et M. WEISSMAN. 1995, «Real-time telecommunication network management : extending event correlation with temporal constraints», dans *Proceedings of the fourth international symposium on Integrated network management*, p. 290–301.
- JAKOBSON, G. et M. D. WEISSMANI. 1993, «Alarm correlation», *IEEE Network*, vol. 7, n° 6, p. 52–59.
- JENSEN, F., S. LAURITZEN et K. OLESEN. 1990, «Bayesian updating in causal probabilistic networks by local computations», *Computational Statistics Quarterly*, vol. 4, p. 269–282.
- JENSEN, F. V. et T. D. NIELSEN. 2007, *Bayesian Networks and Decision Graphs*, Springer, Information Science & Statistics.
- JINGXIAN, L., C. DOUSSON et F. KRIEF. 2011, «A self-diagnosis algorithm based on causal graphs», *Seventh International Conference on Autonomic and Autonomous Systems, ICAS 2011*.
- JINGXIAN, L., C. DOUSSON, B. RADIER et F. KRIEF. 2010, «Towards an autonomic network architecture for self-healing in telecommunications networks», *Mechanisms for Autonomous Management of Networks and Services*, vol. 6155, p. 110—113.
- JORDAAN, J. F. et M. E. PATEROKL. 1993, «Event correlation in heterogeneous networks using the osi management framework», dans *Proceedings of the third international symposium on Integrated network management*, p. 683–695.
- KÄTKER, S. et K. GEIHS. 1997, «A generic model for fault isolation in integrated management systems», *Journal of Network and Systems Management*, vol. 5, n° 2, p. 109–130.
- KÄTKER, S. et M. PATEROK. 1997, «Fault isolation and event correlation for integrated fault management», dans *Proceedings of the fifth international symposium on Integrated network management*, p. 583–596.
- KATZELA, I. et M. SCHWARTZ. 1995, «Schemes for fault identification in communication networks», *IEEE/ACM Transactions on Networking*, vol. 9, n° 2, p. 733—764.
- KHANAFER, R. M., B. SOLANA, J. TRIOLA, R. BARCO, L. MOLTSSEN, Z. ALTMAN et P. LAZARO. 2008, «Automated diagnosis for umts networks using bayesian network approach», *IEEE Transactions on Vehicular Technology*, vol. 57, p. 2451–2461.
- LAGRANGE, J. L. 1888, «Manière plus simple et plus générale de faire usage de la formule de l'équilibre donnée dans la section deuxième, tome 11», dans *MECANIQUE ANALYTIQUE*, p. 77–112.
- LAURITZEN, S. 1995, «The EM algorithm for graphical association models with missing data», *Computational Statistics and Data Analysis*, vol. 19, p. 191–201.
- LAURITZEN, S. 1996, *Graphical models*, Oxford Statistical Science Series, Clarendon Press, ISBN 9780191591228.

-
- LAURITZEN, S. et D. SPIEGELHALTER. 1988, «Local computations with probabilities on graphical structures and their application to expert systems», *Journal of the Royal Statistical Society, Series B*, vol. 50, n° 2, p. 157–224.
- LEWIS, L. 1993, «A case-based reasoning approach to the resolution of faults in communication networks», *Proceedings of the third international symposium on Integrated network management*, p. 671–682.
- LEWIS, L. et G. DREA. 1993, «Extending trouble ticket systems to fault diagnostics», *IEEE Network*, vol. 7, p. 44–51.
- LOR, E. et KAR-WING. 1993, «A network diagnostic expert system for acculink multiplexers based on a general network diagnostic scheme», dans *IFIP TC6/WG6.6 Third International Symposium on Integrated Network Management with participation of the IEEE Communications Society CNOM and with support from the Institute for Educational Services*, p. 659–669.
- MADSEN, A. L. et F. V. JENSEN. 1999, «Lazy propagation : a junction tree inference algorithm based on lazy evaluation», *Artificial Intelligence*, vol. 113, p. 203–245.
- MALGORZATA, S. et S. S. ADARSHPAL. 2001, «Multi-layer fault localization using probabilistic inference in bipartite dependency graph», *University of Delaware, Newark, DE, Technical Report*.
- MALGORZATA, S. et S. S. ADARSHPAL. 2004, «A survey of fault localization techniques in computer networks», *Elsevier Science of Computer Programming*, vol. 53, p. 165–194.
- MELCHIORI, C. et L. M. ROCKENBACH TAROUÇO. 1999, «Fault management in computer networks using case-based reasoning : Dumbo system», dans *Proceedings of the Third International Conference on Case-Based Reasoning and Development*, p. 510–524.
- MIYAZAWA, M. et K. NISHIMURA. 2011, «Scalable root cause analysis assisted by classified alarm information model based algorithm», dans *7th International Conference on Network and Service Management, CNSM 2011*, p. 1–4.
- NAIM, P., P. H. WUILLEMIN, P. LERAY, O. POURRET et A. BECKER. 2008, *Réseaux Bayésiens*, EYROLLES.
- PAU, L. F. 1986, «Survey of expert systems for fault detection, test generation and maintenance», *Expert Systems*, vol. 3, p. 100–110.
- PEARL, J. 1985, «Bayesian networks : A model of self-activated memory for evidential reasoning», dans *Proceedings of the 7th Conference of the Cognitive Science Society*, p. 329–334.
- PEARL, J. 1986, «Fusion, propagation, et structuring in belief networks», *Artificial Intelligence*, vol. 29, p. 241–288.
- PEARL, J. 1988, *Probabilistic reasoning in intelligent systems : networks of plausible inference*, Morgan Kaufmann, ISBN 9781558604797.
- PENIDO, G. C., J. M. S. NOGUEIRA et M. C. MACHADO. 1999, «An automatic fault diagnosis and correction system for telecommunications management», dans *Proceedings of the Sixth IFIP/IEEE International Symposium on Integrated Network Management*, p. 777–791.

-
- PFANZAGL, JOHANN et R. HAMBÖKER. 1994, *Parametric statistical theory*, Walter de Gruyter, Berlin, DE, ISBN 3-11-013863-8, p. 207–208.
- PHAM, T. D. 1988, «Expert systems in engineering», *Springer-Verlag New York*.
- RALPH, B. 2001, «Highlights of the european inreca projects», dans *Proceedings of the 4th International Conference on Case-Based Reasoning, ICCBR-2001, Vancouver, Canada, Springer*.
- RAMANATHAN, S., D. CASWELL et S. NEAL. 2000, «Auto-discovery capabilities for service management : An isp case study», *Journal of Network and Systems Management*, vol. 8, n° 4, p. 457–482.
- REATEGUI, E. B., J. A. CAMPBELL et B. F. LEO. 1997, «Combining a neural network with case-based reasoning in a diagnostic system», *Artificial Intelligence in Medicine, Case-based Reasoning : Opportunities and Applications in Health Care*, vol. 9, n° 1, p. 5–27.
- RICHTER, M. M. 1995, «The knowledge contained in similarity measures. invited talk at the first international conference on cbr (iccbr-95)», <http://www.cbrweb.org/documents/Richtericcbr95remarks.html>.
- RUSSEL, S. et P. NORVIG. 2003, *Artificial Intelligence : A Modern Approach*, Prentice Hall.
- SANCHEZ, J. M., I. G. BEN YAHIA et N. CRESPI. 2015, «Self modeling based diagnosis of software defined networks», *1st Conference on network softwarization (NetSoft)*.
- SASISEKHARAM, R., V. SESHADRI et M. W. SHOLOM. 1996, «Data mining and forecasting in large scale telecommunication networks», *IEEE Expert, AT&T Bell Labs, Rutgers University*, vol. 11, p. 37–43.
- SAVAGE, L. J. 1972, *The Foundation of Statistics*, Dover Books on Mathematics Series, Dover Publications, ISBN 9780486623498.
- SAVAGE, L. J. 1976, «On rereading R. A. fisher», *The Annals of Statistics*, vol. 4, n° 3, p. 441–500.
- SHIFFMAN, D. 2012, *The nature of code*, Shannon Fry.
- STEINDER, M. et A. S. SETHI. 2002, «End-to-end service failure diagnosis using belief networks», *Network Operation and Management Symposium*, p. 375—390.
- STEINDER, M. et A. S. SETHI. 2004, «Probabilistic fault diagnosis in communication systems through incremental hypothesis updating», *Comput Networks*, vol. 45, p. 537—562.
- YANG, B.-S., T. HAN et K. YONG-SU. 2003, «Integration of art-kohonen neural network and case-based reasoning for intelligent fault diagnosis», *School of Mechanical Engineering, Pukyong National University, San 100, Yongdang-dong, Nam-gu, Busan 608-739, South Korea*.
- YEMINI, S. A., S. KLIGER, E. MOZES, Y. YEMINI et D. OHSIE. 1996, «High speed and robust event correlation», *IEEE Communications Magazine*, vol. 34, n° 5, p. 82—90.
- ZIEMER, W. P. 1989, *Weakly Differentiable Functions : Sobolev Spaces and Functions of Bounded Variation*, SPRINGER, ISBN 0387970177.

Les réseaux de télécommunication doivent être fiables et robustes pour garantir la haute disponibilité des services. Les opérateurs cherchent actuellement à automatiser autant que possible les opérations complexes de gestion des réseaux, telles que le diagnostic de pannes.

Dans cette thèse nous nous sommes intéressés au diagnostic automatique de pannes dans les réseaux d'accès optiques de l'opérateur Orange. L'outil de diagnostic utilisé jusqu'à présent, nommé DELC, est un système expert à base de règles de décision. Ce système est performant mais difficile à maintenir en raison, en particulier, du très grand volume d'informations à analyser. Il est également impossible de disposer d'une règle pour chaque configuration possible de panne, de sorte que certaines pannes ne sont actuellement pas diagnostiquées.

Dans cette thèse nous avons proposé une nouvelle approche. Dans notre approche, le diagnostic des causes racines des anomalies et alarmes observées s'appuie sur une modélisation probabiliste, de type réseau bayésien, des relations de dépendance entre les différentes alarmes, compteurs, pannes intermédiaires et causes racines au niveau des différents équipements de réseau. Ce modèle probabiliste a été conçu de manière modulaire, de façon à pouvoir évoluer en cas de modification de l'architecture physique du réseau.

Le diagnostic des causes racines des anomalies est effectué par inférence, dans le réseau bayésien, de l'état des noeuds non observés au vu des observations (compteurs, alarmes intermédiaires, etc...) récoltées sur le réseau de l'opérateur. La structure du réseau bayésien, ainsi que l'ordre de grandeur des paramètres probabilistes de ce modèle, ont été déterminés en intégrant dans le modèle les connaissances des experts spécialistes du diagnostic sur ce segment de réseau. L'analyse de milliers de cas de diagnostic de pannes a ensuite permis de calibrer finement les paramètres probabilistes du modèle grâce à un algorithme EM (Expectation Maximization).

Les performances de l'outil développé, nommé PANDA, ont été évaluées sur deux mois de diagnostic de panne dans le réseau GPON-FTTH d'Orange en juillet-août 2015. Dans la plupart des cas, le nouveau système, PANDA, et le système en production, DELC, font un diagnostic identique. Cependant un certain nombre de cas sont non diagnostiqués par DELC mais ils sont correctement diagnostiqués par PANDA. Les cas pour lesquels les deux systèmes émettent des diagnostics différents ont été évalués manuellement, ce qui a permis de démontrer dans chacun de ces cas la pertinence des décisions prises par PANDA.

Mots clés : Diagnostic, Réseau d'accès, GPON-FTTH, Localisation de pannes, Réseau Bayésien, Inférence probabiliste, Estimation paramétrique, Expectation Maximization

Telecommunication networks must be reliable and robust to ensure high availability of services. Operators are currently searching to automate as much as possible, complex network management operations such as fault diagnosis.

In this thesis we are focused on self-diagnosis of failures in the optical access networks of the operator Orange. The diagnostic tool used up to now, called DELC, is an expert system based on decision rules. This system is efficient but difficult to maintain due in particular to the very large volume of information to analyze. It is also impossible to have a rule for each possible fault configuration, so that some faults are currently not diagnosed.

We proposed in this thesis a new approach. In our approach, the diagnosis of the root causes of malfunctions and alarms is based on a Bayesian network probabilistic model of dependency relationships between the different alarms, counters, intermediate faults and root causes at the level of the various network component. This probabilistic model has been designed in a modular way, so as to be able to evolve in case of modification of the physical architecture of the network.

Self-diagnosis of the root causes of malfunctions and alarms is made by inference in the Bayesian network model of the state of the nodes not observed in view of observations (counters, alarms, etc.) collected on the operator's network. The structure of the Bayesian network, as well as the order of magnitude of the probabilistic parameters of this model, were determined by integrating in the model the expert knowledge of the diagnostic experts on this segment of the network. The analysis of thousands of cases of fault diagnosis allowed to fine-tune the probabilistic parameters of the model thanks to an Expectation Maximization algorithm.

The performance of the developed probabilistic tool, named PANDA, was evaluated over two months of fault diagnosis in Orange's GPON-FTTH network in July-August 2015. In most cases, the new system, PANDA, and the system in production, DELC, make an identical diagnosis. However, a number of cases are not diagnosed by DELC but are correctly diagnosed by PANDA. The cases for which self-diagnosis results of the two systems are different were evaluated manually, which made it possible to demonstrate in each of these cases the relevance of the decisions taken by PANDA.

Keywords: Self-Diagnosis, access network, GPON-FTTH, fault localization, Bayesian network, probabilistic inference, parameter estimation, Expectation Maximization