



HAL
open science

Réseau bayésien dynamique hybride : application à la modélisation de la fiabilité de systèmes à espaces d'états discrets

Florence Petiet

► To cite this version:

Florence Petiet. Réseau bayésien dynamique hybride : application à la modélisation de la fiabilité de systèmes à espaces d'états discrets. Automatique. Université Paris-Est, 2019. Français. NNT : 2019PESC2014 . tel-02390519

HAL Id: tel-02390519

<https://theses.hal.science/tel-02390519>

Submitted on 3 Dec 2019

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.

Thèse présentée pour obtenir le grade de docteur
Université Paris-Est

École doctorale MSTIC

Discipline : Génie informatique, automatique et traitement du signal

Réseau bayésien dynamique hybride : application à la modélisation de la fiabilité de systèmes à espaces d'états discrets

PAR : Florence PETIET

Directeur : Laurent BOUILLAUT, Chargé de Recherche, HDR, IFSTTAR

Encadrant : Olivier FRANÇOIS, Ingénieur de Recherche, IFSTTAR

MEMBRES DU JURY:

Rapporteur : Alexandre AUSSEM, Professeur des universités, Université Lyon 1

Rapporteur : Philippe WEBER, Professeur des universités, Université de Lorraine

Examineur : Anne BARROS, Professeure des universités, NTNU

Examineur : Roland DONAT, Docteur, Président de EdgeMind

Date de soutenance : 1er juillet 2019

A Ghislain

A Grand-Père

A Grand-Mère

Remerciements

Pour commencer, je remercie le jury d'avoir consacré du temps à l'évaluation de mes travaux et d'avoir assisté à ma soutenance, et Anne Barros d'avoir accepté de le présider. Je remercie particulièrement mes rapporteurs, Philippe Weber et Alexandre Aussem pour le temps qu'ils ont pris pour lire mon manuscrit et pour leurs remarques qui m'ont permis de mieux mettre en perspective l'ensemble de mes travaux.

Mes remerciements vont ensuite à Laurent Bouillaut, qui a dirigé ma thèse, et à Olivier François, qui l'a encadrée.

Je tiens à remercier une nouvelle fois Anne Barros, pour son accueil à NTNU à Trondheim, et pour l'ouverture scientifique que ce séjour de recherche m'a donné. Je remercie également l'Ecole Doctorale MSTIC pour la bourse de mobilité qu'elle m'a accordée. Je tiens par ailleurs à remercier Sylvie Cach pour son accompagnement, son aide et sa gentillesse tout au long de mes années de thèse.

Je tiens à remercier l'ensemble des membres du GRETTIA actuels et passés pour l'accueil amical que j'ai reçu. Je remercie particulièrement Mustapha, Neïla, Latifa, Pierre-Alain, Jean-Patrick, Régine, Etienne, Allou, Cyril, croisés aux conseils du département COSYS, à la cantine, ou dans les couloirs.

J'ai une pensée particulière pour les post-doctorants et les doctorants que j'ai côtoyés : Josquin, Andry, et Zineb pour leur accueil et leur patience à mon arrivée, ainsi que Moncef, Maxime, Florian, Nassim, Milad, Negin (pour les séances coiffure!).

J'ai une pensée particulière pour M.-Khalil, devenu un ami, et pour Anne-Sarah, également devenue une amie, mais aussi une témoin de mariage.

Je remercie mes "amis du lycée" qui, sans forcément le savoir, m'ont donné le courage de partir en classe prépa après le bac, puis quelques années après, de démarrer une thèse.

Enfin je tiens à remercier ma famille et mes autres amis de m'avoir encouragée pendant toute ma thèse, particulièrement oncle Xavier qui m'a incitée à faire une thèse.

Je remercie tout particulièrement François, mon mari, qui a notamment supporté une fin de thèse coïncidant avec la préparation de notre mariage.

Résumé

L'analyse de fiabilité fait partie intégrante de la conception et du fonctionnement d'un système, en particulier pour les systèmes exécutant des applications critiques. Des travaux récents ont montré l'intérêt d'utiliser les réseaux bayésiens dans le domaine de la fiabilité pour modéliser la dégradation d'un système. Les modèles graphiques de durée sont un cas particulier des réseaux bayésiens, qui permettent de s'affranchir de la propriété markovienne des réseaux bayésiens dynamiques. Ils s'adaptent aux systèmes dont le temps de séjour dans chaque état n'est pas nécessairement distribué exponentiellement, comme c'est le cas dans la plupart des applications industrielles. Des travaux antérieurs ont toutefois montré des limitations à ces modèles en terme de capacité de stockage et de temps de calcul, en raison du caractère discret de la variable temps de séjour. Nous proposons d'utiliser une variable "temps de séjour" continue pour limiter cette complexité calculatoire. Après avoir remarqué que la loi de Weibull permettait de représenter convenablement les données de durée de vie issues de retours d'expérience sur les composants qui nous intéressent, l'objectif est d'intégrer des variables de temps de séjour suivant une distribution de Weibull dans un modèle de durée graphique en proposant un nouvel algorithme d'inférence *ad-hoc*.

Après une présentation des réseaux bayésiens, et plus particulièrement des modèles graphiques de durée et leur limitation, ce rapport s'attache à présenter le nouveau modèle MGDHW, pour modèle graphique de durée hybride Weibull, permettant la modélisation du processus de dégradation. Un algorithme original permettant l'inférence dans un tel réseau est alors détaillé.

Ne disposant pas de données particulières pour cette étude, différentes bases de données ont été simulées pour apprendre d'une part un modèle graphique de durée et d'autre part un modèle graphique de durée hybride-Weibull, afin de comparer les qualités de leurs apprentissages et l'efficacité de leur algorithme d'inférence respectifs.

Abstract

Reliability analysis is an integral part of the design and the operation of a system, especially for systems running critical applications. Recent works have shown the interest of using Bayesian Networks in the field of reliability, to model the degradation of a system. The Graphical Duration Models are a specific case of Bayesian Networks, which make it possible to overcome the Markovian property of dynamic Bayesian Networks. They adapt to systems whose sojourn-time in each state is not necessarily exponentially distributed, which is the case for most industrial applications. Previous works have however shown limitations in these models in terms of storage capacity and computing time, due to the discrete nature of the sojourn time variable. We propose to use a continuous sojourn time variable to limit this computational complexity. After noting that the Weibull distribution correctly represents lifetime data from feedback on the components we are interested in, the aim is to integrate sojourn time variables following a Weibull distribution in a Graphical Duration Model by proposing a new *ad-hoc* algorithm.

After a presentation of the Bayesian networks, and more particularly graphical duration models, and their limitations, this report focus on presenting the new model WHGDM, for Weibull-Hybrid Graphical Duration Model allowing the modeling of the degradation process. An original algorithm allowing the inference in such a network is then detailed.

Because we do not have specific data for this study, various databases were simulated to learn on one hand a Graphical Duration Model, and on an other hand a Hybrid - Weibull Graphical Duration Model, in order to compare the qualities of their respective learning and the effectiveness of their respective inference algorithms.

Table des matières

Introduction générale	1
1 Contexte de l'étude	5
1.1 Cadre général	5
1.1.1 Un peu d'histoire...	5
1.1.2 Sûreté de fonctionnement : la science des défaillances	6
1.1.3 Politiques de maintenance	9
1.2 Processus de dégradation	10
1.2.1 Maintenance basée sur la fiabilité	10
1.2.2 Éléments bibliographiques	11
2 Réseaux bayésiens	15
2.1 Généralités sur les Réseaux Bayésiens	15
2.1.1 Historique	15
2.1.2 Définition générale	16
2.1.3 Utilisations des réseaux bayésiens	16
2.1.4 Contexte d'utilisation des réseaux bayésiens au laboratoire GRETIA	18
2.2 Formalisme des réseaux bayésiens	20
2.2.1 Définitions et généralités	20
2.2.2 Réseau bayésien dynamique	21
2.2.3 Modèle graphique de durée	25
2.2.4 Apprentissage	27
2.2.5 Inférence	29
2.3 Des limitations des MGD vers une nouvelle problématique	37
3 MGDHW	39
3.1 Généralités sur le formalisme des MGDHW	39
3.1.1 État de l'art sur les réseaux bayésiens admettant des variables continues	39
3.1.2 Formalisme	40
3.1.3 Apprentissages	42
3.2 Inférence	43

3.2.1	Algorithme itératif	43
3.2.2	Méthode de calcul	45
4	Résultats expérimentaux	51
4.1	Présentation du protocole expérimental	51
4.1.1	Introduction	51
4.1.2	Choix d'un paramètre initial impactant la taille de F	52
4.1.3	Lois de probabilités conditionnelles du système initial	55
4.1.4	Protocole expérimental	57
4.2	Création d'une base de trajectoires	58
4.2.1	Taille T des trajectoires de $\mathcal{D}^{\alpha_{ok}}$	59
4.2.2	Simulation d'une trajectoire	61
4.2.3	Stockage	62
4.2.4	Exemple	62
4.3	Apprentissages du MGD et du MGDHW	63
4.3.1	Phase d'apprentissage	64
4.3.2	Qualité d'un apprentissage	68
4.3.3	Résultats : comparaisons des qualités d'apprentissage	70
4.4	Comparaison des inférences	73
4.4.1	Phase d'inférence	75
4.4.2	Comparaison des temps de traitement	76
4.4.3	Calcul théorique de la fiabilité	77
4.4.4	Précision	78
4.5	Conclusion	80
	Conclusions et perspectives	81
	Conclusions	81
	Perspectives	82
	Bibliographie	92
	Liste des figures	94
	Liste des tables	95
	Liste des algorithmes	97

Introduction générale

Depuis la seconde moitié du XXe siècle et le développement des secteurs dits "à risque" (aviation civile, nucléaire, aérospatiale), un nouveau domaine scientifique a vu le jour afin d'aider à maintenir en état opérationnel un système pour lequel les exigences en disponibilité, sécurité, fiabilité et maintenabilité sont élevées : la sûreté de fonctionnement, également appelée science des défaillances. Les nouvelles exigences sociétales de disponibilité et de qualité de service et les contraintes environnementales ont contribué au développement d'outils plus complexes pour ce nouveau domaine. De plus, des événements tels que l'accident nucléaire de Tchernobyl en 1986, qui fut à la base du développement des concepts modernes de maintenance basée sur la fiabilité, ont conduit à ce que les exigences liées à la sûreté de fonctionnement soient devenues de plus en plus importantes dans l'industrie. De nouvelles approches sont ainsi sans cesse proposées par la communauté scientifique afin d'assurer aux systèmes un niveau de fonctionnement satisfaisant, tout en veillant aux différents enjeux économiques, environnementaux, sociétaux, et réglementaires. Pour ce faire, une étape généralement nécessaire est l'étude du processus de dégradation conduisant aux défaillances redoutées. De nombreux travaux ont ainsi vu le jour, tels que l'ouvrage de [COCOZZA-THIVENT 1997], dont le but est de montrer concrètement ce que peut apporter l'étude des processus stochastiques dans le domaine de la fiabilité, et celui de [AVEN et JENSEN 1999], qui présente les grands axes de la fiabilité du point de vue des processus stochastiques.

En outre, avec l'évolution du numérique dans l'entreprise, de plus en plus de retours d'expérience sont disponibles, notamment sous forme de bases de données plus ou moins bien formalisées.

Depuis une vingtaine d'années, de nombreuses méthodes d'analyse numérique pour la fiabilité sont donc apparues, depuis de simples lois de survie jusqu'à des modèles dynamiques qui présentent l'avantage de pouvoir modéliser l'évolution de la dégradation au cours du temps. Une introduction de ces méthodes est présentée dans [RAUSAND et HØYLAND 2004].

Si les chaînes de Markov [MARKOV 1971] sont certainement en 2018 une des approches dynamiques les plus utilisées dans la littérature pour modéliser la dégradation de systèmes à espaces d'états discret, les réseaux bayésiens [JENSEN 1996; PEARL 1985], et surtout leur extension temporelle, nommée réseaux bayésiens dyna-

miques (RBD) [MURPHY 2002], présentent l'avantage de généraliser plusieurs types de modèles dynamiques. En effet, il a été montré qu'un RBD pouvait modéliser une chaîne de Markov, ou encore un réseau de Pétri stochastique [NATKIN 1980].

Les réseaux bayésiens permettent de représenter de façon intuitive et factorisée la distribution jointe de plusieurs variables, par l'utilisation conjointe de graphiques et de distributions de probabilité.

Leur puissance de modélisation et de calcul a ainsi contribué à faire des RBD un formalisme aujourd'hui très utilisé en modélisation de la fiabilité et de la maintenance de systèmes à espaces d'états finis et discrets, comme l'illustrent [BROY et al. 2011] et [ZAHRA et al. 2013] en utilisant les RBD pour modéliser la fiabilité d'un système.

Ce constat a conduit le laboratoire GRETTIA à mener des travaux qui ont conduit au développement de modèles de dégradation et de modèles de maintenance, tels que les modèles graphiques de durée (MGD) [DONAT et al. 2010], mis en place il y a une dizaine d'années. Contrairement aux RBD standards, les MGD contiennent une variable de temps de séjour discrète qui modélise le temps passé dans chaque état du système. L'introduction de ce noeud permet de s'affranchir des limitations des réseaux bayésiens dynamiques qui, de par leur propriété markovienne, nécessitent des taux de transition constants, et ne sont donc pas adaptés au comportement d'un grand nombre d'applications industrielles. Les MGD sont toutefois limités par le caractère nécessairement discret des variables qui le décrivent (états du système, temps passé dans chaque état), en particulier de la variable modélisant le temps de séjour passé dans l'état de fonctionnement courant. L'impact de ces limites a été mis en évidence par la thèse de Josquin Foulliaron [FOULLIARON 2015], développant des outils de pronostic à partir du formalisme des réseaux bayésiens dynamiques et des modèles graphiques de durée dans le but de permettre la modélisation de politiques de maintenance préventives prévisionnelles. Il a notamment été mis en évidence qu'avec l'augmentation de la taille de la matrice décrivant les temps de séjour, la complexité spatiale du réseau bayésien proposé pour estimer et actualiser la distribution de la durée de vie résiduelle pouvait mener à des saturations de la mémoire lors des calculs d'inférence.

Ces constats ont conduit au sujet de ma thèse dont l'objet est de proposer de nouvelles extensions des MGD permettant de s'affranchir de cette limite de lois de temps de séjour discrètes. Pour cela, il a été imaginé d'élaborer un modèle graphique de durée ayant une variable "durée de séjour" continue. Certains travaux se sont déjà intéressés aux RB continus. Ainsi, [LAURITZEN 1992] a proposé un formalisme de RB statique, dans lequel les variables sont soit discrètes de parents discrets, soit

continues avec une distribution gaussienne conditionnelle en leurs parents. [MORAL et al. 2001] a quant à lui proposé un modèle à base de mélange d'exponentielles tronquées. Toutefois, dans un MGD, la nature des distributions conditionnelles des variables dépend de la valeur de la variable de temps de séjour : elles sont nécessairement déterministes lorsque le temps de séjour restant indique qu'il n'y a pas encore de changement d'état. A notre connaissance, il n'existe pas d'étude s'intéressant à des RBD comportant des variables ayant une distribution conditionnelle dont la nature dépend de la valeur des variables parentes.

L'objectif de la thèse était donc de permettre à la variable modélisant le temps de séjour dans un nouvel état d'être continue et de suivre une loi de Weibull après changement d'état. En effet, cette loi, très souple et communément utilisée en fiabilité, s'avère pertinente pour modéliser le comportement de systèmes industriels. Elle est par exemple efficace pour les systèmes qui se basent sur la répétition d'un élément structurel, tels que les voies de chemin de fer, ou les flottes de véhicules, sujets auxquels ces travaux sont destinés en premier lieu. Nous verrons, par ailleurs, que nos travaux sont généralisables à une plus grande famille de lois.

Pour présenter ces travaux, le manuscrit proposé s'organise en quatre chapitres.

Une introduction du cadre général d'élaboration d'un modèle de dégradation est tout d'abord présentée dans le chapitre 1.

Le chapitre 2 présente ensuite quelques cas d'utilisation du réseau bayésien, pour bien en comprendre l'intérêt dans l'approche considérée. Puis il aborde le formalisme des réseaux bayésiens et leur extension dynamique. Les limites de ces derniers sont ensuite exposées, puis les modèles graphiques de durée (MGD) sont introduits. Le chapitre présente enfin l'apprentissage des paramètres décrivant un MGD et développe une technique d'inférence au sein du réseau bayésien.

Le chapitre 3 présente en détail le réseau bayésien hybride particulier proposé, appelé modèle graphique de durée hybride Weibull (MGDHW). Ce dernier a pour objectif de permettre à la distribution du temps de séjour d'être continue et de suivre une loi de Weibull. Ce nouveau formalisme nécessite un algorithme original d'inférence, qui fait l'objet de la deuxième partie de ce chapitre 3.

Le chapitre 4 présente la méthodologie mise en place pour valider le modèle proposé. Ne disposant pas de retours d'expérience utilisable dans le cadre de cette thèse, une bases de données est d'abord simulée sur un cas "jouet". Ce dernier est un système non réparable à trois états de fonctionnement. À partir d'un certain nombre de séquences d'états de fonctionnement et pour différents paramètres initiaux, un modèle graphique de durée hybride Weibull et un modèle graphique de durée ont

été appris, et les qualités d'apprentissage et les complexités spatiales sont évaluées afin de comparer les deux approches. La fiabilité est ensuite estimée, d'une part à l'aide d'une inférence classique pour le MGD et, d'autre part, à l'aide de l'inférence mise en place au chapitre 3 pour le MGDHW, puis des comparaisons sont effectuées entre les temps de calcul, et entre ces estimations de fiabilité et la fiabilité théorique.

Pour finir, des conclusions générales sur ce travail de thèse sont proposées et des perspectives de poursuites de ces travaux sont introduites et détaillées.

Contexte de l'étude

Notre objectif est de modéliser la dégradation de systèmes à états discrets, dans un objectif de maintenance. Ce chapitre présente le cadre de la sûreté de fonctionnement, ainsi que les différentes stratégies de maintenance existantes. L'objectif à terme étant d'étudier une stratégie de maintenance sur un système à états discrets à partir de données, la maintenance basée sur la fiabilité fera l'objet de la deuxième partie, avec notamment un état de l'art sur la modélisation de la dégradation.

1.1 Cadre général

1.1.1 Un peu d'histoire...

À l'apparition de l'industrie, l'activité de maintenance se résumait à des opérations de dépannage ou de réparation après défaillance. Peu d'études de sûreté étaient réalisées. Ceci a conduit à de nombreux accidents tragiques, faisant des centaines de victimes. Nous pouvons citer les exemples de l'explosion de la poudrerie de Grenelle en 1794, de la catastrophe ferroviaire de Meudon en 1842 (figure 1.1), et du naufrage du Titanic en 1912 notamment dû à l'absence préalable d'étude de sûreté approfondie.



FIGURE 1.1 – *Catastrophe ferroviaire de Meudon, [HENRY 1842]*

Il n'y avait alors aucune notion d'intervention préventive. C'est l'expansion des chemins de fer, de l'automobile, et surtout de l'aviation, essentiellement entre les

deux guerres mondiales, qui a permis le développement d'exigences de sûreté de fonctionnement, l'industrie prenant de plus en plus conscience de l'importance de pouvoir se fier dans ses équipements.

L'enjeu était alors de prendre en compte tous les risques afin de minimiser leurs conséquences, en prenant des mesures adéquates, c'est-à-dire en définissant une politique de maintenance préventive qui conserve le système dans un niveau de fonctionnement satisfaisant.

Initialement [ZWINGELSTEIN 1996], les fabricants planifiaient leurs maintenances selon une fréquence élevée d'intervention, pour se protéger contre toutes les perturbations majeures. Mais ces interventions étaient nécessairement associées à des coûts de maintenance très élevés. Si cette démarche était coûteuse pour l'industriel, elle était indispensable vis-à-vis de l'opinion publique, alors de plus en plus critique.

Ces raisons économiques ont poussé les fabricants à chercher à affiner leurs politiques de maintenance afin de réduire les coûts tout en maintenant une haute qualité de service. De nombreuses études [KAPUR et LAMBERSON 1977] ont ainsi eu pour objet la sûreté de fonctionnement, ainsi que la maintenance de systèmes industriels.

1.1.2 Sûreté de fonctionnement : la science des défaillances

Avant toute chose, il est nécessaire de définir ce que vont désigner les mots «système» et «composant» dans cette thèse. Un système est un assemblage de composants liés entre eux et destinés à accomplir une ou plusieurs fonctions. Un système est considéré "en panne" lorsqu'il n'est plus en mesure d'accomplir la fonction qui lui est dévolue.

Un système non réparable est un système non entretenu qui ne peut pas retourner seul à un meilleur état une fois qu'il cesse de fonctionner correctement. Un système réparable est un système qui, après avoir perdu une ou plusieurs fonctionnalités, peut être restauré par une action de maintenance.

La sûreté de fonctionnement est l'aptitude d'un système à satisfaire à une ou plusieurs fonctions requises dans des conditions données [VILLEMEUR 1988]. Elle permet notamment d'évaluer statistiquement le risque pris en fonction des choix de politiques de maintenance, à travers quatre aptitudes regroupées sous le sigle FMDS (ou RAMS en anglais) : la fiabilité, la maintenabilité, la disponibilité, et la sécurité.

Nous allons donc brièvement introduire chacune de ces grandeurs, même si l'essentiel des travaux de cette thèse portent sur l'estimation de la fiabilité, voire de la disponibilité.

Fiabilité

La fiabilité d'un système est son aptitude à rester constamment opérationnel pendant une durée donnée.

C'est la probabilité que le système fonctionne au-delà de l'instant t . On peut aussi la voir comme la probabilité qu'il soit toujours dans un état de fonctionnement jusqu'à l'instant t . On l'appelle aussi fonction de survie.

La fiabilité est définie par la fonction $R : \mathbf{N}^* \mapsto [0, 1]$:

$$\begin{aligned} R(t) &= P(\text{Première défaillance} > t) \\ &= P(\text{Système non défaillant sur } [0, t]) \\ &= P(X_1 \in \mathcal{U}; \dots; X_t \in \mathcal{U}) \end{aligned}$$

avec \mathcal{U} l'ensemble des états de fonctionnement et X_t l'état du système à l'instant t .

Il est possible de tirer de cette définition de la fiabilité d'autres métriques intéressantes, telles que le taux de défaillance, défini par $\lambda(t) = -\frac{1}{R(t)} \frac{dR(t)}{dt}$. [PHAM 2006]

Lorsque le taux instantané de défaillance diminue avec le temps, on parle de « mortalité infantile » : les systèmes ayant des « défauts de jeunesse » ont des défaillances précoces, les systèmes qui « survivent » sont intrinsèquement robustes. Cela peut aussi décrire une situation de rodage.

Lorsque le risque de défaillance augmente avec le temps, cela indique un phénomène d'usure.

Dans le cas général, le taux de défaillance n'est pas monotone. On a souvent une première période de mortalité infantile (élimination des systèmes défectueux, rodage), avec un λ décroissant, puis éventuellement une période de pannes aléatoires avec un λ constant, et enfin une période d'usure avec un λ croissant. C'est ce qu'on appelle communément courbe de baignoire (figure 1.2).

Maintenabilité

La maintenabilité d'un système est son aptitude à être remis rapidement dans un état opérationnel.

C'est la probabilité que la maintenance du système soit achevée au temps t , sachant que le système est défaillant à l'instant initial.

La maintenabilité est définie par la fonction $M : \mathbf{N}^* \mapsto [0, 1]$:

$$M(t) = P(\text{Système réparé dans l'intervalle } [0, t])$$

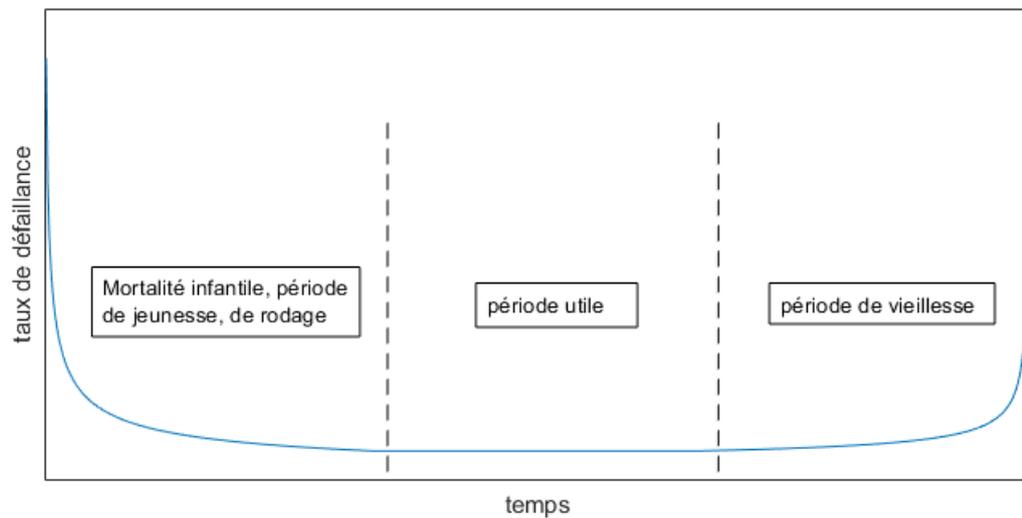


FIGURE 1.2 – Courbe en baignoire

Disponibilité

La disponibilité d'un système est son aptitude à être dans un état opérationnel à un instant donné.

C'est la probabilité que le système soit dans un état de fonctionnement à l'instant t , quels que soient ses états passés.

La disponibilité est définie par la fonction $A : \mathbf{N}^* \mapsto [0, 1]$:

$$\begin{aligned} A(t) &= P(\text{Système non défaillant à l'instant } t) \\ &= P(X_t \in \mathcal{U}) \end{aligned}$$

avec \mathcal{U} l'ensemble des états de fonctionnement.

Une disponibilité importante est compatible avec une fiabilité faible, pour peu que l'appareil puisse être maintenu très fréquemment et rapidement.

Sécurité

La sécurité d'un système est son aptitude à ne pas connaître de pannes considérées comme catastrophiques pendant une durée donnée.

C'est la probabilité que le système évite de faire apparaître, dans des conditions données, des événements critiques ou catastrophiques.

La fiabilité, la maintenabilité, la disponibilité et la sécurité sont ainsi des éléments permettant de quantifier la sûreté de fonctionnement. Nous allons maintenant expo-

ser différentes stratégies de maintenance, ainsi que leurs avantages et inconvénients.

1.1.3 Politiques de maintenance

L'optimisation des stratégies de maintenance est un point clé de toute industrie, pour des raisons économiques, sécuritaires, et réglementaires. Il peut s'agir d'actions d'entretien du système, ou de renouvellements. Les différentes actions sont regroupées suivant le schéma 1.3.

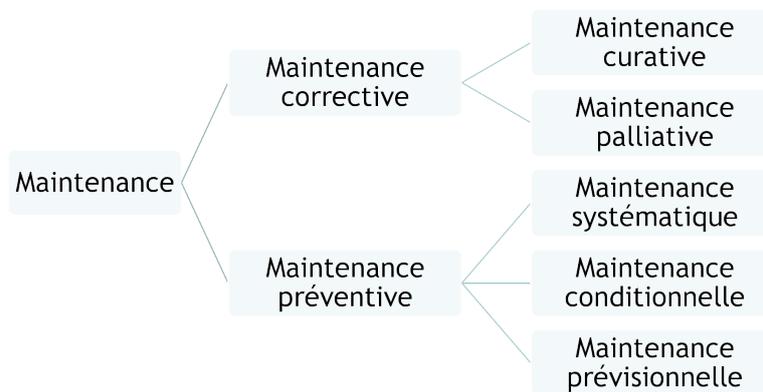


FIGURE 1.3 – *Politiques de maintenance*

La maintenance corrective est une maintenance effectuée après la détection d'une défaillance. Elle vise à rétablir un état opérationnel que le système avait quitté. La maintenance palliative est une solution provisoire, alors que la maintenance curative a pour objet de rétablir le système dans un état de fonctionnement. La maintenance palliative doit être suivie d'une maintenance curative. Faire le choix d'une maintenance corrective entraîne nécessairement une période d'indisponibilité du système.

La maintenance préventive, quant à elle, consiste en un entretien qui permet de maintenir les équipements en bon état de fonctionnement, en prévoyant l'inspection, la détection et la correction des défaillances naissantes, soit avant qu'elles ne se produisent, soit avant qu'elles ne s'aggravent et conduisent à une panne.

La maintenance systématique désigne des opérations effectuées systématiquement, soit selon un calendrier, avec une périodicité temporelle fixe, soit selon une périodicité d'usage, telle qu'un nombre de pièces produites, ou encore un nombre d'heures de fonctionnement. Cette maintenance nécessite de choisir pertinemment la périodicité, qui soit un bon compromis entre un trop grand nombre de maintenances qui coûterait cher, et un trop petit nombre de maintenance qui n'assurerait pas de façon certaine la disponibilité du matériel.

La maintenance conditionnelle est effectuée après que l'information renvoyée par un ou plusieurs indicateurs indique que l'équipement va bientôt tomber en panne ou que ses performances se détériorent. L'enjeu est ici de déterminer des seuils pertinents d'alerte, de définir à partir de quel niveau d'information de l'indicateur il est nécessaire d'intervenir. Si cette maintenance a l'avantage d'agir au plus près de la défaillance, elle présente l'inconvénient de ne pas pouvoir prévoir à l'avance la charge de maintenance (comme les moyens à mettre en œuvre ou le personnels à mobiliser), qui n'est connue qu'au moment du dépassement du seuil de l'indicateur.

La maintenance prévisionnelle cherche quant à elle à prédire à l'avance l'instant de défaillance, ou tout au moins l'instant de dépassement d'un seuil de dégradation conduisant à un déclenchement d'une maintenance préventive conditionnelle. Ceci permet d'agir au plus près de la défaillance, ou juste avant l'événement redouté, et ainsi de gagner en disponibilité et en coût d'entretien. Cela permet d'anticiper la charge de maintenance, ce qui peut aussi être un gain pour une entreprise.

L'élaboration d'une politique de maintenance optimale – par rapport aux critères de la sûreté de fonctionnement et par rapport aux coûts – pour une situation donnée constitue un problème scientifique et technique majeur. Il y a notamment un consensus fiabilité-maintenabilité à trouver.

La connaissance disponible sur le système peut aider à élaborer une stratégie de maintenance. Nous allons donc d'abord présenter un processus d'élaboration d'une maintenance optimale, puis nous nous intéresserons à l'étape sur laquelle les travaux ont porté : la modélisation du processus de dégradation.

1.2 Processus de dégradation

1.2.1 Maintenance basée sur la fiabilité

Un moyen pour établir une maintenance avec une fiabilité et une disponibilité requises, au coût le plus bas, est d'utiliser ce qu'on appelle la maintenance basée sur la fiabilité (MBF).

La première description générale de la MBF a été proposée par NOWLAN [NOWLAN et HEAP 1978]. L'ouvrage de ZWINGELSTEIN [ZWINGELSTEIN 1996] est une des principales références en français au sujet de la MBF.

C'est une méthode essentiellement basée sur la connaissance exacte du comportement fonctionnel et du dysfonctionnement des systèmes. L'ensemble des connaissances disponibles sur la fiabilité du système, l'historique de ses défaillances, sont étudiés, afin d'effectuer une analyse détaillée des modes de défaillance et de leurs effets. L'approche MBF inclut aussi des calculs de la probabilité de défaillance et

de fiabilité du système. La tâche de maintenance est ainsi dimensionnée en fonction des conséquences des défaillances, et non par rapport aux défaillances proprement dites. Il s'agit alors de comprendre comment le dysfonctionnement se produit et de trouver les actions de maintenance nécessaires pour que les phénomènes identifiés ne se produisent pas.

L'analyse MBF permet de déterminer les tâches de maintenance appropriées, c'est-à-dire le plan de maintenance le plus pertinent, pour traiter chacun des modes de défaillance identifiés ainsi que leurs conséquences. Elle permet d'optimiser la disponibilité des équipements avec le coût le plus bas possible, au niveau de fiabilité requis. L'opérateur évite ainsi la sur-qualité (coûteuse) et la sous-qualité (à l'origine des défaillances).

La mise en œuvre d'une telle démarche pour un système donné nécessite toujours :

1. un modèle représentant le ou les processus de dégradation du système ;
2. un modèle représentant les différentes solutions de maintenance envisageables ;
3. une fonction d'utilité pour évaluer chaque politique de maintenance réalisable.

L'objectif de l'approche MBF est finalement de maximiser la fonction d'utilité en ajustant les paramètres de maintenance pour réaliser le meilleur compromis possible entre les coûts d'entretien et la disponibilité du système.

La modélisation du système est généralement l'étape la plus délicate de la MBF. L'objectif est d'expliquer au mieux le processus d'évolution des différents états de fonctionnement du système. L'éventuel Retour d'EXpérience (REX) facilite la compréhension des processus de dégradation des composants du système et de leurs interactions. La présence et l'exploitation d'un REX présentent également l'avantage de fournir une modélisation relativement proche de la réalité. En revanche, s'il n'y a pas ou peu de données REX exploitables, il appartient aux experts de proposer leur modèle, ce qui peut conduire parfois à l'introduction de biais importants. Le reste de la méthode dépend des choix de l'utilisateur, comme expliqué dans la section 2.1.4.

1.2.2 Éléments bibliographiques

Une étape importante de la MBF est donc la modélisation du processus de dégradation du système ou de ses composants, appelée également analyse de fiabilité.

Les travaux ayant porté sur ce sujet étant très denses, l'état de l'art sera orienté dans le cadre d'étude de cette thèse, qui se positionne par rapport à des travaux antérieurs tous basés sur des systèmes à espace d'états discrets et finis. Il existe aussi

un grand nombre de méthodes à espace d'état continu, mais qui ne font pas l'objet de cette thèse.

Pour modéliser la défaillance d'un système à espace d'état discrets et finis, il existe deux grandes classes de modèles : les modèles de survie et les modèles de dégradation.

Établir un modèle de survie consiste à décrire la durée de fonctionnement de l'entité, du composant, par une loi de probabilité [LAWLESS 2011 ; NELSON 2004].

Toutes les lois statistiques sont envisageables. La plus connue est la loi exponentielle. Cette loi modélise la durée de vie d'un phénomène sans mémoire, ou sans vieillissement, ou sans usure : le taux de défaillance ne dépend pas de l'âge du système. La loi exponentielle est pertinente par exemple dans le domaine de la radioactivité [RUTHERFORD et SODDY 1903]. On rencontre aussi souvent la loi de Weibull [WEIBULL 1951], qui présente l'avantage de pouvoir modéliser chacune des trois phases de défaillance de la courbe en baignoire (voir section 1.1.2) en fonction des valeurs de son paramètre de forme [SINGPUREWALLA et SHI SONG 1988]. La loi Gamma [LAPLACE 1836] permet également de modéliser les trois phases de la vie d'un système (cf. figure 1.2) mais elle présente une décroissance trop rapide pour beaucoup d'applications. La loi Log-normale [FRANCIS 1879], quant à elle, modélise la durée de vie par une courbe en forme de cloche. Elle ne représente que la première et la dernière phases de la vie d'un système. La loi de Bertholon [BERTHOLON 2001] présente l'avantage de mixer une loi exponentielle et une loi de Weibull et donc permet de modéliser dans un même contexte la phase de maturité et la phase de vieillesse d'un système.

Il existe aussi des modèles de survie prenant en compte la présence de plusieurs variables, ce sont les modèles à hasard proportionnel [KAY 1977], tel que le modèle de Cox [COX 1972], qui peut être employé lorsqu'on cherche à évaluer l'effet de certains facteurs, appelés covariables, sur la durée de survie, et que l'on dispose d'un jeu de données dans lequel les événements redoutés sont rares.

Toutefois, les modèles de survie ne permettant pas de décrire l'évolution de l'entité vers la panne, la dynamique de dégradation ne peut être étudiée avec de tels modèles. Ils ne prennent pas en compte les états intermédiaires avant la panne. Pour modéliser la dégradation d'un système, la classe des processus de Lévy a souvent été retenue du fait des accroissements indépendants et stationnaires de ces processus. Ils permettent de modéliser l'évolution de la dégradation au cours du temps. Ils peuvent s'appliquer pour garantir un niveau minimum de fiabilité, ou pour déterminer quand remplacer un composant. Cette classe englobe en particulier des processus largement étudiés dans la littérature [COCOZZA-THIVENT 1997 ; ROSS 1996] dont le

mouvement brownien ou processus de Wiener [WIENER 1923], le processus gamma [ABDEL-HAMEED 1975], le processus de Poisson [POISSON 1837], et le processus gaussien.

Ces processus ne permettent cependant pas de modéliser la dynamique de dégradation de systèmes dont le comportement est plus compliqué que la somme des comportements des composants du système, appelés systèmes complexes. De tels systèmes mélangent les phénomènes, les dynamiques, ils sont décrits par des mélanges de lois. Pour pouvoir déterminer comment le fonctionnement ou la panne de chaque composant élémentaire va impacter le système global, il est possible de réaliser une Analyse des modes de défaillance, de leurs effets et de leur criticité (AMDEC). Cet outil ne permet toutefois pas de tenir compte de la présence de plusieurs pannes sur des composants élémentaires. Pour s'adapter à de telles structures, des méthodes complémentaires ont été mises en place : les diagrammes de fiabilité [PAGÈS et GONDRAN 1980], les réseaux de pétri [PETRI 1962], les chaînes de Markov [MARKOV 1971], les arbres de défaillance [WATSON 1961], les réseaux de neurones [JIANG et al. 2011 ; MCCULLOCH et PITTS 1943], et les réseaux bayésiens [JENSEN 1996], qui vont être présentés dans le chapitre 2.

Ces outils de modélisation du processus de dégradation sont présentés dans plusieurs ouvrages de référence, tel que [RAUSAND et HØYLAND 2004 ; ZIO 2007].

Il a été montré que les réseaux bayésiens, que nous présenterons en détail par la suite, sont une généralisation des chaînes de Markov, des réseaux de pétri, des arbres de défaillance. Il est donc apparu intéressant d'utiliser ce formalisme, d'autant plus que ce formalisme est de plus en plus rencontré dans le domaine de la fiabilité.

C'est donc dans le cadre plus général des réseaux bayésiens qu'il a été choisi de travailler pour cette thèse.

Le chapitre suivant présente donc le formalisme des réseaux bayésiens, en mettant l'accent sur les modèles graphiques de durée, et sur l'inférence au sein du réseau.

Réseaux bayésiens

Nous avons vu que les réseaux bayésiens sont un formalisme pertinent pour modéliser la dégradation de systèmes à espace d'états discrets. L'objectif de ce chapitre est de présenter les réseaux bayésiens, et plus particulièrement le modèle graphique de durée.

Nous allons d'abord exposer brièvement des généralités sur les réseaux bayésiens, puis sur les réseaux bayésiens dynamiques. Ensuite nous présenterons les modèles graphiques de durée, qui sont un cas particulier des réseaux bayésiens dynamiques. Nous introduirons ensuite brièvement des approches permettant d'apprendre un réseau bayésien, et enfin nous expliquerons ce qu'est l'inférence, et comment l'effectuer.

2.1 Généralités sur les Réseaux Bayésiens

2.1.1 Historique

Les réseaux bayésiens (RB) ont été nommés ainsi par Pearl en 1985. Ils se situent à l'intersection entre la théorie des graphes [GROSS et al. 2004] et la théorie des probabilités [RAO 1984]. Les premières avancées théoriques ont été constituées par les travaux de Judea Pearl à l'Université de Californie à Los Angeles et de Finn Jensen à l'Université d'Aalborg [JENSEN 1996 ; PEARL 1985]. Les développements de l'usage des réseaux bayésiens sont directement liés à l'augmentation de la puissance de calcul des ordinateurs.

Dès leur mise en place, les réseaux bayésiens ont donné lieu à de nombreuses applications opérationnelles, telles que le décodage du génome, dans le cadre du projet Human Genome [BECKER et al. 1998 ; NAÏM et al. 2007], ou le traitement de la parole [BACH et JORDAN 2005]. Le développement de méthodes basées sur les réseaux bayésiens doit par ailleurs beaucoup à la section recherche de Microsoft, qui y a beaucoup contribué dans le milieu des années 1990 [HECKERMAN 1995]. Microsoft a ainsi mis en place son célèbre "trombone", dont le fonctionnement sera expliqué en 2.1.3, après une présentation des réseaux bayésiens et de leur fonctionnement.

2.1.2 Définition générale

Les réseaux bayésiens [JENSEN 1996] sont des outils mathématiques reposant sur la théorie des probabilités et la théorie des graphes. Ce sont des modèles graphiques probabilistes qui permettent de représenter intuitivement la distribution d'un ensemble de variables aléatoires. Ils représentent graphiquement loi jointe d'un ensemble de variables aléatoires, exploitant la relation d'indépendance conditionnelle entre ces mêmes variables aléatoires. C'est une représentation factorisée de cette loi jointe. Le formalisme des réseaux bayésiens sera explicité plus en détail dans la partie théorique du chapitre, dans la section 2.2.

Les trois points clefs de la mise en place d'un réseau bayésien sont l'apprentissage de la structure, l'apprentissage des probabilités et enfin l'inférence. L'apprentissage de la structure consiste à définir les nœuds et les arcs, c'est à dire les variables et les liens parents-enfants. Ceci ne fera pas l'objet d'une présentation détaillée dans le cadre de cette thèse. L'apprentissage des probabilités permet de quantifier les relations entre les variables définies par la structure apprise précédemment. Les choses seront présentées plus en détail dans la section 2.2.4. Enfin, l'inférence consiste à propager l'information dans le réseau bayésien. L'inférence permet de calculer les probabilités marginales des variables, c'est à dire les probabilités a priori. Elle permet aussi, à partir d'une connaissance partielle ou totale de certains éléments du graphe, d'affiner la connaissance d'autres éléments du graphe, en propageant de proche en proche l'information disponible. L'inférence fera l'objet de la section 2.2.5.

Avant de présenter les éléments théoriques et le formalisme des réseaux bayésiens, listons quelques applications dans la section suivante.

2.1.3 Utilisations des réseaux bayésiens

Les réseaux bayésiens sont applicables à de nombreux domaines, ils ont de nombreuses utilisations, que ce soit en santé [BELLOT 2002; PEREZ 2010], en reconnaissance des formes [JONQUIÈRES 2000], en gestion du risque [BOUEJLA 2014], en sciences du langage [DEVIREN et al. 2004], en agro-alimentaire [PINAUD et al. 2008], en agriculture [KRISTENSEN et RASMUSSEN 2002], ou encore en écologie [POLLINO et al. 2007].

Ils permettent de représenter les liens de dépendance entre les variables d'un système. Grâce à leur utilisation, il est possible d'affiner les connaissances de certaines variables, et donc d'exploiter les données disponibles. Le traitement des données, par la construction et l'utilisation d'un réseau bayésien, permet de découvrir de nouvelles informations issues de ces données, d'en tirer de nouvelles connaissances.

Les illustrations suivantes permettront d'en comprendre l'utilité et le fonctionnement.

Compagnon Office

Le produit Office Assistant, ou plus communément appelé trombone de Microsoft, a été intégré à Office à partir de la version 97 jusqu'à la version 2003. Il était l'un des premiers assistants intelligents qui interagissait avec l'utilisateur en temps réel. Cet outil fouillait pour lui les fichiers d'aide.

Une des fonctions du trombone était de répondre à une question posée par l'utilisateur, en proposant une ou plusieurs réponses, avec les instructions correspondantes.

Une autre fonction du trombone était d'"observer" les actions effectuées, et de proposer des astuces à l'utilisateur pour améliorer son usage de l'application, en lui pré-réalisant par exemple des tâches de bureautique, ou en le conseillant sur l'utilisation des fonctionnalités de bureau, en lui présentant des astuces et des raccourcis clavier.

Le système d'aide était fondé sur les réseaux bayésiens. Les informations recueillies pendant l'interaction de l'utilisateur avec le système, comme les mots écrits, les pauses effectuées, un ralentissement de la tâche, un clic sur la touche "retour", ou une recherche d'action, étaient transformés en observations pour le réseau bayésien par le modèle de synthèse. Le module d'inférence produisait ensuite les décisions, et enfin, le module de contrôle les exécutait [KADIE et al. 2001].

Filtre antispam

Le filtre antispam sert à ce que des courriers électroniques indésirables soient automatiquement reconnus comme tels, et rangés dans le dossier de spams.

Le filtre calcule en fait la probabilité qu'un message soit un spam, en regardant s'il vérifie ou non un certain nombre de propriétés, comme par exemple l'adresse mail d'origine, les fautes d'orthographe, des mots-clefs, et le nombre de destinataires. Les propriétés à vérifier constituent les différents noeuds du réseau bayésien. Les différentes probabilités conditionnelles sont mises à jour au fur et à mesure de l'utilisation de la messagerie par l'utilisateur, à partir de ses choix précédents au sujet classement passé de messages antérieurs, Si le calcul de la probabilité qu'a le nouveau message reçu d'être un spam dépasse un certain seuil, alors le message sera considéré comme spam [SAHAMI et al. 1998].

Autres exemples

Pour établir un diagnostic médical, le médecin analyse les symptômes. Il pose des questions en fonction de ce à quoi il pense comme diagnostic, pour écarter ou confirmer les causes les plus probables. Puis en fonction des réponses obtenues, il pose de nouvelles questions, peut demander un examen, en tenant compte de l'impact, des conséquences qu'il pourrait avoir, comme une biopsie par exemple [BELLOT 2002]. Cet exemple fait aussi appel à une autre utilisation des réseaux bayésiens, à savoir l'aide à la décision. Les risques des différentes conséquences de décisions sont alors estimés, pour aider l'utilisateur à faire le choix le plus optimal, en fonction du poids qu'il attribue à chaque conséquence.

Les réseaux bayésiens ont de nombreuses autres utilisations, comme notamment la simulation, l'évaluation, l'analyse de données, le contrôle des systèmes, la prédiction, la résolution de contraintes, et la reconnaissance de formes [NAÏM et al. 2007].

Ils ont notamment été utilisés par le GRETTIA dans le cadre du développement d'un modèle de maintenance, qui est présenté dans la partie suivante. C'est dans ce contexte que s'inscrit cette thèse.

2.1.4 Contexte d'utilisation des réseaux bayésiens au laboratoire GRETTIA

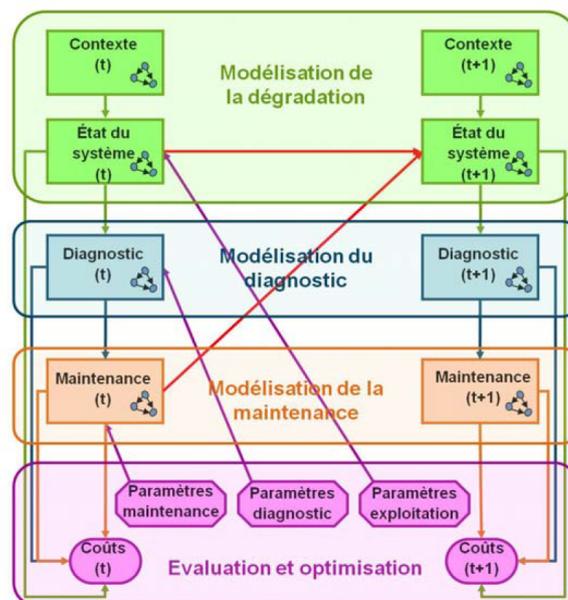


FIGURE 2.1 – Approche générique pour développer des modèles de maintenance

Ces travaux font suite à de précédentes études, qui ont développé successivement différentes particularités de réseaux bayésiens pour mettre en place un formalisme permettant d'évaluer, de comparer, d'optimiser diverses stratégies de maintenance. Ce formalisme s'appuie sur la maintenance basée sur la fiabilité (voir section 1.2.1) qui nécessite un module de dégradation, un module de maintenance, et un module d'utilité (figure 2.1). L'outil développé par le GRETTIA s'appuie sur les réseaux bayésiens pour chacun des modules. Il s'appelle VirMaLab [BOUILLAUT et al. 2011], pour Virtual Maintenance Laboratory (figure 2.2).

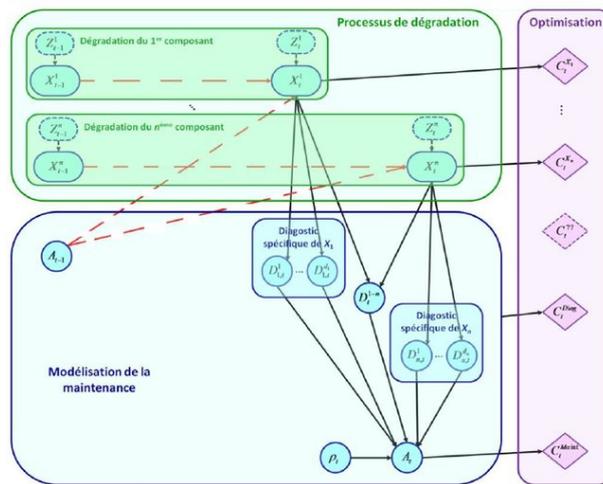


FIGURE 2.2 – Approche VirMaLab

La dégradation du système est modélisée par un réseau bayésien, ainsi que le diagnostic de l'état du système, les actions de maintenance, et le coût de la maintenance, l'aspect modulaire des réseaux bayésiens permettant de tout regrouper sous le même formalisme.

Le module de dégradation est constitué des différentes variables qui peuvent avoir un impact sur la dégradation du système. Il y a l'état de fonctionnement du système, il peut y avoir le contexte : s'il s'agit d'une machine de production, il peut y avoir la cadence de production, s'il s'agit de rails, il peut y avoir par exemple la charge supportée par la zone étudiée, ou encore le rayon des courbes.

Le module de diagnostic est constitué d'un ou plusieurs capteurs. Il existe plusieurs types de diagnostics, comme par exemple les diagnostics visuels, les ultrasonores, ou encore ceux issus de la signalisation. La table de probabilité associée est constituée du taux de fausse alarme et du taux de bonne détection.

Le module de maintenance va dépendre du type de maintenance considéré. Dans le cas de la maintenance corrective, la maintenance est activée si le diagnostic indique que le système est en panne. Dans le cas de la maintenance préventive conditionnelle, la maintenance est activée si le diagnostic indique qu'un des paramètres a dépassé un

certain seuil. Dans le cas de la maintenance préventive systématique, la maintenance est activée selon un échéancier. Si le pas de temps est régulier, la maintenance est activée si le temps écoulé depuis la dernière maintenance est égal à ce pas. La mise en place de la maintenance préventive prévisionnelle demande une estimation au fil de l'eau de la RUL (RemainingUseful Life) pour faire du pronostic [FOULLIARON 2015].

Après avoir présenté d'une part quelques applications des réseaux bayésiens et d'autre part l'utilisation des réseaux bayésiens au sein du modèle de maintenance développé par le GRETTIA, nous allons présenter leur formalisme. Enfin, nous reviendrons sur les travaux de [FOULLIARON 2015].

2.2 Formalisme des réseaux bayésiens

2.2.1 Définitions et généralités

Un réseau bayésien [JENSEN 1996] [PEARL 1985] [NAÏM et al. 2007] est un graphe orienté sans circuit, dans lequel chaque nœud représente une variable du système, et chaque arc représente une relation de dépendance directe entre deux variables. À chaque nœud est associée une loi de probabilité conditionnelle sachant ses parents.

Un réseau bayésien discret est donc défini par un couple $\mathcal{M} = (\mathcal{G}, (p_n)_{1 \leq n \leq N})$ avec

- \mathcal{G} un graphe orienté sans circuit : $\mathcal{G} = (\mathbf{X}, \mathcal{E})$ avec $\mathbf{X} = (X_1, \dots, X_N)$ l'ensemble des nœuds associé chacun à une variable aléatoire X_i dans un espace fini et dénombrable Ω_{X_i} et \mathcal{E} l'ensemble des arcs, c'est-à-dire les dépendances entre les variables aléatoires
- $(p_n)_{1 \leq n \leq N}$ un ensemble de lois de probabilité conditionnelles (LPC) telles que p_n désigne la loi de probabilité conditionnelle associée à la variable aléatoire X_n sachant ses parents X_{pa_n} , pa_n se référant à la séquence des indices parents de la variable aléatoire X_n dans \mathcal{G}

Pour illustrer cela, considérons une machine de production à trois états de fonctionnement. À cette machine est associée un capteur, qui renseigne sur l'état de fonctionnement de la machine. Plusieurs cadences de production sont possibles. Soit X une variable discrète modélisant l'état du système, O une variable discrète modélisant l'information envoyée par capteur, Z une variable modélisant la vitesse de production. X peut être « ok », « dégradé » pour les états de fonctionnement et « en panne ». O peut être « ok », « dégradé », « en panne ». Z peut être « rapide » ou « lent » (cf figure 2.3).

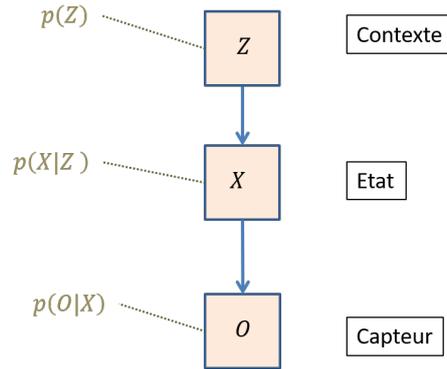


FIGURE 2.3 – Exemple de réseau bayésien

Les relations d'indépendance conditionnelle modélisées par les arcs du graphe permettent de factoriser la distribution de probabilité conjointe de l'ensemble des variables aléatoires \mathbf{X} comme suit :

$$P(\mathbf{X}) = P(X_1, \dots, X_N) = \prod_{n=1}^N P(X_n | pa_n) \quad (2.1)$$

L'exemple de la figure 2.3 donne la loi jointe suivante :

$$P(Z, X, O) = P(Z)P(X|Z)P(O|X) \quad (2.2)$$

Si cette démarche-là répond à beaucoup de besoins, elle ne tient absolument pas compte de phénomènes dynamiques. C'est ce qui a conduit au développement des réseaux bayésiens dynamiques, qui sont l'objet de la prochaine section.

2.2.2 Réseau bayésien dynamique

Lorsque certaines variables varient au cours du temps, il a été proposé une extension des réseaux bayésiens, appelée réseaux bayésiens dynamiques (RBD) [MURPHY 2002]. Les RBD ont permis d'unifier de nombreuses approches à partir de la modélisation de séries temporelles, telles que les modèles de Markov cachés.

Un RBD vise à modéliser la distribution de probabilité d'un ensemble de variables aléatoires $(X_t)_{1 \leq t \leq T} = (X_{1,t}, \dots, X_{N,t})_{1 \leq t \leq T}$.

Il se compose d'une paire de réseaux bayésiens $(\mathcal{M}_1, \mathcal{M}_\rightarrow)$:

\mathcal{M}_1 définit la distribution a priori, le modèle initial $P(X_{1,1}, \dots, X_{N,1})$.

\mathcal{M}_\rightarrow définit le modèle de transition qui décrit les dépendances entre les variables dans les tranches précédentes et les variables dans la tranche t , i. e. la distribution de $X_t | X_1, \dots, X_{t-1}$.

On appelle n -DBN un réseau bayésien dynamique d'ordre n , c'est à dire qui comporte $n + 1$ tranches de temps. La dépendance conditionnelle de X_t par rapport aux tranches de temps précédentes s'arrête à la tranche $t - n$. X_t est indépendant de X_{t-n-1} conditionnellement à X_{t-n}, \dots, X_{t-1} [WEBER et al. 2004].

$\mathcal{M}_{\rightarrow}$ définit le modèle de transition qui décrit la distribution $X_t|X_{t-1}, \dots, X_{t-n}$ qui vérifie :

$$\begin{aligned} P(X_t|X_{t-1}, \dots, X_{t-n}) &= P(X_{1,t}, \dots, X_{N,t}|X_{1,t-1}, \dots, X_{N,t-1}, \dots, X_{1,t-n}, \dots, X_{N,t-n}) \\ &= \prod_{n=1}^N P(X_{n,t}|pa_{X_{n,t}}) \end{aligned} \quad (2.4)$$

où $pa_{X_{n,t}}$ sont les indices des variables parentes de la variable $X_{n,t}$ dans le graphe de $\mathcal{M}_{\rightarrow}$.

Un réseau bayésien dynamique à trois tranches de temps est illustré par la figure 2.4.

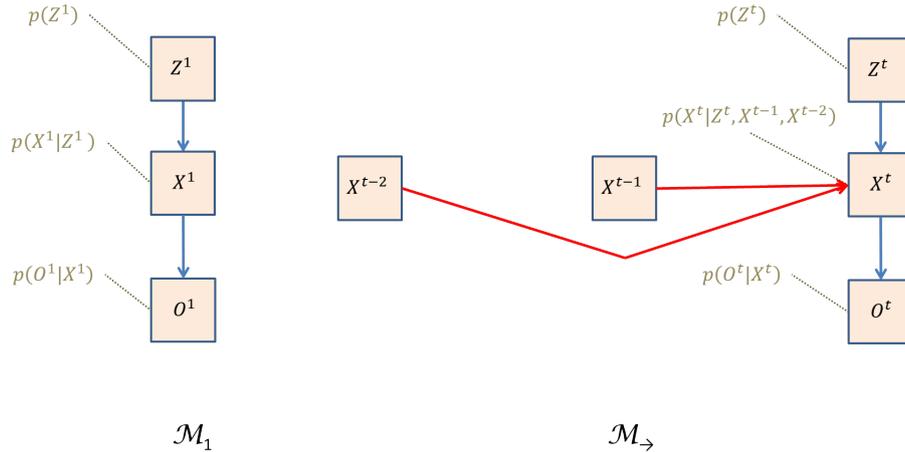


FIGURE 2.4 – Réseau Bayésien Dynamique à trois tranches de temps

La figure 2.5 présente un exemple de RBD à deux tranches de temps.

Au delà des ouvrages de référence [MURPHY 2002 ; NAÏM et al. 2007], le lecteur pourra se pencher sur la thèse de Roland Donat [DONAT 2009] qui présente les réseaux bayésiens statiques et dynamiques de manière pédagogique.

Un réseau bayésien dynamique est en réalité un réseau bayésien statique, répété plusieurs fois. Le RBD hérite donc des propriétés pratiques du RB statique, notamment en ce qui concerne l'apprentissage et l'inférence. Il n'y aura ainsi pas de distinction entre les deux dans la suite.

Comme évoqué dans la section 2.1.3, il existe de nombreuses applications aux réseaux bayésiens et aux réseaux bayésiens dynamiques.

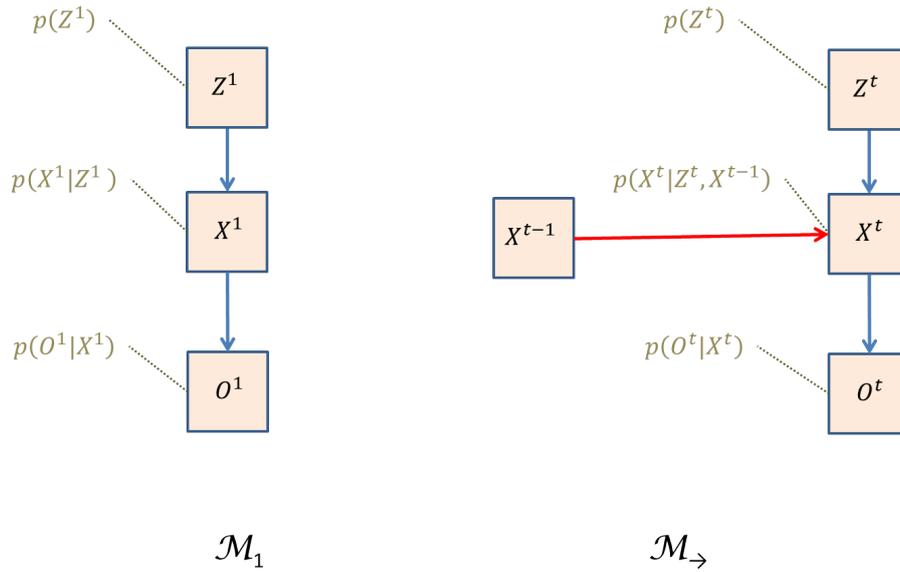


FIGURE 2.5 – Réseau Bayésien Dynamique

Modélisation du processus de dégradation par un réseau bayésien

Des travaux récents ont ainsi montré l'intérêt d'utiliser les réseaux bayésiens dans le domaine de la fiabilité. WEBER explique ainsi comment utiliser les réseaux bayésiens dynamiques pour étudier la fiabilité d'un système dynamique modélisé par une chaîne de Markov [BEN SALEM et al. 2006; WEBER et JOUFFE 2003]. Cette méthode permet de prendre en compte l'influence du temps ou de variables exogènes sur les modes de dégradation du système. Il montre que la structure graphique des réseaux bayésiens dynamiques permet, grâce à la simplicité de spécification des dépendances, de fournir une représentation compacte du modèle.

BOUDALI montre comment modéliser la fiabilité d'un système complexe en utilisant des réseaux bayésiens [BOUDALI et DUGAN 2005]. Il a proposé une intégration de l'aspect dynamique en transformant les arbres de défaillance dynamique en réseaux bayésiens dynamiques.

LANGSETH, lui, décrit les propriétés du cadre d'applicabilité des réseaux bayésiens à la sûreté de fonctionnement au sens large.

Application à la fiabilité et à la disponibilité

L'utilisation des RBD permet d'étudier la fiabilité d'un système. En effet, la fiabilité s'écrit (cf. section 1.1.2)

$$R(t) = P(X_1 \in \mathcal{U}; \dots; X_t \in \mathcal{U}). \quad (2.5)$$

avec \mathcal{U} l'ensemble des états de fonctionnement.

L'estimation de la fiabilité est un calcul d'inférence.

Notons que dans notre cas d'étude, le système ne peut pas se réparer lui-même. L'état de défaillance est supposé absorbant. La fiabilité est alors égale à la disponibilité :

$$R(t) = P(X_t \in \mathcal{U})$$

Le calcul de fiabilité est alors réduit au calcul de la distribution marginale de X_t . Ce calcul d'inférence comporte moins d'étapes de calculs que l'estimation de la loi jointe de X_1, \dots, X_t donné par 2.5.

L'estimation d'une fiabilité, ou d'une disponibilité est un calcul qui peut se révéler nécessaire.

En effet, sous estimer la fiabilité, *i.e.* estimer une fiabilité inférieure à la fiabilité réelle, c'est calculer une $P(X_t \neq \text{panne})$ inférieure à la réalité, donc une $P(X_t = \text{panne})$ trop élevé. L'action de maintenance va donc être commandée, alors que ce n'était pas nécessaire. Cela conduit à des politiques de maintenance onéreuses. À l'inverse, surestimer la fiabilité, c'est sous-estimer la probabilité que le système tombe en panne, l'ordre de maintenance n'est alors pas donné, et le système souffre donc d'une perte de disponibilité.

Estimer la fiabilité d'un système à l'aide d'une modélisation par un réseau bayésien dynamique, puis la comparer avec la fiabilité théorique du système, permet d'effectuer une évaluation de la modélisation. Cela sera utilisé dans la partie 4.4.

Limites des réseaux bayésiens dynamiques

La factorisation de la loi jointe permise par les réseaux bayésiens dynamiques leur confère une grande puissance de calcul.

Toutefois, un réseau bayésien dynamique d'ordre n , c'est à dire à $n + 1$ tranches consécutives de temps, suit la propriété de Markov d'ordre n . En effet, l'état du système au temps $t + 1$, X_{t+1} , est déterminé entièrement par les états du système aux temps $t, \dots, t - (n - 1)$, c'est à dire X_t, \dots , et $X_{t-(n-1)}$ et ne dépend pas de l'état à des instants plus tôt $1, \dots, t - n$. L'état du système au temps $t + 1$, X_{t+1} , ne dépend que des n états précédents.

Dans la suite, nous nous placerons dans un contexte à deux tranches de temps, avec un réseau bayésien dynamique d'ordre 1. Il s'agit du cas le plus général. En effet, un réseau bayésien dynamique d'ordre n peut se ramener à un réseau bayésien dynamique d'ordre 1, en rassemblant n tranches de temps dans un vecteur. Le

vecteur comportant les états aux temps $t+1, \dots, t-(n-2)$ est déterminé entièrement par le vecteur comportant les états aux temps $t, \dots, t-(n-1)$.

Or, dans un modèle qui suit la propriété de Markov d'ordre 1, l'état du système au temps $t+1$ ne dépend que de l'état du système à l'instant t , il ne dépend pas de la valeur de cet instant t . Le taux de défaillance est alors constant, ce qui signifie que le système est toujours dans une phase de maturité (cf. figure 1.2) et section 1.1.2). Un taux de défaillance constant ne convient toutefois pas pour de nombreux systèmes, tels que les composants mécaniques [LANTERNIER et al. 2006] ou encore les composants électroniques [RINGLER 1982]. Une modélisation markovienne pouvant donc introduire des biais non négligeables dans les estimations, il était nécessaire d'utiliser un autre type de modèle.

Dans un modèle markovien, le temps de séjour dans un état suit nécessairement une loi géométrique, comme le montre l'équation 2.6, en posant S_1 le nombre de tranches où le système reste dans le même état que X_1 . En posant $p_{i \rightarrow i}$ la probabilité que le système au temps $t+1$ soit dans l'état i sachant qu'il est dans l'état i au temps t , on a :

$$\begin{aligned} P(S_1 = k | X_1 = i) &= P(X_2 = i \dots, X_k = i, X_{k+1} \neq i | X_1 = i) \\ &= P(X_{k+1} \neq i | X_k = i) P(X_k = i | X_{k-1} = i) \dots P(X_2 = i | X_1 = i) \\ &= (1 - p_{i \rightarrow i}) p_{i \rightarrow i}^{k-1} \end{aligned} \quad (2.6)$$

Ceci conduit à l'utilisation d'un modèle semi-markovien, modèle dans lequel le temps de séjour dans un état peut suivre une loi aléatoire quelconque, comme l'ont montré BARBU, LIMNIOS et OPRISAN [BARBU et LIMNIOS 2004; LIMNIOS et OPRISAN 2001].

2.2.3 Modèle graphique de durée

Une structure originale de réseau bayésien a alors été proposée par le laboratoire. Ces réseaux bayésiens particuliers, qui s'appuient sur les modèles de durée de Murphy [MURPHY 2002], sont appelés modèles graphiques de durée (MGD) [DONAT 2009]. Ils permettent aux temps de séjour de suivre des lois quelconques, afin de s'adapter aux systèmes dont le temps de séjour dans chaque état n'est pas nécessairement distribué exponentiellement.

Le MGD est un réseau bayésien dynamique particulier. A chaque variable d'état du système va être associée une variable temps de séjour dans cet état. Cette variable temps de séjour pourra suivre n'importe quelle loi, tout en suivant une forme

particulière : elle sera décrétementée d'une unité lorsqu'il n'y a pas de changement d'état.

La figure 2.6 présente un modèle graphique de durée : il s'agit du réseau bayésien dynamique de la figure 2.5 auquel ont été ajoutées les variables modélisant le temps de séjour dans l'état de fonctionnement courant.

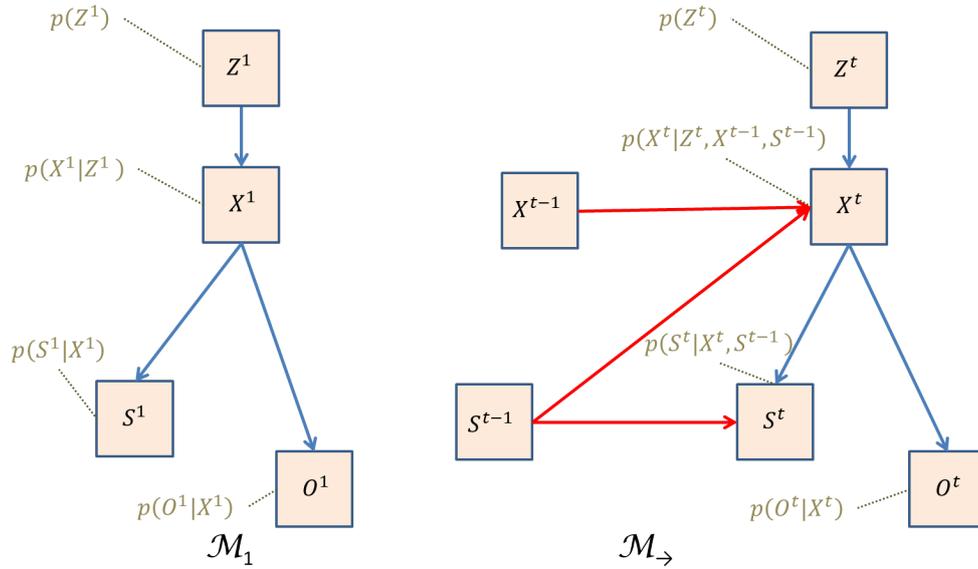


FIGURE 2.6 – Modèle graphique de durée

Pour alléger les notations, les variables de contexte (Z^i) seront écartées dans la suite. La figure 2.7 présente le modèle graphique de durée associé.

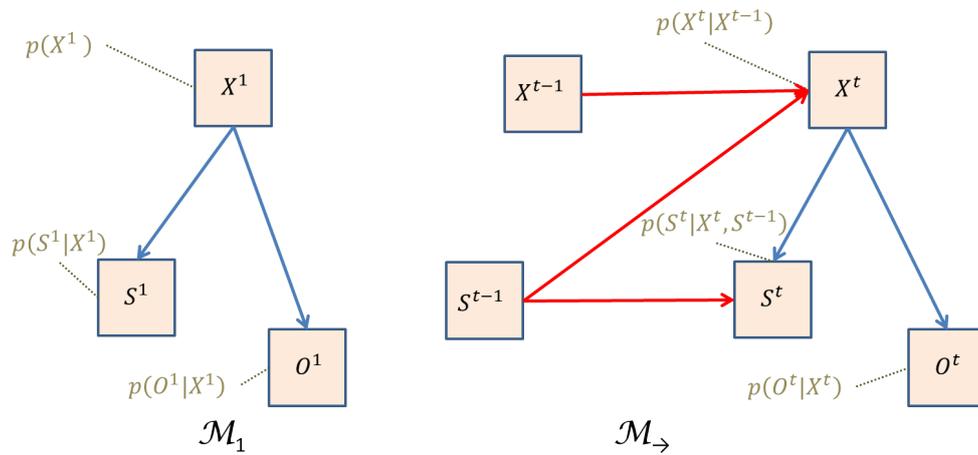


FIGURE 2.7 – Modèle graphique de durée

La distribution de probabilité conditionnelle associée à la distribution de l'état

initial du système est définie sur le domaine discret et fini $\Omega_X = 1, \dots, N_X$ par :

$$P(X_1 = i) = Q^{ini}(i) \quad (2.7)$$

La LPC initiale du temps de séjour donne les distributions pour chaque état initial. Cette LPC est définie sur le domaine discret et fini $\Omega_S = 1, \dots, N_S$ par :

$$P(S_1 = k | X_1 = i) = F^{ini}(i, k) \quad (2.8)$$

Ensuite, il est nécessaire de définir les LPC naturelles des états du système et les LPC des temps de séjour.

Une transition, un changement d'état se produit si et seulement si $S_{t-1} = 1$.

$$P(X_t = j | X_{t-1} = i, S_{t-1} = 1) = Q^{sys}(i, j) \quad (2.9)$$

où Q^{sys} est une matrice $N_X \cdot N_X$ appelée matrice de transition naturelle du système.

Un nouveau temps de séjour est considéré selon la LPC suivante :

$$P(S_t = k | X_t = i, S_{t-1} = 1) = F^{sys}(i, k) \quad (2.10)$$

où F^{sys} est une matrice $N_X \cdot N_S$. Dans certains cas, le temps de séjour S_t dépend également de l'état précédent X_{t-1} , d'où la distinction avec la loi initiale des temps de séjour P_{S_1} . Toutefois, cette dépendance est souvent omise pour éviter des problèmes de sur-paramétrisation, pour réduire la complexité, ou encore lorsqu'il y a peu de données disponibles pour l'apprentissage. On a donc $F^{ini} = F^{sys} = F$.

Lorsqu'il n'y a pas de transition, c'est-à-dire pas de changement d'état, le système reste déterministe dans l'état précédent i .

$$P(X_t = j | X_{t-1} = i, S_{t-1} \geq 2) = I(i, j) \quad (2.11)$$

et le temps de séjour dans l'état actuel est décrémenté de manière déterministe d'une unité :

$$P(S_t = k | X_t = i, S_{t-1} = k' \geq 2) = \begin{cases} 1 & \text{si } k = k' - 1 \\ 0 & \text{sinon} \end{cases} \quad (2.12)$$

2.2.4 Apprentissage

L'utilisation d'un réseau bayésien nécessite d'abord la construction de ce réseau. Cela consiste en la détermination de la structure graphique - nœuds et arcs - et

l'association d'une distribution de probabilité conditionnelle à chaque variable.

Structure graphique

La première étape consiste en l'élaboration de la structure graphique du réseau. Elle comporte les nœuds, qui correspondent aux variables et les différents arcs orientés qui représentent les relations de dépendance conditionnelle entre les variables. Beaucoup de travaux ont porté sur l'apprentissage automatique de structure. La thèse de FRANÇOIS passe en revue la plupart des méthodes existantes en 2008 tout en les comparant empiriquement sur divers exemples classiques à l'aide de bases de données complètes ou incomplètes [FRANÇOIS 2006]. De nouvelles méthodes d'apprentissage de structure sont présentées dans [SCANAGATTA et al. 2015] et dans [O'GORMAN et al. 2015]. L'apprentissage de la structure ne fait cependant pas l'objet de cette thèse. Dans notre cas, tout se fera de manière dite experte : nous définirons nous-même la structure du réseau modélisant le fonctionnement du système. Nous serons par contre plus concernés par l'apprentissage des probabilités détaillé ci-après.

Paramètres probabilistes

La deuxième étape est l'estimation des paramètres du réseau. Appelons $(\theta_1, \dots, \theta_D)$ l'ensemble des paramètres probabilistes à estimer. L'estimation, notée $(\hat{\theta}_1, \dots, \hat{\theta}_D)$ peut être établie à partir d'informations contenues dans une base de données répertoriant un certain nombre de réalisations du processus étudié. Cette base est souvent qualifiée de base d'exemples ou de base de REX (Retour d'EXpérience) dans l'industrie.

La méthode du maximum de vraisemblance [FISHER 1922] consiste à calculer le jeu de paramètres qui maximise la vraisemblance des données observées. La probabilité d'un événement est estimée par la fréquence d'apparition de l'événement dans la base de données. Ceci nécessite que les données soient complètes, notamment que les trajectoires atteignent toutes l'état de panne.

Toutefois, il peut arriver que certaines configurations n'existent pas dans la base d'apprentissage, ce qui conduit à l'estimation d'une probabilité nulle pour ces configurations. Or ce n'est pas parce qu'une configuration n'a pas été observée qu'elle est impossible. L'estimation par le maximum de vraisemblance peut conduire à des situations de sur-apprentissage.

Pour éviter cela, il est possible d'utiliser des *a priori* de Dirichlet, qui permettent d'assurer qu'une probabilité non nulle soit effectivement estimée strictement positive.

La sous-section 4.3.1 présente une méthode d'apprentissage des tables de probabilité à partir de trajectoires, en utilisant la méthode d'estimateur de *maximum a posteriori* [ROBERT 1994] avec *a priori* de Dirichlet uniformes.

En pratique, les bases de données peuvent être incomplètes. La méthode d'apprentissage basée sur l'algorithme d'*Expectation-Maximisation* (EM) permet de faire face au problème des données censurées ou des données manquantes [COWELL et al. 1999 ; DEMPSTER et al. 1977 ; NEAL et HINTON 1998].

Dans de nombreuses situations encore, il n'existe pas (ou très peu) de données, d'historique de défaillance. Il est alors nécessaire de faire appel aux avis d'experts, puis de mettre en place une procédure d'élicitation, pour aider l'expert à formaliser ses connaissances, à retranscrire en probabilités son ressenti, et ainsi permettre de définir certains paramètres [NEAPOLITAN 2003].

NAÏM présente l'ensemble des notions précédentes dans [NAÏM et al. 2007].

Les bases de données de retour d'expérience et les avis d'experts permettent ainsi d'apprendre la structure et les paramètres des lois de probabilité conditionnelles des variables du modèle graphique de durée permettant de modéliser la dégradation du système, ainsi que ceux du module de diagnostic. Le module de maintenance dépend du type de maintenance étudiée. Les paramètres et la structure du module d'utilité sont quant à eux déterminés par l'utilisateur qui souhaite évaluer une certaine stratégie de maintenance. DONAT présente dans [DONAT 2009] plus de détails sur le formalisme décrivant les lois de chacun des nœuds du modèle VirMaLab (2.1.4).

Après avoir présenté l'apprentissage des réseaux bayésiens, intéressons-nous aux calculs probabilistes que ces derniers permettent.

2.2.5 Inférence

Introduction

Les réseaux bayésiens sont particulièrement utiles en raison de la possibilité d'inférer des probabilités à travers eux.

Il est par exemple possible de calculer la probabilité d'une variable conditionnellement à une de ses variables enfants, de calculer les probabilités marginales, et de calculer la probabilité jointe de plusieurs variables. Tous ces calculs sont appelés calculs d'inférence. Ils peuvent être effectués *a priori*, ou encore *a posteriori*, c'est-à-dire en tenant compte de données. Ces données sont appelées observations, dont la définition fera l'objet de la sous-partie suivante.

L'inférence est uniquement un problème informatique dû à la NP-complétude [COOPER 1990]. Il n'y a aucun problème théorique [NAÏM et al. 2007]. Divers algo-

rithmes d'inférence peuvent être utilisés. Il en existe deux sortes : les algorithmes d'inférence exacte et les algorithmes d'inférence approchée.

L'une des procédures d'inférence exacte les plus classiques repose sur l'utilisation d'un arbre de jonction [LAURITZEN et SPIEGELHALTER 1988].

Une autre technique connue est l'algorithme d'élimination de variables [DECHTER 1999], que nous présenterons au chapitre 2.2.5.

Donat a, quant à lui, mis en place un algorithme d'inférence spécifique pour le MGD présenté dans [DONAT et al. 2010].

Néanmoins, dans nos expériences, nous nous concentrerons sur l'usage de l'algorithme d'élimination de variables pour son efficacité et sa facilité de mise en oeuvre, et parce qu'elle est à l'origine de l'algorithme présenté dans la section 3.2.

Il existe également des algorithmes d'inférence approchée, comme par exemple les méthodes variationnelles [JAAKKOLA et JORDAN 1999 ; JORDAN et al. 1999], le filtrage particulière [ARULAMPALAM et al. 2002], ou la méthode d'échantillonnage Markov Chain Monte Carlo [GILKS et al. 1995 ; HASTINGS 1970] .

Comme dit précédemment, les calculs d'inférence peuvent être effectués après l'injection d'informations supplémentaires dans le réseau. Le paragraphe suivant présente différents types d'observations possibles.

Observations

Une observation certaine sur une variable signifie que l'observateur peut affirmer avec certitude que la variable est dans un certain état signalé. Par exemple, $P(X_2|X_1 = ok)$ désigne la distribution de probabilité de X_2 lorsque la variable X_1 est observée dans l'état "ok".

Il est également possible d'avoir ce qu'on appelle des observations incertaines, appelées *soft-evidence* par la communauté anglophone. BEN MRAD présente dans sa thèse [BEN MRAD 2015] différents types d'observations incertaines qu'il est possible de trouver dans un réseau bayésien.

Il peut par exemple y avoir une incertitude sur l'observation de la variable. C'est le cas lorsque l'état de la variable est connu par un capteur, une source, qui n'est pas entièrement fiable. L'observation incertaine peut être remplacée par l'ajout d'un nœud virtuel O qui transforme l'observation incertaine sur X en observation certaine sur O . À l'inverse, il est possible de remplacer le nœud O (c.f. figure 2.7) par de l'observation incertaine sur l'état du système X , et ainsi de simplifier le réseau. Cela peut s'avérer très intéressant dans le cas de plusieurs capteurs.

Nous disposons de la distribution de probabilité $P(O|X)$ qui représente la probabilité de l'information du capteur conditionnellement à l'état réel. La table 2.1

présente un exemple.

TABLE 2.1 – Exemple de table de probabilité conditionnelle de capteur

X_t	P_{capteur}		
	capteur		
	vert	orange	rouge
ok	0.7	0.2	0.1
dégradé	0.1	0.8	0.1
panne	0.1	0.2	0.7

À l'aide du théorème de Bayes, exprimons la probabilité de l'état X conditionnellement à l'observation par le capteur (équation 2.13).

$$\begin{aligned}
 P(X|\text{observation}) &= P(X) * \frac{P(\text{observation}|X)}{P(\text{observation})} \\
 &= P(X) * \frac{P(\text{observation}|X)}{\sum_x P(\text{observation}|X=x) * P(X=x)} \quad (2.13)
 \end{aligned}$$

Ainsi, pour obtenir la probabilité d'un état sachant l'observation, la probabilité marginale de cet état est pondérée par la probabilité d'avoir cette observation sachant cet état, à un coefficient multiplicateur près.

Pour une variable X donnée, la *soft evidence* p_{se} est constituée des probabilités $p_{se}(x)$ d'obtenir l'observation considérée sachant les différents états x . $p_{se}(x) = P(\text{observation de } X \text{ via le capteur } O|X=x)$.

En prenant l'exemple donné par la table 2.1, pour une observation "verte" du capteur, les probabilités de chacun des états du système seront pondérées respectivement par 0.7 0.1 et 0.1.

Les $P(X=x)$ sont pondérés par les $p_{se}(x)$. Une normalisation est effectuée à la fin.

Il est aisé d'imaginer un grand nombre de situation où il est utile de connaître la valeur de l'état du système à un instant t , en disposant d'observations incertaines de X_1, \dots, X_t , via les capteurs.

Il s'agit de calculer $P(X_t|\text{observations})$.

$$\begin{aligned}
 P(X_t = x_t | \text{observations}) &= P(X_t | \text{obs}_1, \dots, \text{obs}_t) \\
 &= \frac{P(X_t, \text{obs}_1, \dots, \text{obs}_t)}{\sum_{x_t} P(X_t = x_t, \text{obs}_1, \dots, \text{obs}_t)} \\
 &\propto P(X_t, \text{obs}_1, \dots, \text{obs}_t) \\
 &\propto \sum_{x_1, \dots, x_{t-1} \in (\Omega_X)^{t-1}} P(x_1, \dots, x_{t-1}, x_t, \text{obs}_1, \dots, \text{obs}_t) \\
 &\propto \sum_{x_1, \dots, x_{t-1} \in (\Omega_X)^{t-1}} \prod_{u=1}^t p_X(x_u | \text{pa}(x_u)) \times P(\text{obs}_t | x_t) \\
 &\propto \sum_{x_1, \dots, x_{t-1} \in (\Omega_X)^{t-1}} \prod_{u=1}^t p_X(x_u | \text{pa}(x_u)) \times p_{se_u}(x_u) \quad (2.14)
 \end{aligned}$$

avec $p_{se_u}(x_u) = P(\text{observation de } X_u \text{ via le capteur} | X_u = x_u)$, l'observation via le capteur de l'état X à l'instant u .

La normalisation sur chacun des états x_t donne ensuite le résultat.

Si une variable X_u n'est pas observée par le capteur, si aucune information n'est disponible pour cet instant, il est possible de prendre par exemple $p_{se_u}(x_t) = P(\text{obs}_t | x_t) = 1$ pour chacun des états de fonctionnement. La valeur choisie a peu d'importance en raison de la normalisation finale, du moment qu'une même valeur est associée à chacun des états pour aboutir à une distribution uniforme.

L'exemple de la figure 2.7 peut maintenant devenir l'exemple de la figure 2.8, la "soft evidence" étant contenue dans la variable X , partiellement observée.

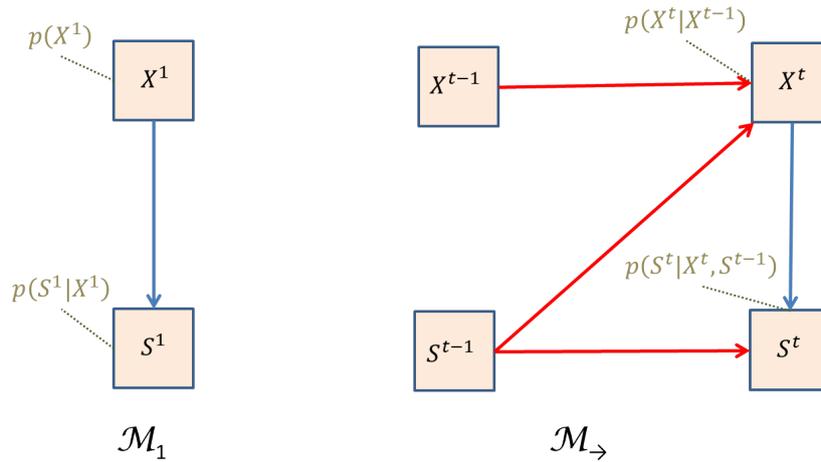


FIGURE 2.8 – Modèle graphique de durée

Pour tenir compte de la pondération possible, servant à inclure d'éventuelles observations incertaines, nous utiliserons dans la suite la notion de potentiel plutôt

que de probabilités. La normalisation des potentiels à la fin d'un calcul renverra une probabilité.

Une fois que la connaissance partielle ou totale de variables est injectée dans le réseau, elle se propage afin d'affiner la connaissance d'autres variables.

Parmi toutes les méthodes d'inférence possibles, nous nous sommes intéressés à l'élimination de variable. D'autres algorithmes d'inférence sont présentés dans [LAURITZEN et SPIEGELHALTER 1988], [DONAT 2009], [JAAKKOLA et JORDAN 1999], [ARULAMPALAM et al. 2002], [GILKS et al. 1995], [HASTINGS 1970], comme présenté en 2.2.5.

Élimination de variables

La méthode d'élimination de variables [DECHTER 1999] a été choisie ici pour sa facilité de compréhension et de mise en place, et parce que l'algorithme d'inférence qui sera présenté dans la section 3.2 s'en inspire, ce qui permettra de comparer des méthodes équivalentes.

La principe de la méthode repose sur le fait que pour tout $\mathbf{x}_r \in \Omega_{\mathbf{X}_r}$

$$P(\mathbf{X}_r = \mathbf{x}_r) = \sum_{\mathbf{x}_e \in \Omega} \prod_{d=1}^D p_d(x_d | pa(x_d)). \quad (2.15)$$

"Éliminer une variable" signifie effectuer une sommation de la loi jointe factorisée sur le domaine de la variable en question. La méthode consiste donc à éliminer une à une les variables non présentes dans \mathbf{X}_r , de telle sorte qu'en fin de procédure il ne reste plus que la LPC de \mathbf{X}_r . Le résultat de l'équation 2.15 ne dépend pas de l'ordre dans lequel sont effectuées les sommations, mais cet ordre peut avoir une influence sur la complexité des calculs à effectuer [DONAT 2009].

L'algorithme (1), inspiré de celui établi par [DONAT 2009], présente la méthode d'élimination de variables appliquée aux calculs d'inférence dans le cadre des réseaux bayésiens.

Illustrons cela par un exemple. Considérons le réseau bayésien de la figure 2.9, et calculons la distribution de probabilité de (X_3, X_5) à l'aide de l'algorithme.

La requête de l'inférence est $\mathbf{X}_r = (X_3, X_5)$.

Il s'agit d'éliminer une à une les variables qui n'appartiennent pas à la requête \mathbf{X}_r , c'est-à-dire les variables X_4 , X_2 et X_1 .

L'ordre choisi est $\sigma(X_1, \dots, X_5) = (X_3, X_5, X_4, X_2, X_1)$.

Initialement, $\mathcal{P} = \{p_1, \dots, p_5\}$

La première itération $d=1$ donne :

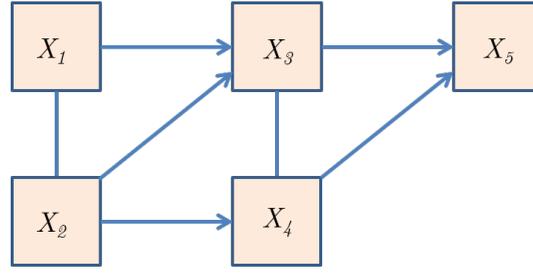


FIGURE 2.9 – Réseau bayésien

— $Y_1 = \sigma(X_1) = X_3$

— $Y_1 = X_3$ appartient à la requête $\mathbf{X}_r = (X_3, X_5)$

La seconde itération $d=2$ donne :

— $Y_2 = \sigma(X_2) = X_5$

— X_5 appartient à la requête $\mathbf{X}_r = (X_3, X_5)$

La troisième itération $d=3$ donne :

— $Y_3 = \sigma(X_3) = X_4$

— X_4 n'appartient pas à la requête $\mathbf{X}_r = (X_3, X_5)$

— $\mathcal{B}_{X_4} = \{m \in \mathcal{P} = \{p_1, \dots, p_5\} \mid \Omega_{X_4} \in \text{Dom}(m)\} = \{p_4, p_5\}$

— $\mathcal{P} = \{p_1, \dots, p_5\} \setminus \{p_4, p_5\} = \{p_1, p_2, p_3\}$

— $c_{X_4} = \sum_{\Omega_{X_4}} p_4 \times p_5$

— $\mathcal{P} = \{p_1, p_2, p_3, c_{X_4}\}$

La variable X_4 a été éliminée, cette étape a permis le calcul de c_{X_4} qui est en fait $P(X_5 \mid X_2, X_3)$ (cf. figure 2.10).

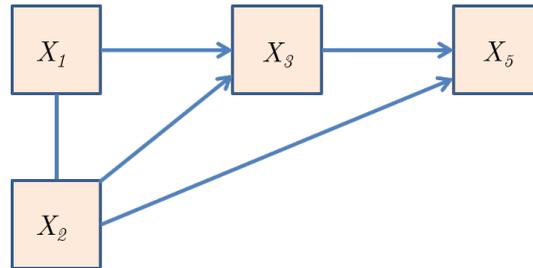


FIGURE 2.10 – Élimination de la variable X_4

La quatrième itération $d=4$ donne :

— $Y_4 = \sigma(X_4) = X_2$

— X_2 n'appartient pas à la requête $\mathbf{X}_r = (X_3, X_5)$

— $\mathcal{B}_{X_2} = \{m \in \mathcal{P} = \{p_1, p_2, p_3, c_{X_4}\} \mid \Omega_{X_2} \in \text{Dom}(m)\} = \{p_2, p_3, c_{X_4}\}$

— $\mathcal{P} = \{p_1, p_2, p_3, c_{X_4}\} \setminus \{p_2, p_3, c_{X_4}\} = \{p_1\}$

— $c_{X_2} = \sum_{\Omega_{X_2}} p_2 \times p_3 \times c_{X_4}$

— $\mathcal{P} = \{p_1, c_{X_2}\}$

La variable X_2 a été éliminée, cette étape a permis le calcul de c_{X_2} qui est en fait $P(X_5, X_3|X_1)$ (cf. figure 2.11).

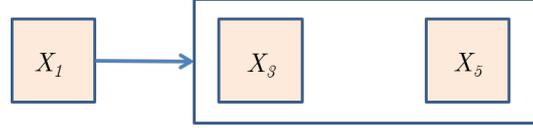


FIGURE 2.11 – Élimination de la variable X_2

La cinquième itération $d=5$ donne :

- $Y_5 = \sigma(X_5) = X_1$
- X_1 n'appartient pas à la requête $\mathbf{X}_r = (X_3, X_5)$
- $\mathcal{B}_{X_1} = \{m \in \mathcal{P} = \{p_1, c_{X_2}\} | \Omega_{X_1} \in \text{Dom}(m)\} = \{p_1, c_{X_2}\}$
- $\mathcal{P} = \{p_1, c_{X_2}\} \setminus \{p_1, c_{X_2}\} = \{\}$
- $c_{X_1} = \sum_{\Omega_{X_1}} p_1 \times c_{X_2}$
- $\mathcal{P} = \{c_{X_1}\}$

La variable X_1 a été éliminée, cette étape a permis le calcul de c_{X_1} qui est $P(X_5, X_3)$ (cf. figure 2.12).



FIGURE 2.12 – Élimination de la variable X_1

Notre exemple ne comportait pas d'observation (certaines ou incertaines), la normalisation est donc égale à 1.

$$P(X_3, X_5) = \prod_{m \in \mathcal{P}} m = c_{X_1}$$

L'algorithme a ainsi permis de calculer de proche en proche

$$P(X_3 = x_3, X_5 = x_5) = \sum_{x_1} p_1(x_1) \sum_{x_2} p_2(x_2|x_1) p_3(x_3|x_1, x_2) \times \sum_{x_4} p_4(x_4|x_2, x_3) p_5(x_5|x_3, x_4) \quad (2.16)$$

Algorithm 1: Élimination de variable

Entrées :

1. un ensemble de LPC $\{p_d\}_{1 \leq d \leq D}$ caractérisant la loi de $\mathbf{X} = (X_1, \dots, X_D)$ représenté par un RB
2. un vecteur aléatoire \mathbf{X}_r extrait de X représentant la requête de l'inférence.
3. un ordre d'élimination noté $\sigma(X_1, \dots, X_D)$
4. des observations certaines et des observations partielles

Sorties : la loi $P(\mathbf{X}_r | \text{observations})$
pour chaque $d = 1, \dots, D$ **faire**

```

// initialisation des potentiels : prise en compte des
// observations et des observations incertaines

```

```

si  $X_d$  est observée alors

```

```

|  $p_d \leftarrow \delta_{x_d}$ ;

```

```

sinon si  $X_d$  est partiellement observée alors

```

```

|  $p_d \leftarrow p_d \cdot p_{se_d}$ ;

```

```

fin

```

fin

```

 $\mathcal{P} \leftarrow \{p_1, \dots, p_D\}$ ;

```

pour chaque $d = 1, \dots, D$ **faire**

```

 $Y_d \leftarrow \sigma(X_d)$ ;

```

```

si  $Y_d$  n'appartient pas à la requête  $\mathbf{X}_r$  alors

```

```

|  $\mathcal{B}_{Y_d} \leftarrow \{m \in \mathcal{P} | \Omega_{Y_d} \in \text{Dom}(m)\}$  // LPC dont le domaine contient
|  $\Omega_{Y_d}$ 

```

```

|  $\mathcal{P} \leftarrow \mathcal{P} \setminus \mathcal{B}_{Y_d}$ ;

```

```

|  $c_{Y_d} \leftarrow \sum_{\Omega_{Y_d}} \prod_{m \in \mathcal{B}_{Y_d}} m$ ;

```

```

|  $\mathcal{P} \leftarrow \mathcal{P} \cup \{c_{Y_d}\}$ 

```

```

fin

```

fin

```

// normalisation et résultat

```

```

 $P(\mathbf{X}_r | \text{observations}) \leftarrow \frac{\prod_{m \in \mathcal{P}} m}{\sum_{\mathbf{x}_r} \prod_{m \in \mathcal{P}} m}$ ;

```

La complexité algorithmique de la méthode d'élimination de variables est étudiée en détail par DESCHTER dans [DECHTER 1999]. L'essentiel de la complexité est contenu dans l'étape de calcul de c_{X_d} . Cette complexité est directement liée au nombre de potentiels, et aux tailles des espaces d'état des variables.

2.3 Des limitations des MGD vers une nouvelle problématique

Intéressons-nous à la taille de l'espace d'état Ω_S de la variable S qui représente le temps de séjour dans l'état de fonctionnement courant.

Pour modéliser un système par un MGD, l'utilisateur doit déterminer l'espace d'état de la variable "temps de séjour". La valeur maximale de cet espace d'état, qui est la taille de F^{sys} , doit être telle que toutes les durées de séjour dans tous les états lui soient inférieures.

Si l'utilisateur dispose de données de retour d'expérience, la taille de F^{sys} peut par exemple être définie comme étant le maximum sur les différents états de l'ensemble des temps passés dans chacun des états.

Dans le cas où il ne dispose pas de base de retour d'expérience pour modéliser la table de probabilité F^{sys} empiriquement, il peut s'appuyer sur les avis d'experts. Il se trouve que, de par sa souplesse, la loi de Weibull arrive assez souvent à coller au comportement réel d'un système [CARROLL 2003]. Nous avons choisi de rester dans ce contexte-là pour ces travaux. Toutefois, nous montrerons dans la section 3.2 que les développements proposés admettent une certaine forme de généralité.

Après avoir estimé les paramètres des distributions de Weibull, il est nécessaire de les discrétiser pour créer la table F^{sys} . La discrétisation d'une loi de Weibull [BRACQUEMOND et GAUDOIN 2003], qui est à support infini, nécessite le choix d'une borne. Cette troncature entraînera nécessairement une perte d'information. Si elle est trop basse, elle ne représentera pas correctement la loi du temps de séjour. Elle doit donc être choisie suffisamment élevée pour garantir une bonne modélisation de la distribution du temps de séjour.

L'espace d'état de la variable S , $\Omega_S = 1, \dots, N_S$ peut donc nécessiter une grande taille, *i.e.* un N_S élevé, pour que la représentation de la distribution du temps de séjour soit précise.

La table F^{sys} (formule en 2.10) contient $N_X \cdot N_S$ éléments. Ainsi, le choix d'une valeur N_S trop importante aura des conséquences immédiates sur l'espace nécessaire pour stocker les distributions de probabilités conditionnelles.

On a d'autre part vu à la fin de la section 2.2.5 que la complexité algorithmique de l'inférence est aussi liée à la taille des espaces d'état des variables. Ainsi, une grande valeur N_S aura aussi des conséquences immédiates sur la complexité de l'inférence, et donc sur le temps de calcul.

Pour illustrer cela, en estimant par un calcul d'inférence la fiabilité de l'exemple développé dans la section 2.2.1, modélisé par un MGD (cf. figure 2.8), pour diffé-

rentes tailles de la table de probabilité du temps de séjour dans l'état de fonctionnement, on constate avec la figure 2.13 que le temps de calcul augmente de façon exponentielle, passant de 500 secondes = 7.5 minutes à 31700 secondes c'est à dire plus de 8 heures, et que le calcul ne peut plus s'effectuer passé une certaine taille de F^{sys} .

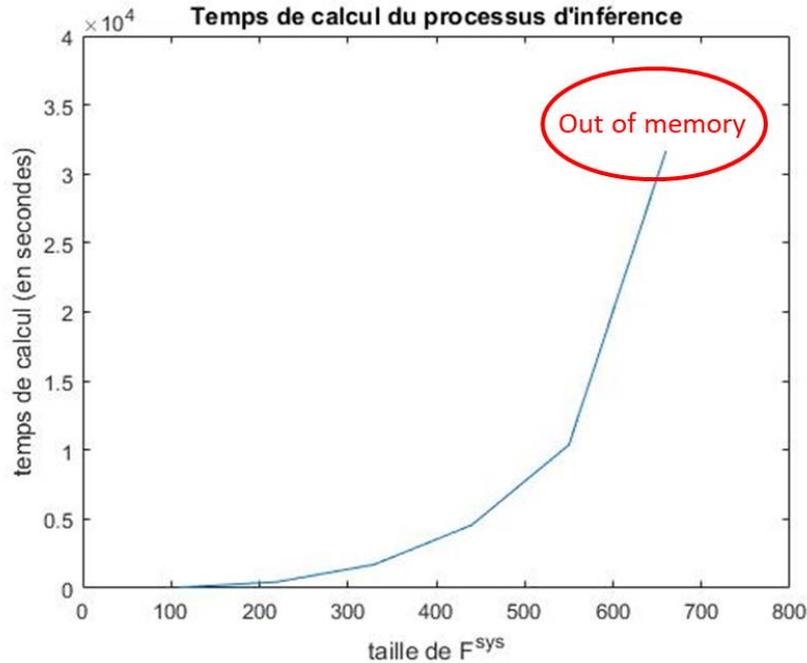


FIGURE 2.13 – Explosion du temps de calcul d'inférence dans le MGD

Pour conclure, un grand espace d'état de la variable durée de séjour induit donc des problèmes techniques en matière de capacité de stockage et de temps de calcul.

Le problème de complexité a été soulevé par [DONAT 2009].

De son côté, FOULLIARON a développé un algorithme d'inférence spécifique réduisant la complexité du calcul d'inférence dans le modèle de dégradation appelé modèle graphique de durée conditionnelle, mais la faisabilité de la méthode qu'il a mise en place est très dépendante de la borne supérieure des lois du temps de séjour dans chacun des états [FOULLIARON 2015].

En considérant une variable de durée continue, décrite par un nombre de paramètres très inférieur à N_S , l'espace nécessaire pour stocker la LPC continue sera inférieur à l'espace de stockage nécessité par la loi de probabilité d'une variable temps de séjour discrète, et les temps de calculs d'inférence seront également inférieurs comme nous le verrons dans la section 4.4.2.

MGDHW : Une approche hybride pour la modélisation des processus de dégradation

L'objectif de la thèse est de permettre à la variable durée de séjour du modèle graphique de durée d'être continue. Pour cela, nous proposons une extension du modèle graphique de durée : le modèle graphique de durée hybride-Weibull. La description du formalisme fait l'objet de la première partie de ce chapitre. La deuxième partie présente l'algorithme spécifique d'inférence mis en place.

3.1 Généralités sur le formalisme des MGDHW

Avant de présenter le modèle hybride mis en place, étudions ce qui a été fait auparavant dans la littérature concernant l'intégration de variables continues dans les réseaux bayésiens.

3.1.1 État de l'art sur les réseaux bayésiens admettant des variables continues

Dans la théorie des réseaux bayésiens, il existe plusieurs approches qui contiennent à la fois des variables continues et discrètes, telles que les réseaux bayésiens conditionnels hybrides gaussiens [LAURITZEN 1992], les mélanges d'exponentielles [KOLLER et al. 1999], ou encore les mélanges d'exponentielles tronquées [MORAL et al. 2001].

Les réseaux bayésiens conditionnels hybrides gaussiens mêlent des variables discrètes et des variables continues qui suivent des lois gaussiennes conditionnellement à leurs variables parentes. L'inférence exacte est possible dans de tels réseaux en utilisant l'algorithme de l'arbre de jonction [LAURITZEN 1992].

Toutefois, dans un réseau bayésien conditionnel hybride gaussien, les variables discrètes ne peuvent avoir de parents continus. Cette restriction peut être partiellement levée en utilisant la fonction logit ou probit, généralisée dans le cas multinomial

par la fonction softmax [MURPHY 1999].

Une autre façon de lever cette restriction est l'utilisation d'un mélange d'exponentielles [KOLLER et al. 1999], mais l'inférence n'est qu'approchée, et pas exacte.

Le modèle de mélange d'exponentielles tronquées a été introduit dans [MORAL et al. 2001]. Ce modèle permet l'inférence exacte via un algorithme de l'arbre de jonction, celui de Shafer-Shenoy, et autorise les variables discrètes à avoir des parents continus [COBB et SHENOY 2006].

Dans le modèle graphique de durée, la variable "durée de séjour" est déterministe lorsqu'il n'y a pas de changement d'état : elle est décrémentée d'une unité. Ainsi, sa distribution ne peut pas être modélisée par une gaussienne ou par un mélange d'exponentielles tronquées, et les travaux précédemment cités ne sont pas directement applicables aux MGD dans un objectif de rendre continue la variable "durée de séjour". Toutefois, l'existence de ces approches hybrides encourage à envisager une approche hybride des modèles graphiques de durée, qui respecte la forme particulière de la distribution de la variable temps de séjour dans le modèle graphique de durée.

Nous allons commencer par poser le formalisme du modèle graphique de durée dont les variables «durée de séjour» sont continues et suivent une loi de Weibull après chaque changement d'état.

3.1.2 Formalisme

Soit $(X_t)_{1 \leq t \leq T}$ une suite de variables aléatoires représentant l'état du système sur une période de longueur T . Soit $(S_t)_{1 \leq t \leq T}$ une suite de variables aléatoires représentant le temps restant avant une modification de l'état du système. Ces variables sont appelées variables de durée ou variables de temps de séjour. Un modèle graphique de durée hybride Weibull est illustré dans la figure 3.1.

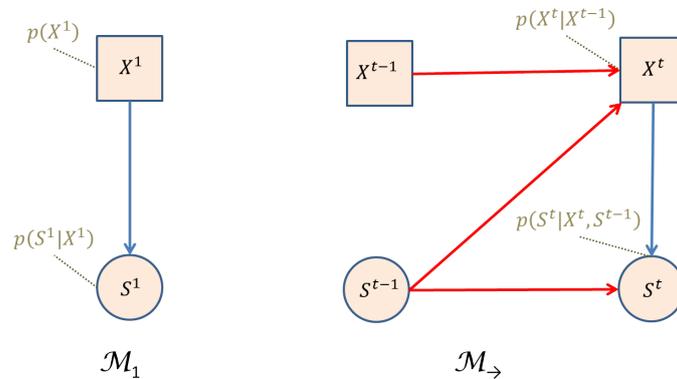


FIGURE 3.1 – Modèle graphique de durée Hybride-Weibull

Les expressions (2.7), (2.8), (2.9), (2.10), (2.11) et (2.12) introduites au chapitre

2 pour décrire un MGD deviennent les expressions (3.1), (3.2), (3.4), (3.5), (3.6) et (3.7) détaillées ci-après.

La distribution de probabilité conditionnelle associée à la distribution de l'état initial du système doit être définie comme suit, sur le domaine discret et fini $\Omega_X = 1, \dots, N_X$:

$$P(X_1 = i) = Q^{ini}(i) \quad (3.1)$$

La LPC initiale du temps de séjour donne la distribution pour chaque état initial, elle est définie sur le domaine continu $\Omega_S =]0, +\infty[$:

$$f_{S_1|X_1=i} = f_{\mathcal{W}_{\alpha_i, \beta_i}} \quad (3.2)$$

avec

$$f_{\mathcal{W}_{\alpha, \beta}}(s) = \frac{\beta}{\alpha} \left(\frac{s}{\alpha}\right)^{\beta-1} e^{-\left(\frac{s}{\alpha}\right)^\beta} \quad (3.3)$$

où α est le paramètre d'échelle et β le paramètre de forme. Les valeurs de α et β dépendent de l'état courant.

Dans un premier temps, nous avons choisi de nous intéresser à la loi de Weibull [WEIBULL 1951], cette loi apparaissant naturellement dans l'étude de la distribution limite du minimum d'un échantillon iid, par exemple dans l'étude de la distribution des différents temps de séjour d'un système constitué de composants dont l'état est déterminé par le pire état de ces derniers. En effet, avec X_1, \dots, X_n une suite de variables iid de fonction de densité f sur $]0, +\infty[$ telle que $\lim_{x \rightarrow 0^+} \frac{\alpha^\beta f(x)}{\beta x^{\beta-1}} = 1$, alors $n^{\frac{1}{\beta}} \min(X_1, \dots, X_n)$ converge en loi lorsque n tend vers l'infini vers la loi de Weibull $\mathcal{W}(\alpha, \beta)$ [Ycart 2004].

Dans le cas d'une variable temps de séjour S_t continue, une transition (un changement d'état) a lieu si et seulement si $S_{t-1} \leq 1$.

La probabilité conditionnelle de transition de l'état du système devient dans ce cas :

$$P(X_t = j | X_{t-1} = i, S_{t-1} \leq 1) = Q^{sys}(i, j) \quad (3.4)$$

où Q^{sys} est une matrice $N_X \cdot N_X$, appelée matrice de transition naturelle du système.

Étant donné qu'il s'agit d'une table de changement d'état, et un système ne restant par définition pas dans le même état après changement d'état, les $Q^{sys}(i, i)$ sont nuls pour tout i différent de l'état de panne.

Le système ne pouvant se réparer lui-même, les probabilités qu'il passe dans un meilleur état de fonctionnement après une transition naturelle sont nulles.

TABLE 3.1 – Table de probabilité conditionnelle de transition naturelle d'un système à trois états de fonctionnement sans réparation

	état ok	état dégradé	état panne
état ok	0	q	1-q
état dégradé	0	0	1
état panne	0	0	1

Dans un système à trois états de fonctionnement "ok", "dégradé", "panne", il n'y a donc qu'à définir $Q^{sys}(ok, deg)$ ou $Q^{sys}(ok, panne)$, qui sont tels que $Q^{sys}(ok, deg) + Q^{sys}(ok, panne) = 1$.

La table 3.1 illustre la table de transition naturelle d'un système à trois états de fonctionnement.

Suite à une transition, un nouveau temps de séjour est calculé selon la LPC suivante, qui est la LPC initiale des temps de séjour :

$$f_{S_t=s|S_{t-1}\leq 1, X_t=i} = f_{W_{\alpha_i, \beta_i}}(s) \quad (3.5)$$

Des paramètres de Weibull sont nécessaires pour chacun des états de fonctionnement. L'état de panne ne requiert pas de distribution de Weibull associée, puisqu'il est absorbant.

Par contre, lorsqu'il n'y a pas de transition, il n'y a pas de changement d'état, le système reste de façon déterministe dans l'état précédent i , la matrice de transition est alors la matrice identité :

$$P(X_t = j | X_{t-1} = i, S_{t-1} > 1) = I(i, j) \quad (3.6)$$

Le temps de séjour dans l'état actuel est quant à lui décrétement de manière déterministe d'une unité :

$$f(S_t = s | X_t = i, S_{t-1} = s' > 1) = \begin{cases} 1 & \text{si } s = s' - 1 \\ 0 & \text{sinon} \end{cases} = \delta_{s=s'-1} \quad (3.7)$$

Le formalisme du modèle graphique de durée hybride Weibull étant posé, intéressons-nous au processus d'apprentissage d'un tel réseau, avant de présenter le processus d'inférence à mettre en place.

3.1.3 Apprentissages

L'apprentissage de la structure d'un MGDHW ne diffère pas de l'apprentissage de la structure d'un réseau bayésien classique. Une fois la structure particulière du

modèle graphique de durée hybride Weibull apprise, il faut passer à l'apprentissage des paramètres. Pour ces apprentissages, les méthodes présentées en section 2.2.4 sont directement applicables. L'apprentissage des paramètres des lois de Weibull modélisant les durées de séjour dans chacun des états nécessite une étape supplémentaire, consistant en une régression linéaire.

L'apprentissage des paramètres probabilistes d'un modèle graphique de durée hybride Weibull dans le cas de données complètes sera développé avec la présentation de l'exemple dans la section 4.3.1, pour lequel la structure sera supposée connue.

3.2 Inférence

3.2.1 Algorithme itératif

Présentons maintenant l'algorithme spécifique mis en place pour effectuer les calculs d'inférence dans un MGDHW.

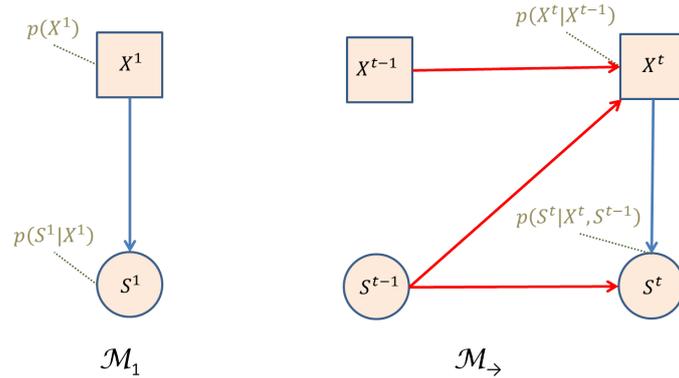


FIGURE 3.2 – Modèle graphique de durée Hybride-Weibull

Développons le calcul de la distribution de probabilité de la variable X_t sachant des observations incertaines sur toutes les variables d'état de fonctionnement précédentes. Cela correspond au calcul de la probabilité d'un état à un instant donné en ayant un historique d'observations des états précédents.

Nous cherchons à évaluer $P(X_t|se)$. La première étape consiste à calculer la probabilité $P(X_t, se)$, puis la deuxième réside en une normalisation de $P(X_t, se)$ sur X_t pour obtenir $P(X_t|se)$. Pour rappel, $p_{se_t}(x_t)$ désigne la probabilité que X soit dans l'état x_t sachant qu'il est observé dans l'état se_t .

L'algorithme s'inspire de l'algorithme d'élimination de variables, présenté en section 2.2.5.

Le principe repose sur le fait que

$$\begin{aligned}
 P(X_t, se) &= \sum_{x_1 \in \Omega_{X_1}} \int_0^\infty \dots \sum_{x_{t-1} \in \Omega_{X_{t-1}}} \left[\int_0^\infty P(X_t | X_{t-1} = x_{t-1}, S_{t-1} = s_{t-1}) p_{se_t}(x_t) \times \right. \\
 &\quad \left. f(S_{t-1} = s_{t-1} | X_{t-1} = x_{t-1}, S_{t-2} = s_{t-2}) ds_{t-1} \right] \times \\
 &\quad P(X_{t-1} = x_{t-1} | X_{t-2} = x_{t-2}, S_{t-2} = s_{t-2}) p_{se_{t-1}}(x_{t-1}) \times \\
 &\quad \dots \times f(S_1 = s_1 | X_1 = x_1) ds_1 \times P(X_1 = x_1) p_{se_1}(x_1) \quad (3.8)
 \end{aligned}$$

L'idée est d'éliminer successivement les variables $S_{t-1}, X_{t-1}, S_{t-2}, \dots, X_2, S_1$, et X_1 , par des intégrations et des sommations successives. Éliminer S_u , consiste à passer de $P(X_t, obs_{u+1}, \dots, obs_t | X_u, S_u = s_u)$ à $P(X_t, obs_{u+1}, \dots, obs_t | X_u, S_{u-1} = s_{u-1})$ par une intégration. Éliminer X_u , consiste à passer de $P(X_t, obs_{u+1}, \dots, obs_t | X_u = x, S_{u-1})$ à $P(X_t, obs_u, \dots, obs_t | X_{u-1} = x, S_{u-1})$ par une sommation.

Pour simplifier les notations, posons

$$\begin{cases} \Phi_{t,u} = P(X_t, obs_{u+1}, \dots, obs_t | X_u, S_{u-1}) \text{ pour } u \in \llbracket 2; t-1 \rrbracket \\ \Phi_{t,1} = P(X_t, obs_2, \dots, obs_t | X_1) \\ \Lambda_{t,u} = P(X_t, obs_u, \dots, obs_t | X_{u-1}, S_{u-1}) \text{ pour } u \in \llbracket 2; t \rrbracket \\ \Lambda_{t,1} = P(X_t, obs_1, \dots, obs_t) \end{cases} \quad (3.9)$$

$$\Lambda_{t,t} = P(X_t, obs_t | X_{t-1}, S_{t-1}) \quad (3.10)$$

est donné par les formules 3.4 et 3.6.

Pour $u \in \llbracket 2; t-1 \rrbracket$, $\Phi_{t,u}$ se calcule à partir de $\Lambda_{t,u+1}$:

$$\Phi_{t,u} = \int_0^\infty \Lambda_{t,u+1} \times f(S_u = s_u | X_u, S_{u-1}) ds_u, \quad (3.11)$$

$f(S_u = s_u | X_u, S_{u-1})$ étant donné par les formules 3.5 et 3.7. Cette étape élimine la variable S_u .

Pour $u \in \llbracket 2; t-1 \rrbracket$, $\Lambda_{t,u}$ se calcule à partir de $\Phi_{t,u}$:

$$\Lambda_{t,u} = \sum_{x \in \Omega_{X_u}} \Phi_{t,u} \times P(X_u = x | X_{u-1}, S_{u-1}) \times P(obs_u | X_u = x), \quad (3.12)$$

$P(X_u = x | X_{u-1}, S_{u-1})$ étant donné par les formules 3.4 et 3.6. Cette étape élimine la variable X_u .

Une fois $\Lambda_{t,2}$ calculé, $\Phi_{t,1}$ se calcule ainsi :

$$\Phi_{t,1} = \int_0^{\infty} \Lambda_{t,2} f(S_1 = s_1 | X_1) ds_1 \quad (3.13)$$

Cette étape élimine la variable S_1 .

Puis $\Lambda_{t,1}$ se calcule à partir de $\Phi_{t,1}$:

$$\Lambda_{t,1} = \sum_{x \in \Omega_{X_1}} \Phi_{t,1} \times P(X_1 = x) P(obs_1 | X_1 = x_1) \quad (3.14)$$

Cette étape élimine la variable X_1 .

Les équations 3.10 3.11 3.12 3.13 3.14 définissent un algorithme itératif, qui permet à terme de calculer $\Lambda_{t,1} = P(X_t, obs)$.

Nous allons maintenant présenter en détails les étapes successives de ces calculs.

3.2.2 Méthode de calcul

Notations

Dans la suite, $A.B$ désignera le produit matriciel de Hadamard : chacun de ses coefficient est le produit "terme à terme" des coefficients des matrices A et B .

Par défaut, un vecteur v désigne un vecteur colonne, et v^τ sa transposée. v^τ désigne donc un vecteur ligne.

$A.v$ désignera la matrice constituée des colonnes de A multipliées par l'élément de v correspondant à l'indice de la colonne.

$v^\tau.A$ désignera la matrice constituée des lignes de A multipliées par l'élément de v correspondant à l'indice de la ligne.

Avec v et w deux vecteurs, $v.w$ désigne le produit scalaire des deux vecteurs, c'est à dire la somme du produit terme à terme de leurs éléments.

Initialisation

Regardons ce qui constitue $\Lambda_{t,t}$, la première étape du calcul.

$\Lambda_{t,t} = P(obs_t | X_t) \times P(X_t = x | X_{t-1}, S_{t-1})$ est connu de la description du réseau, via les formules 3.4 et 3.6. $\Lambda_{t,t}$ peut s'exprimer en fonction de la valeur de s_{t-1} comme suit :

$$\Lambda_{t,t} = \begin{cases} I.p_{set} & \text{si } 1 < s_{t-1} \\ Q^{sys}.p_{set} & \text{si } s_{t-1} \leq 1 \end{cases} \quad (3.15)$$

Posons $Q_{\Lambda_{t,t}} = [I.p_{set} \quad Q^{sys}.p_{set}]$. $Q_{\Lambda_{t,t}}(1) = I.p_{set}$ décrit $\Lambda_{t,t}$ pour les $s > 1$, $Q_{\Lambda_{t,t}}(2) = Q^{sys}.p_{set}$ décrit $\Lambda_{t,t}$ pour les $s \leq 1$

Intégration : calcul de Φ

Passons au calcul présenté dans la formule 3.11. $\Lambda_{t,u+1}$ est décrit par $Q_{\Lambda_{t,u+1}} = [Q_{\Lambda_{t,u+1}}(1) \quad \dots \quad Q_{\Lambda_{t,u+1}}(t-u+1)]$ de la manière suivante :

$$\Lambda_{t,u+1} = \begin{cases} Q_{\Lambda_{t,u+1}}(1) = \prod_{\eta=u+1}^{t-1} p_{se\eta}^\tau . I . p_{set} & \text{si } t-u < s_u \\ Q_{\Lambda_{t,u+1}}(2) = \prod_{\eta=u+1}^{t-1} p_{se\eta}^\tau . Q^{sys} . p_{set} & \text{si } t-u-1 < s_u \leq t-u \\ Q_{\Lambda_{t,u+1}}(3) & \text{si } t-u-2 < s_u \leq t-u-1 \\ \vdots & \\ \vdots & \\ Q_{\Lambda_{t,u+1}}(t-u+1) & \text{si } s_u \leq 1 \end{cases} \quad (3.16)$$

L'objectif est de calculer $\Phi_{t,u} = \int_0^\infty \Lambda_{t,u+1} \times f(s_u | X_u, S_{u-1}) ds_u$. L'expression de $f(S_u = s_u | X_u, S_{u-1})$ est donnée par 3.5 et 3.7. Pour les $s_{u-1} > 1$, il n'y a pas de transition, la loi du temps de séjour s'écrit $f(s_u | X_u, S_{u-1} = s_{u-1}) = \delta_{s_u = s_{u-1} - 1}$, donc $\Phi_{t,u}(s_{u-1}) = \Lambda_{t,u+1}(s_{u-1})$ pour tout $1 < s_{u-1}$.

Il reste à calculer $\Phi_{t,u}$ pour $s_{u-1} \leq 1$, ce qui correspond à un changement d'état. Dans ce cas, $f_{S_u=s | S_{u-1} \leq 1, X_u = x_u} = f_{\mathcal{W}_{\alpha x_u, \beta x_u}}(s)$.

Il vient alors :

$$\begin{aligned} & Q_{\Phi_{t,u}}(t-u+2) \\ = & \Phi_{t,u}(s_{u-1} \leq 1) \\ = & \int_0^\infty f_{\mathcal{W}_{\alpha x_u, \beta x_u}} \times \Lambda_{t,u+1} ds_u \\ = & \sum_{k=1}^{t-u} \int_{k-1}^k f_{\mathcal{W}_{\alpha x_u, \beta x_u}}(s_u) ds_u \times Q_{\Lambda_{t,u+1}}(t-u-k+2) + \\ & \int_{t-u}^\infty f_{\mathcal{W}_{\alpha x_u, \beta x_u}}(s_u) ds_u \times Q_{\Lambda_{t,u+1}}(1) \\ = & \sum_{k=1}^{t-u} (F_{\mathcal{W}_{\alpha x_u, \beta x_u}}(k) - F_{\mathcal{W}_{\alpha x_u, \beta x_u}}(k-1)) ds_u \times Q_{\Lambda_{t,u+1}}(t-u-k+2) + \\ & (1 - F_{\mathcal{W}_{\alpha x_u, \beta x_u}}(t-u)) ds_u \times Q_{\Lambda_{t,u+1}}(1) \end{aligned} \quad (3.17)$$

$Q_{\Phi_{t,u}} = [Q_{\Phi_{t,u}}(1), \dots, Q_{\Phi_{t,u}}(t-u+2)]$ est alors construit comme suit :

$$\Phi_{t,u} = \left\{ \begin{array}{ll} Q_{\Phi_{t,u}}(1) & = \prod_{\eta=u+1}^{t-1} p_{se_{\eta}}^{\tau} \cdot I \cdot p_{se_t} & \text{si } t-u+1 < s_{u-1} \\ Q_{\Phi_{t,u}}(2) & = \prod_{\eta=u+1}^{t-1} p_{se_{\eta}}^{\tau} \cdot Q^{sys} \cdot p_{se_t} & \text{si } t-u < s_{u-1} \\ & & \leq t-u+1 \\ Q_{\Phi_{t,u}}(3) & = Q_{\Lambda_{t,u+1}}(3) & \text{si } t-u-1 < s_{u-1} \\ & & \leq t-u \\ \vdots & & \\ Q_{\Phi_{t,u}}(t-u+1) & = Q_{\Lambda_{t,u+1}}(t-u+1) & \text{si } 1 < s_{u-1} \leq 2 \\ Q_{\Phi_{t,u}}(t-u+2) & = \sum_{k=1}^{t-u} \int_{k-1}^k f_{\mathcal{W}_{\alpha_{x_u}, \beta_{x_u}}}(s_u) ds_u & \text{si } s_{u-1} \leq 1 \\ & \times Q_{\Lambda_{t,u+1}}(t-u-k+2) \\ + Q_{\Lambda_{t,u+1}}(1) & \times \int_{t-u}^{\infty} f_{\mathcal{W}_{\alpha_{x_u}, \beta_{x_u}}}(s_u) ds_u \end{array} \right. \quad (3.18)$$

Remarquons que la forme de la loi considérée pour caractériser la variable "temps de séjour" après changement d'état intervient à cette étape des calculs. Il est *a priori* possible de considérer n'importe quelle loi, tant que le calcul analytique de sa fonction de répartition reste possible. Dans le cas d'une densité de Weibull, nous obtenons

$$\begin{aligned} \int_{k-1}^k f_{\mathcal{W}_{\alpha_{x_u}, \beta_{x_u}}}(s_u) ds_u &= F_{\mathcal{W}_{\alpha_{x_u}, \beta_{x_u}}}(k) - F_{\mathcal{W}_{\alpha_{x_u}, \beta_{x_u}}}(k-1) \\ &= e^{-\left(\frac{k-1}{\alpha_{x_u}}\right)^{\beta_{x_u}}} - e^{-\left(\frac{k}{\alpha_{x_u}}\right)^{\beta_{x_u}}} \end{aligned}$$

Sommation : calcul de Λ

$\Phi_{t,u}$ est décrit par $Q_{\Phi_{t,u}}$ dont les éléments sont présentés en 3.18. Détaillons maintenant le calcul présenté dans la formule 3.12, qui permet de passer de $\Phi_{t,u}$ à $\Lambda_{t,u}$.

$$\Lambda_{t,u} = \sum_{x \in \Omega_{X_u}} P(X_u = x | X_{u-1}, S_{u-1}) \times P(obs_u | X_u = x) \times \Phi_{t,u}$$

L'expression de $P(X_u = x | X_{u-1}, S_{u-1})$ est donnée par par 3.4 et 3.6 : $P(X_u = x | X_{u-1}, S_{u-1} > 1) = I$ et $P(X_u = x | X_{u-1}, S_{u-1} \leq 1) = Q^{sys}$.

Il vient alors

$$\Lambda_{t,u} = \begin{cases} p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(1) & \text{si } t - u + 1 < s_{u-1} \\ p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(2) & \text{si } t - u < s_{u-1} \leq t - u + 1 \\ p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(3) & \text{si } t - u - 1 < s_{u-1} \leq t - u \\ \vdots & \\ p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(t - u + 1) & \text{si } 1 < s_{u-1} \leq 2 \\ Q^{sys} * (p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(t - u + 2)) & \text{si } s_{u-1} \leq 1 \end{cases} \quad (3.19)$$

c'est à dire

$$\Lambda_{t,u} = \begin{cases} \prod_{\eta=u}^{t-1} p_{se_\eta}^\tau \cdot I \cdot p_{se_t} & \text{si } t - u + 1 < s_{u-1} \\ \prod_{\eta=u}^{t-1} p_{se_\eta}^\tau \cdot Q^{sys} \cdot p_{se_t} & \text{si } t - u < s_{u-1} \leq t - u + 1 \\ p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(3) & \text{si } t - u - 1 < s_{u-1} \leq t - u \\ \vdots & \\ p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(t - u + 1) & \text{si } 1 < s_{u-1} \leq 2 \\ Q^{sys} * (p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(t - u + 2)) & \text{si } s_{u-1} \leq 1 \end{cases} \quad (3.20)$$

$Q^{sys} * (p_{se_u}^\tau \cdot Q_{\Phi_{t,u}}(t - u + 2))$ représente la matrice de transition naturelle du système pendant $t - u + 1$ tranches de temps.

Dernière intégration

Ces étapes sont répétées jusqu'à l'obtention de $\Lambda_{t,2}$, qui est décrit par $Q_{\Lambda_{t,2}} = [Q_{\Lambda_{t,2}}(1) \quad \dots \quad Q_{\Lambda_{t,2}}(t)]$ de la manière suivante :

$$\Lambda_{t,2} = \begin{cases} Q_{\Lambda_{t,2}}(1) = \prod_{\eta=u+1}^{t-1} p_{se_\eta}^\tau \cdot I \cdot p_{se_t} & \text{si } t - 1 < s_1 \\ Q_{\Lambda_{t,2}}(2) = \prod_{\eta=u+1}^{t-1} p_{se_\eta}^\tau \cdot Q^{sys} \cdot p_{se_t} & \text{si } t - 2 < s_1 \leq t - 1 \\ Q_{\Lambda_{t,2}}(3) & \text{si } t - 3 < s_1 \leq t - 2 \\ \vdots & \\ Q_{\Lambda_{t,2}}(t) & \text{si } s_1 \leq 1 \end{cases} \quad (3.21)$$

L'objectif est cette fois de calculer $\Phi_{t,1} = \int_0^\infty \Lambda_{t,2} \times f(S_1 = s_1 | X_1) ds_1$ (formule 3.13).

L'expression de $f(S_1 = s_1 | X_1)$ est donnée par la formule 3.2 : $f_{S_1=s_1|X_1=x_1} = f_{W_{\alpha_{x_1}, \beta_{x_1}}}(s)$

Il vient alors

$$\begin{aligned}
\Phi_{t,1} &= \int_0^{\infty} \Lambda_{t,2} \times f(S_1 = s_1 | X_1) ds_1 \\
&= \int_0^{\infty} \Lambda_{t,2}(s_1) \times f_{\mathcal{W}_{\alpha_{x_1}, \beta_{x_1}}}(s) ds_1 \\
&= \sum_{k=1}^{t-1} \int_{k-1}^k \Lambda_{t,2}(s_1) \times f_{\mathcal{W}_{\alpha_{x_1}, \beta_{x_1}}}(s) ds_1 + \int_{t-1}^{\infty} \Lambda_{t,2}(s_1) \times f_{\mathcal{W}_{\alpha_{x_1}, \beta_{x_1}}}(s) ds_1 \\
&= \sum_{k=1}^{t-1} \int_{k-1}^k Q_{\Lambda_{t,2}}(t-k+1) \times f_{\mathcal{W}_{\alpha_{x_1}, \beta_{x_1}}}(s) ds_1 + \int_{t-1}^{\infty} Q_{\Lambda_{t,2}}(1) \times f_{\mathcal{W}_{\alpha_{x_1}, \beta_{x_1}}}(s) ds_1
\end{aligned}$$

Dernière sommation

$\Phi_{t,1}$ est désormais calculé. Il ne reste plus qu'à obtenir $\Lambda_{t,1} = \sum_{x \in \Omega_{X_1}} \Phi_{t,1} \times P(X_1 = x)P(obs_1 | X_1 = x_1)$ (formule 3.14).

L'expression de $P(X_1 = x)$ est donnée par la formule 3.1 : $P(X_1) = Q^{ini}$.

Il vient alors

$$\begin{aligned}
\Lambda_{t,1} &= \sum_{x \in \Omega_{X_1}} \Phi_{t,1} \times P(X_1 = x)P(obs_1 | X_1 = x_1) \\
&= \sum_{x \in \Omega_{X_1}} \Phi_{t,1} \times Q^{ini} \times p_{se1}
\end{aligned}$$

Il suffit ensuite de normaliser $\Lambda_{t,1}$ pour obtenir $P(X_t | obs)$:

$$P(X_t | obs) = \frac{\Lambda_{t,1}}{\sum_{\Omega_{X_t}} \Lambda_{t,1}}$$

La manière dont sont utilisées les distributions de Weibull dans les différentes étapes de l'inférence ne sont pas spécifiques à la nature de la loi. La nature de la distribution du temps de séjour après changement d'état a peu d'importance, il serait aisé de remplacer la loi de Weibull par une autre loi, du moment que la fonction de répartition reste calculable. Une condition est toutefois de considérer une loi dont les paramètres s'apprennent facilement, pour ne pas perdre en vitesse d'apprentissage, et qui soit caractérisée par un petit nombre de paramètres pour n'utiliser qu'un faible espace de stockage mémoire.

Après avoir été présenté, l'algorithme d'inférence mis en place au sein du MGDHW va maintenant être validé sur un jeu de données. C'est l'objet de la partie suivante.

Résultats expérimentaux

4.1 Présentation du protocole expérimental

4.1.1 Introduction

L'objectif de cette partie est de présenter le protocole expérimental mis en place pour valider la nouvelle approche. L'enjeu est d'examiner l'apport du MGDHW par rapport aux MGD.

Reprenons l'exemple introduit en 2.2.1 considérant une machine de production à trois états : « ok » et « dégradé » pour les états de fonctionnement et « en panne » pour l'état de défaillance.

Il s'agit de modéliser la dynamique de dégradation de la machine de production par, d'une part, un MGDHW et, d'autre part, un MGD.

Pour effectuer ces modélisations, il faut disposer d'un historique d'apprentissage, constitué de "trajectoires" représentant différentes évolutions de l'état du système vers l'état de panne.

Une trajectoire naturelle correspond à l'observation d'une séquence d'états du système au cours du temps en le laissant vieillir sans jamais intervenir, jusqu'à la panne.

La figure 4.1 présente un exemple de trajectoire.

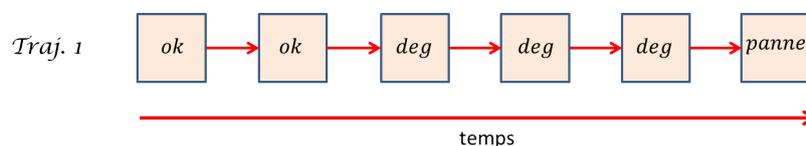


FIGURE 4.1 – Exemple de trajectoire

Ne disposant pas de base de retour d'expérience (REX), nous avons simulé une base de trajectoires, en considérant un système dont les temps de séjour dans chaque nouvel état de fonctionnement suivent une loi de Weibull. Ce système est modélisé par le MGDHW de la figure 4.2, avec X l'état de fonctionnement, et S la variable "temps de séjour".

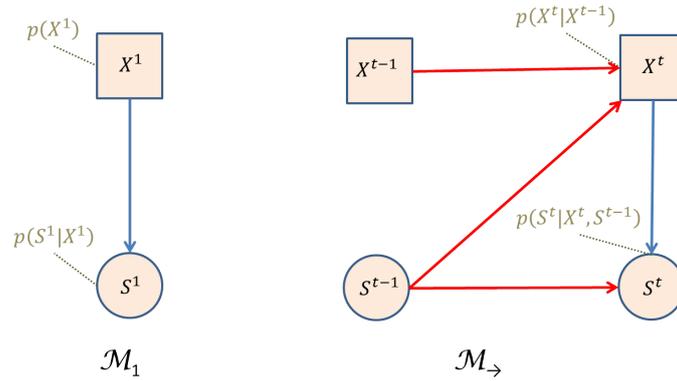


FIGURE 4.2 – Modèle graphique de durée Hybride-Weibull

Il s'agira notamment de comparer l'inférence réalisée avec un MGDHW et celle réalisée avec un MGD afin de mettre en évidence les limites et avantages de l'approche proposée dans cette thèse. La fiabilité théorique d'un tel système étant calculable (4.4.3), c'est le calcul d'inférence donnant l'estimation de la fiabilité qui sera réalisé d'une part dans un MGDHW et d'autre part dans un MGD, afin d'évaluer l'algorithme d'inférence proposé en 3.2.

Si l'état de la machine de production est connu via un capteur, l'information renvoyée par le-dit capteur peut être prise en compte dans le modèle en associant à l'état de la machine une observation incertaine dans le calcul d'inférence (cf. section 2.2.5). Dans la situation présente, il s'agit uniquement de valider l'algorithme d'inférence, il n'est donc pas utile de supposer une quelconque connaissance de l'état du système.

Un des objectifs est notamment d'étudier si le MGDHW permet de s'affranchir du problème posé par la taille de F^{sys} dans le MGD.

4.1.2 Choix d'un paramètre initial impactant la taille de F

Comme la complexité d'un MGD croit fortement en fonction de la taille de l'espace d'état des temps de séjour, nous allons mener des calculs jusqu'à la limite mémoire de notre machine et les comparer avec notre approche MGDHW.

Pour comparer efficacement l'utilisation d'un MGDHW et celle d'un MGD en ce qui concerne l'impact de la taille de l'espace d'état discret de la variable S , tout en partant d'un MGDHW avec une variable S continue, il est utile de déterminer un lien entre les paramètres du MGDHW et la taille de \hat{F}^{sys} . Les modèles seront comparés entre eux par rapport à ces valeurs reliées.

Le moyen choisi pour que la taille de \hat{F}^{sys} , appris des données simulées, varie de façon certaine, prévisible et contrôlée à partir du modèle MGDHW initial, est de

désigner un unique paramètre du modèle dont sont issues les simulations, et de faire varier ce paramètre.

Le paramètre sélectionné est le paramètre d'échelle de la loi de Weibull associée à l'état "ok", α_{ok} . La taille de \hat{F}^{sys} sera alors proche de

$$\alpha_{ok}\beta_{ok}K \int_0^\infty (1 - e^{-u^{\beta_{ok}}})^{K-1} e^{-u^{\beta_{ok}}} u^{\beta_{ok}} du$$

pour des paramètres tels que $\beta_{ok} = \beta_{deg}$ et $\alpha_{ok} \geq \alpha_{deg}$. La suite de ce paragraphe explique la raison de ces choix.

Considérons la méthode d'apprentissage de cette matrice. \hat{F}^{sys} est apprise à partir de K trajectoires considérées parmi toutes celles disponibles dans la base d'apprentissage. Les occurrences de chacun des temps de séjour sont comptabilisées. La taille de \hat{F}^{sys} est donc la plus grande valeur des temps de séjour atteinte, par l'une des K trajectoires, quel que soit l'état courant. Cette taille dépend des trajectoires choisies. C'est une variable aléatoire, l'idée est donc d'en calculer l'espérance.

Soient des paramètres $\alpha_{ok}, \beta_{ok}, \alpha_{deg}, \beta_{deg}$ tels que

$$\begin{aligned} \beta_{ok} &= \beta_{deg} \\ \alpha_{ok} &\geq \alpha_{deg} \end{aligned} \tag{4.1}$$

Soient S_1, \dots, S_K les K temps de séjour des K trajectoires dans l'état "ok". Les S_k sont iid de loi $\mathcal{W}(\alpha_{ok}, \beta_{ok})$.

Soit $S_{ok}^* = \max(S_1, \dots, S_K)$

$$\begin{aligned} F_{S_{ok}^*}(s) &= P(S^* \leq s) \\ &= P(\max(S_1, \dots, S_K) \leq s) \\ &= P(\forall k S_k \leq s) \\ &= P(S_1 \leq s, \dots, S_K \leq s) \\ &= P(S_1 \leq s)^K \\ &= F_{\mathcal{W}(\alpha_{ok}, \beta_{ok})}(s)^K \\ &= (1 - e^{-(\frac{s}{\alpha_{ok}})^{\beta_{ok}}})^K \end{aligned} \tag{4.2}$$

$$\begin{aligned}
 \mathbb{E}(S_{ok}^*) &= \int_0^\infty s f_{S_{ok}^*}(s) ds \\
 &= \int_0^\infty s F'_{S_{ok}^*}(s) ds \\
 &= \int_0^\infty s \frac{d}{ds} \left((1 - e^{-\left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}}})^K \right) ds \\
 &= \int_0^\infty s K (1 - e^{-\left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}}})^{K-1} e^{-\left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}}} \frac{\beta_{ok}}{\alpha_{ok}} \left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}-1} ds \\
 &= K \beta_{ok} \int_0^\infty \frac{s}{\alpha_{ok}} (1 - e^{-\left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}}})^{K-1} e^{-\left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}}} \left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}-1} ds
 \end{aligned}$$

En posant $u = \frac{s}{\alpha_{ok}}$, il vient

$$\mathbb{E}(S_{ok}^*) = \alpha_{ok} K \beta_{ok} \int_0^\infty u (1 - e^{-(u)^{\beta_{ok}}})^{K-1} e^{-(u)^{\beta_{ok}}} (u)^{\beta_{ok}-1} du \quad (4.3)$$

La taille de \hat{F}^{sys} sera égale au maximum des temps de séjour dans les différents états "ok" et "dégradé".

Calculons la probabilité qu'un des temps de séjour dans l'état "dégradé" soit supérieur au maximum des temps de séjour dans l'état "ok", appelé S_{ok}^* .

$$\begin{aligned}
 P(S_{deg} > S_{ok}^*) &= P(S_{deg} > \max(S_1, \dots, S_K)) \\
 &= P(S_{deg} > S_1, \dots, S_{deg} > S_K) \\
 &= P(S_{deg} > S_{ok})^K
 \end{aligned} \quad (4.4)$$

$$\begin{aligned}
 P(S_{deg} > S_{ok}) &= \int_0^\infty P(S_{deg} > s) f_{S_{ok}}(s) ds \\
 &= \int_0^\infty e^{-\left(\frac{s}{\alpha_{deg}}\right)^{\beta_{deg}}} \beta_{ok} \frac{s^{\beta_{ok}-1}}{\alpha_{ok}^{\beta_{ok}}} e^{-\left(\frac{s}{\alpha_{ok}}\right)^{\beta_{ok}}} ds \\
 &= \int_0^\infty e^{-\left(\frac{s^{\beta_{deg}}}{\alpha_{deg}^{\beta_{deg}}}\right) + \left(\frac{s^{\beta_{ok}}}{\alpha_{ok}^{\beta_{ok}}}\right)} \beta_{ok} \frac{s^{\beta_{ok}-1}}{\alpha_{ok}^{\beta_{ok}}} ds \\
 &= \int_0^\infty e^{-s^{\beta} \left(\frac{1}{\alpha_{deg}^{\beta}} + \frac{1}{\alpha_{ok}^{\beta}}\right)} \beta \frac{s^{\beta-1}}{\alpha_{ok}^{\beta}} ds
 \end{aligned} \quad (4.5)$$

Posons

$$\frac{1}{\alpha^{\beta}} = \frac{1}{\alpha_{deg}^{\beta}} + \frac{1}{\alpha_{ok}^{\beta}}$$

$$\begin{aligned}
P(S_{deg} > S_{ok}) &= \frac{1}{\alpha_{ok}^\beta} \int_0^\infty e^{-(\frac{s}{\alpha})^\beta} \beta s^{\beta-1} ds \\
&= \left(\frac{\alpha}{\alpha_{ok}}\right)^\beta \int_0^\infty e^{-(\frac{s}{\alpha})^\beta} \beta s^{\beta-1} \frac{1}{\alpha^\beta} ds \\
&= \left(\frac{\alpha}{\alpha_{ok}}\right)^\beta \int_0^\infty e^{-(\frac{s}{\alpha})^\beta} \beta s^{\beta-1} \frac{1}{\alpha^\beta} ds \\
&= \left(\frac{\alpha}{\alpha_{ok}}\right)^\beta \\
&= \frac{1}{\left(\frac{\alpha_{ok}}{\alpha_{deg}}\right)^\beta + 1} \tag{4.6}
\end{aligned}$$

De 4.4 et 4.6 il vient :

$$P(S_{deg} > S_{ok}^*) = \left(\frac{1}{\left(\frac{\alpha_{ok}}{\alpha_{deg}}\right)^\beta + 1}\right)^K \tag{4.7}$$

$\alpha_{ok} \geq \alpha_{deg}$, et en prenant un $K > 20$, on obtient $P(S_{deg} > S_{ok}^*) < \left(\frac{1}{2}\right)^K < 10^{-6}$

Ainsi, la probabilité que la taille de \hat{F}^{sys} soit définie par le maximum des temps de séjour dans l'état "ok" est supérieure à $1 - 10^{-6}$.

$\mathbb{E}(S_{ok}^*)$ est donc une estimation de la taille de \hat{F}^{sys} .

Pour conclure, avec $\beta_{ok} = \beta_{deg}$, $\alpha_{ok} \geq \alpha_{deg}$ et $K > 20$, la taille de \hat{F}^{sys} sera située autour de $\alpha_{ok} \int_0^\infty (1 - (1 - e^{-(u)^{\beta_{ok}}})^K) du$. Elle évolue donc linéairement avec α_{ok}

Fixer des paramètres α_{deg} et $\beta_{deg} = \beta_{ok}$, et faire varier α_{ok} au delà de α_{deg} fait varier linéairement la taille de \hat{F}^{sys} .

On appellera $\Omega_{\alpha_{ok}}$ l'ensemble choisi de différents α_{ok} , conduisant à des tailles différentes de F^{sys} .

La taille de \hat{F}^{sys} dépendra de la base considérée, mais elle sera proche de

$$\alpha_{ok} K \beta_{ok} \int_0^\infty u (1 - e^{-(u)^{\beta_{ok}}})^{K-1} e^{-(u)^{\beta_{ok}}} (u)^{\beta_{ok}-1} du \tag{4.8}$$

Ainsi, en faisant varier α_{ok} au delà de α_{deg} , la taille de \hat{F} augmentera linéairement, ce qui permettra d'étudier l'impact du choix de la taille de F^{sys} sur l'apprentissage et l'inférence du MGD.

4.1.3 Lois de probabilités conditionnelles du système initial

Dans ce paragraphe, nous allons définir les lois de probabilité du MGDHW modélisant le fonctionnement du système exemple d'où seront tirées les trajectoires, et notamment de définir l'espace $\Omega_{\alpha_{ok}}$.

Pour décrire un MGDHW, il est nécessaire de déterminer les paramètres α_i et β_i des Weibull qui décrivent les temps de séjour (table 4.1) après transition, pour chaque état $i \in \Omega_X$, la distribution de l'état initial Q^{ini} (table 4.2) du système, et la table de transition naturelle Q^{sys} (table 4.3).

La table 4.1 présente un exemple de paramètres d'échelle et de forme des lois de Weibull décrivant les temps de séjour dans les états "ok" et "dégradé". Les paramètres de forme sont choisis identiques et les paramètres d'échelle sont choisis tels que $\alpha_{ok} \geq \alpha_{deg}$ afin d'assurer une variation de la taille de \hat{F}^{sys} linéairement par rapport à la variation de α_{ok} . α_{deg} est par exemple fixé à 100, et les β_{ok} et β_{deg} à 20. α_{ok} va ensuite varier de 100 à 1000 avec un pas de 100 pour tester différentes valeurs de α_{ok} et donc différentes tailles de \hat{F}^{sys} . Dans ces conditions, \hat{F}^{sys} variera d'une taille d'environ 110 éléments à une taille d'environ 1100, avec un pas de 110.

La table 4.2 présente un exemple de distribution de l'état initial Q^{ini} du système. Le système est ici supposé neuf à l'état initial. Si besoin, il est cependant possible de répartir les probabilités entre les différents états possibles.

La table 4.3 présente un exemple de la loi de probabilité Q^{sys} , qui est la distribution de la variable modélisant l'état de système, après une transition. Vu qu'il y a une transition, les probabilités que le système reste dans l'état "ok" ou dans l'état "dégradé" sont nulles. Le système ne pouvant passer seul d'un état à un meilleur état de fonctionnement, les probabilités que le système passe après transition de l'état "dégradé" vers l'état "ok" ou de l'état "panne" vers l'état "ok" ou vers l'état "dégradé" sont nulles. L'état de panne est absorbant, la probabilité que le système passe de "panne" à "panne" est égale à 1. Il reste à déterminer la probabilité que le système passe de l'état "ok" à l'état "dégradé", qui détermine la probabilité que le système passe de l'état "ok" vers l'état "panne". Toutes les valeurs sont possibles, et il a été choisi ici que 10% des transitions sortant de l'état "ok" allaient directement en état de panne. Tout autre paramétrage aurait pu être choisi, mais cela n'aura pas réellement un impact sur la validation du modèle, qui est l'objet de ce chapitre.

TABLE 4.1 – Paramètres de la Weibull caractérisant la LPC des temps de séjour après transition

X_{t-1}	$f_{W_{\alpha,\beta}}$	
	α	β
ok	$100 * u; u \in \llbracket 1; 10 \rrbracket$	20
dégradé	100	20

Après avoir défini les lois de probabilité décrivant le MGDHW initial, passons à la présentation du protocole expérimental mis en place.

TABLE 4.2 – LPC de l'état initial

Q^{ini}		
X_1		
ok	dégradé	panne
1	0	0

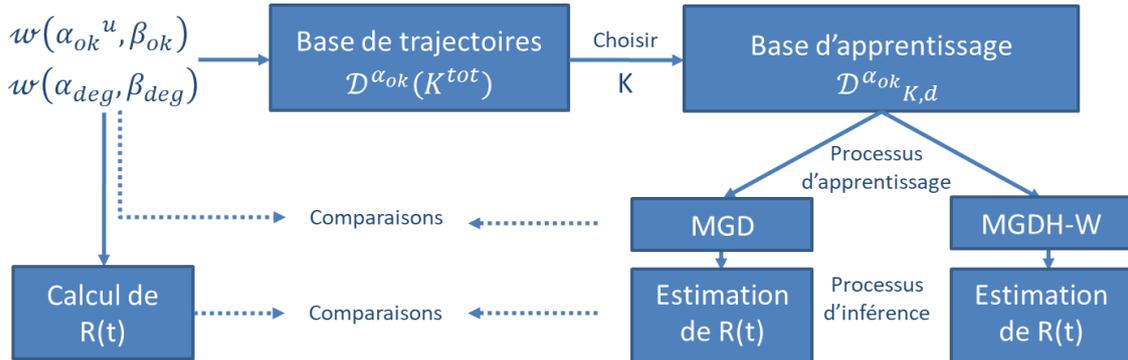
TABLE 4.3 – LPC de transition naturelle

Q^{sys}			
X_t			
X_{t-1}	ok	dégradé	panne
ok	0	0.9	0.1
dégradé	0	0	1
panne	0	0	1

4.1.4 Protocole expérimental

La première partie du protocole expérimental consiste donc en la création d'une base de K^{tot} trajectoires naturelles issues du système décrit par le MGDHW de la figure 4.2 paramétré par les lois présentées en 4.1.3, en créant une base différente pour chacun des α_{ok} de $\Omega_{\alpha_{ok}}$. Cette base de trajectoires est appelée $\mathcal{D}^{\alpha_{ok}}$. Le protocole de création de la base de trajectoire est expliqué dans la section 4.2.

FIGURE 4.3 – Protocole expérimental



Ensuite, puisqu'un objectif est d'étudier l'impact du nombre de trajectoires considérées lors de l'apprentissage sur la qualité de ce dernier, il est nécessaire de faire varier le nombre de trajectoires utilisées dans les divers apprentissages. Les différents nombres de trajectoires sont stockés dans l'espace Ω_K .

Pour chacun des K de Ω_K , K trajectoires sont tirées aléatoirement de la base $\mathcal{D}^{\alpha_{ok}}$ pour réaliser l'apprentissage. L'apprentissage va évidemment dépendre des tra-

jectoires sélectionnées, il est donc nécessaire de réaliser plusieurs tirages pour un nombre donné de trajectoires apprises. Appelons nb_sim le nombre de tirages par valeurs de K . Le d -ième tirage de K trajectoires constitue la base $\mathcal{D}_{K,d}^{\alpha_{ok}}$.

À partir de la base de trajectoires $\mathcal{D}_{K,d}^{\alpha_{ok}}$, un MGD et un MGDHW sont appris, et les apprentissages sont comparés, tel qu'expliqué dans la partie 4.3.

Pour comparer les inférences, on choisit d'estimer la fiabilité du système, puisque celle-ci est directement et facilement calculable (cf. section 4.4.3) depuis le MGDHW initial. Puisqu'une comparaison est établie avec le modèle initial, il est préférable ici de considérer des modèles MGDHW et MGD les mieux appris en terme de précision, c'est-à-dire de prendre $K = K^{tot}$, et $d = 1$ puisque de toute façon il n'y a qu'une façon de sélectionner K^{tot} trajectoires parmi K^{tot} .

Les comparaisons des estimations de fiabilité obtenues dans chacun des modèles donneront une idée de la pertinence de l'algorithme d'inférence développé en 3.2, testé ici sur un cas simple. L'inférence est présentée dans la partie 4.4.

La figure (4.3) et l'algorithme (2) résument et présentent le protocole.

Algorithm 2: Protocole expérimental

Entrées : K^{tot} , $\alpha_i, \beta_i; i \in \Omega_X$ fixés sauf α_{ok} , Q^{ini} , Q^{sys} , $\Omega_{\alpha_{ok}}$, Ω_K , nb_sim

Sorties : Fiabilité, temps de calcul

```

pour chaque  $\alpha_{ok} \in \Omega_{\alpha_{ok}}$  faire
     $\mathcal{D}^{\alpha_{ok}} \leftarrow$  algorithme 3( $Q^{ini}, Q^{sys}, \alpha_{ok}, \beta_{ok}, \alpha_{deg}, \beta_{deg}$ ) // cf. 4.2
    pour chaque  $K \in \Omega_K$  faire
        pour chaque  $d = 1, \dots, nb\_sim$  faire
            // phase d'apprentissage
             $\mathcal{D}_{K,d}^{\alpha_{ok}} \leftarrow$  Sélection aléatoire de K trajectoires de  $\mathcal{D}^{\alpha_{ok}}$ ;
            MGDHW $_{K,d}^{\alpha_{ok}} \leftarrow$  algorithme 6 ( $\mathcal{D}_{K,d}^{\alpha_{ok}}$ ) // cf. 4.3
            MGD $_{K,d}^{\alpha_{ok}} \leftarrow$  algorithme 7 ( $\mathcal{D}_{K,d}^{\alpha_{ok}}$ ) // cf. 4.3
        fin
    fin
    // phase d'inférence
     $R^{(\alpha_{ok})} \leftarrow$  algorithme 10 ( $\alpha_{ok}, \beta_{ok}, \alpha_{deg}, \beta_{deg}, Q^{ini}, Q^{sys}$ );
     $\hat{R}_{MGDHW}^{(\alpha_{ok})} \leftarrow$  algorithme 8 (MGDHW $_{K^{tot},1}^{\alpha_{ok}}$ ) // cf. 4.4
     $\hat{R}_{MGD}^{(\alpha_{ok})} \leftarrow$  algorithme 9 (MGD $_{K^{tot},1}^{\alpha_{ok}}$ ) // cf. 4.4
fin
    
```

4.2 Création d'une base de trajectoires

Pour générer une base de K^{tot} trajectoires, $\mathcal{D}^{\alpha_{ok}}$, à partir du MGDHW décrit en 4.1.3, pour un α_{ok} donné, il est nécessaire de commencer par déterminer une

profondeur T telle que toutes les séquences d'états du système indiquent un état de panne au bout de cette longueur T (section 4.2.1), puis l'étape suivante est la simulation de chacune des K^{tot} trajectoires, de longueur T (section 4.2.2).

L'algorithme 3 regroupe les différentes étapes nécessaires à la création de la base de données $\mathcal{D}^{\alpha_{ok}}$.

Algorithm 3: Création de $\mathcal{D}^{\alpha_{ok}}$: Simulation de K^{tot} trajectoires d'un système modélisé par un MGDHW.

Entrées :

1. Un MGDHW vérifiant la description probabiliste donnée en 4.1.3, *i. e.* des paramètres $Q^{(ini)}$, $Q^{(sys)}$, α_{ok} , β_{ok} , α_{deg} , β_{deg}
2. K^{tot}

Sorties : K^{tot} trajectoires constituées à chaque instant t de l'état x_t du système, et du temps de séjour restant s_t : $\mathcal{D}^{\alpha_{ok}}$

$T^{ini} \leftarrow$ algorithme 4 (α_{ok} , β_{ok} , α_{deg} , β_{deg} , K^{tot}) ;

$T \leftarrow T^{ini}$;

pour chaque $K = 1, \dots, K^{tot}$ **faire**

$\{(x_t, s_t)\}_{1 \leq t \leq T}^{(K)} \leftarrow$ algorithme 5 (MGDHW, T) // Simulation d'une trajectoire

si $x_T \neq \text{panne}$ **alors**

// nouveau T tel que $x_T = \text{panne}$

pour chaque $k = 1, \dots, K$ **faire**

pour chaque $t = T^{(k)}, \dots, T$ **faire**

$x_t^{(k)} \leftarrow \text{panne}$;

fin

fin

fin

fin

$\mathcal{D}^{\alpha_{ok}} \leftarrow \{ \{x_t\}_{1 \leq t \leq T}^{(K)} \}_{1 \leq K \leq K^{tot}}$ // stockage des trajectoires dans $\mathcal{D}^{\alpha_{ok}}$

4.2.1 Taille T des trajectoires de $\mathcal{D}^{\alpha_{ok}}$

L'algorithme 4 donne l'expression d'une profondeur telle que les K^{tot} trajectoires sont dans l'état de panne au bout de cette durée T .

Algorithm 4: Choix de la profondeur T

Entrées : α_{ok} , β_{ok} , α_{deg} , β_{deg} , K^{tot}

Sorties : T

$T = \lfloor F_{\mathcal{W}_{\alpha_{ok}, \beta_{ok}}}^{-1} ((1 - 10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + \lfloor F_{\mathcal{W}_{\alpha_{deg}, \beta_{deg}}}^{-1} ((1 - 10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + 2$

La suite de ce paragraphe démontre pourquoi ce choix de profondeur convient.

Soient S_{ok} et S_{deg} deux variables aléatoires suivant respectivement les distributions $\mathcal{W}(\alpha_{ok}, \beta_{ok})$ et $\mathcal{W}(\alpha_{deg}, \beta_{deg})$.

Soient T_{ok} et T_{deg} définis par :

$$T_{ok} = \lfloor F_{\mathcal{W}_{\alpha_{ok}, \beta_{ok}}}^{-1}((1 - 10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + 1 \quad (4.9)$$

$$T_{deg} = \lfloor F_{\mathcal{W}_{\alpha_{deg}, \beta_{deg}}}^{-1}((1 - 10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + 1 \quad (4.10)$$

T_{ok} et T_{deg} sont tels que :

$$\begin{aligned} P(S_{ok} < T_{ok}) &= F_{\mathcal{W}(\alpha_{ok}, \beta_{ok})}(T_{ok}) \\ &= F_{\mathcal{W}(\alpha_{ok}, \beta_{ok})}(\lfloor F_{\mathcal{W}_{\alpha_{ok}, \beta_{ok}}}^{-1}((1 - 10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + 1) \\ &\geq F_{\mathcal{W}(\alpha_{ok}, \beta_{ok})}(F_{\mathcal{W}_{\alpha_{ok}, \beta_{ok}}}^{-1}((1 - 10^{-6})^{\frac{1}{2K^{tot}}})) \\ &\geq (1 - 10^{-6})^{\frac{1}{2K^{tot}}} \end{aligned} \quad (4.11)$$

$$\begin{aligned} P(S_{deg} < T_{deg}) &= F_{\mathcal{W}(\alpha_{deg}, \beta_{deg})}(T_{deg}) \\ &= F_{\mathcal{W}(\alpha_{deg}, \beta_{deg})}(\lfloor F_{\mathcal{W}_{\alpha_{deg}, \beta_{deg}}}^{-1}((1 - 10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + 1) \\ &\geq F_{\mathcal{W}(\alpha_{deg}, \beta_{deg})}(F_{\mathcal{W}_{\alpha_{deg}, \beta_{deg}}}^{-1}((1 - 10^{-6})^{\frac{1}{2K^{tot}}})) \\ &\geq (1 - 10^{-6})^{\frac{1}{2K^{tot}}} \end{aligned} \quad (4.12)$$

Or

$$S_{ok} < T_{ok} \text{ et } S_{deg} < T_{deg} \Rightarrow S_{ok} + S_{deg} < T_{ok} + T_{deg} \quad (4.13)$$

donc

$$\begin{aligned} P(S_{ok} + S_{deg} < T_{ok} + T_{deg}) &\geq P(S_{ok} < T_{ok} \cap S_{deg} < T_{deg}) \text{ (de 4.13)} \\ &= P(S_{ok} < T_{ok}) * P(S_{deg} < T_{deg}) \text{ (car } S_{ok} \text{ et } S_{deg} \text{ indép.)} \\ &\geq (1 - 10^{-6})^{\frac{1}{2K^{tot}}} * (1 - 10^{-6})^{\frac{1}{2K^{tot}}} \text{ (de 4.11 et 4.12)} \\ &\geq (1 - 10^{-6})^{\frac{1}{K^{tot}}} \end{aligned} \quad (4.14)$$

Soit $T = T_{ok} + T_{deg}$

$$\begin{aligned} P(X_T^{(1)} = \text{panne} \cap \dots \cap X_T^{(K^{tot})} = \text{panne}) &= P(X_T = \text{panne})^{K^{tot}} \\ &= P(S_{ok} + S_{deg} < T)^{K^{tot}} \\ &= P(S_{ok} + S_{deg} < T_{ok} + T_{deg})^{K^{tot}} \\ &\geq (1 - 10^{-6})^{\frac{K^{tot}}{K^{tot}}} \text{ (de 4.14)} \end{aligned} \quad (4.15)$$

La profondeur de $T = \lfloor F_{\mathcal{W}_{\alpha_{ok}, \beta_{ok}}}^{-1}((1-10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + \lfloor F_{\mathcal{W}_{\alpha_{deg}, \beta_{deg}}}^{-1}((1-10^{-6})^{\frac{1}{2K^{tot}}}) \rfloor + 2$ permet d'assurer à plus de 99.9999% que les K^{tot} trajectoires sont dans l'état de panne au bout de T tranches de temps.

Vu que 99.9999% n'est pas égal à 100%, il est tout à fait possible qu'une trajectoire - ou plus - ne soit pas dans l'état de panne au bout de ce temps T . Si c'est le cas, T doit être augmenté, de manière à ce que cette trajectoire soit en panne à l'instant T . La procédure est la suivante, présentée dans un cas plus général non limité à trois états de fonctionnement :

Si l'état à l'instant T , x_T , est l'état précédent l'état de panne, il suffit d'ajouter $\lfloor s_T \rfloor + 1$ à T pour que la trajectoire soit en état de panne.

S'il s'agit d'un autre état, l'instant d'apparition de l'état de panne est inconnu. x_T est l'état courant à T , s_T est le temps restant dans cet état. Au temps $T + \lfloor s_T \rfloor + 1$, il y a un changement d'état. Un nouvel état $x_{T+\lfloor s_T \rfloor+1} \sim Q^{sys}(x_T, \cdot)$ est généré. S'il s'agit de l'état de panne, T est mis à jour : $T = T + \lfloor s_T \rfloor + 1$. S'il ne s'agit pas de l'état de panne, il faut simuler le temps de séjour dans l'état correspondant, puis simuler un nouvel état, et ainsi de suite.

Enfin, si le stockage des bases de données nécessite que toutes les trajectoires aient la même profondeur, il faut encore ajouter des états de panne aux trajectoires précédemment simulées pour qu'elles aillent bien jusqu'au nouveau T .

Il était plus intéressant en terme de vitesse de calculs pour la création de la base de trajectoire de fixer au préalable une profondeur des trajectoires plutôt que de simuler des trajectoires sans fin prédéfinie.

Maintenant que la longueur des trajectoires est déterminée, montrons comment simuler une des K^{tot} trajectoires.

4.2.2 Simulation d'une trajectoire

Nous utilisons la simulation de Monte Carlo [HENRION 1988] pour générer la base de données de trajectoires.

Commençons par simuler une trajectoire de longueur T générée par le système modélisé par un MGDHW.

À $t = 1$, un état de fonctionnement est généré selon la loi de probabilité Q^{ini} , puis un temps de séjour dans l'état courant est généré selon la distribution de Weibull de paramètres associés à l'état courant. C'est l'initialisation de la trajectoire.

À un instant donné t , le précédent temps de séjour calculé s_{t-1} renseigne si un changement d'état s'est produit entre les instants $t - 1$ et t . S'il n'y a pas de changement d'état, la valeur x_{t-1} est affecté à x_t , et s_{t-1} est décrémenté d'une unité avant d'être affecté à s_t . S'il y a eu un changement d'état, un nouvel état x_t doit être

génééré selon $Q^{sys}(x_{t-1}, \cdot)$, ainsi qu'un nouveau temps de séjour s_t selon $\mathcal{W}(\alpha_{x_t}, \beta_{x_t})$.

L'algorithme 5 présente ceci.

Algorithm 5: Simulation d'une trajectoire d'un système représenté par un MGDHW

Entrées :

1. Un MGDHW (*i. e.* des paramètres Q^{ini}, Q^{sys} , α et β pour les états non absorbants)
2. La taille de la trajectoire à simuler, notée T

Sorties : $\{(x_t, s_t)\}_{1 \leq t \leq T}$, une trajectoire de taille T constituée à chaque instant t de l'état x_t du système, et du temps de séjour restant s_t

// Initialisation de la trajectoire

Générer $x_1 \sim Q^{ini}$;

Générer $s_1 \sim \mathcal{W}(\alpha_{x_1}, \beta_{x_1})$;

pour chaque $t = 2, \dots, T$ **faire**

si $s_{(t-1)} < 1$ **alors**
 // Transition naturelle
 Générer $x_t \sim Q^{sys}(x_{t-1}, \cdot)$;
 Générer $s_t \sim \mathcal{W}(\alpha_{x_t}, \beta_{x_t})$;

sinon
 $x_t \leftarrow x_{t-1}$;
 $s_t \leftarrow s_{t-1} - 1$;

fin

fin

4.2.3 Stockage

Nous disposons de K^{tot} ensembles $\{(x_t, s_t)\}_{1 \leq t \leq T}$. Le but de cette section 4.2 est de créer une base de données semblables à celles qui pourraient être disponibles dans la réalité. Les seules données qu'il est possible de connaître réellement, ce sont les états aux temps discrets $1, \dots, T$. Les valeurs réelles des temps de séjour, les instants réels des changements d'état ne sont pas connus. Ainsi ne sont stockés dans $\mathcal{D}^{\alpha_{ok}}$ que les séquences des états aux temps $1, \dots, T$.

4.2.4 Exemple

La table 4.4 donne un exemple de trois trajectoires supposées issues du système de production décrit précédemment.

Plutôt que d'afficher l'ensemble des états, il suffit d'indiquer le nombre de tranches d'états où le système est dans l'état considéré.

Ces quatre trajectoires correspondent aux successions d'états de la figure 4.4

TABLE 4.4 – Quatre trajectoires simulées du système de production. Dans ce contexte le nombre de trajectoires K est égal à 4

	Etat courant	Temps de séjour
Traj. 1	ok	2
	dégradé	3
	panne	3
Traj. 2	ok	3
	dégradé	1
	panne	4
Traj. 3	ok	7
	panne	1
Traj. 4	ok	1
	dégradé	3
	panne	4

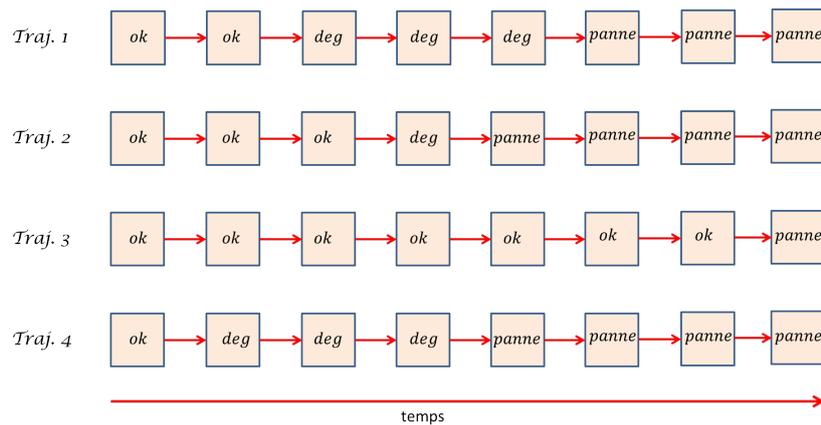


FIGURE 4.4 – Exemple de trajectoires

4.3 Apprentissages du MGD et du MGDHW

L'objet de cette partie est de montrer comment apprendre les paramètres des lois de probabilités conditionnelles d'un MGDHW et d'un MGD à partir des bases de trajectoires précédemment constituées. Ensuite, une évaluation de ces apprentissages est mise en place, ainsi qu'une comparaison de l'espace mémoire requis. Tout ceci est effectué avec différents nombres de trajectoires constituant la base d'apprentissage, et avec différentes valeurs de paramètres de simulation de ces trajectoires, afin d'étudier leur impact.

Algorithm 6: Phase d'apprentissage : apprentissage du MGDHW $_{K,d}^{\alpha_{ok}}$ sur la base $\mathcal{D}_{K,d}^{\alpha_{ok}}$

Entrées : $\mathcal{D}_{K,d}^{\alpha_{ok}}$
Sorties : MGDHW $_{K,d}^{\alpha_{ok}}$
 Apprentissage du MGDHW $_{K,d}^{\alpha_{ok}}$ // sous-section 4.3.1
 Temps de calcul de l'apprentissage;
 Poids de MGDHW $_{K,d}^{\alpha_{ok}}$;
 $D_{KL}(\mathcal{W}(\alpha^{\alpha_{ok}}, \beta), \mathcal{W}(\hat{\alpha}^{\alpha_{ok}, K, d}, \hat{\beta}^{K, d}))$;

Algorithm 7: Phase d'apprentissage : apprentissage du MGD $_{K,d}^{\alpha_{ok}}$ sur la base $\mathcal{D}_{K,d}^{\alpha_{ok}}$

Entrées : $\mathcal{D}_{K,d}^{\alpha_{ok}}$
Sorties : MGD $_{K,d}^{\alpha_{ok}}$
 Apprentissage du MGD $_{K,d}^{\alpha_{ok}}$ // sous-section 4.3.1
 Temps de calcul de l'apprentissage;
 Poids de MGD $_{K,d}^{\alpha_{ok}}$;
 $D_{KL}(F^{(\alpha_{ok})}, \hat{F}^{(\alpha_{ok}, K, d)})$;

4.3.1 Phase d'apprentissage

En se donnant un nombre K de trajectoires simulées, et un paramètre α_{ok} , on apprend un MGD et un MGDHW, en variant les K trajectoires choisies aléatoirement.

Soit $\mathcal{D}_{K,d}^{\alpha_{ok}}$ une base de trajectoires. Ces données contiennent l'historique de fonctionnement, sans maintenance, de K machines de production identiques et indépendantes, de leur mise en service jusqu'à leur panne (cf. les trajectoires de la table 4.4 par exemple). Pour chacune des K machines, nous possédons la liste à chacun des instants de la valeur de l'état de fonctionnements, liste qui peut être mis sous la forme de listes des différents états traversés et des temps de séjour associé à chacun de ces états.

Il s'agit à présent de calculer les estimations des tables relatives au MGDHW et au MGD, c'est-à-dire \hat{Q}^{sys} qui est à la fois relatif au MGDHW et au MGD, $\hat{\alpha}$ et $\hat{\beta}$ qui sont relatifs au MGDHW, et \hat{F}_{MGD} , qui est relatif au MGD. Les estimations sont apprises à partir de $\mathcal{D}_{K,d}^{\alpha_{ok}}$.

Posons $S_{max}^{(\alpha_{ok}, K, d)} = \max_i(S_{max_{i,j,K}})$, noté S_{max} par la suite, le maximum des temps de séjour passés dans un quelconque état par les K trajectoires de $\mathcal{D}_{K,d}^{\alpha_{ok}}$.

La table 4.5 fournit un exemple de nombre d'occurrences, noté $N_{x,s}$, où le système est resté $s \in S = \{1, \dots, S_{max}\}$ unités de temps dans l'état $x \in X = \{\text{ok}, \text{dégradé}, \text{panne}\}$.

L'ensemble des tables de cette partie 4.3 présentant l'apprentissage ne sont pas relatives à des trajectoires de la base de données issues du MGDHW présenté dans la partie 4.2 et dont l'inférence est étudiée dans la partie 4.4. En effet, les tables de probabilité correspondant à la base de données de la partie 4.2 sont trop grandes pour pouvoir être présentées dans ce rapport, puisque la thèse s'intéresse à de grandes dimensions de temps de séjour. Un autre exemple a donc été choisi pour la partie sur l'apprentissage, à titre illustratif, indépendamment des autres parties du chapitre.

TABLE 4.5 – Nombre d'observations des temps de séjour allant de 1 à S_{max} trimestres pour chaque état (excepté l'état absorbant panne) dans les $K = 2000$ trajectoires disponibles.

état	Temps de séjour											total
	1	2	3	4	5	6	7	8	9	10	11	
ok	14	155	372	510	429	316	133	60	10	0	1	2000
dégradé	68	623	798	295	36	2	0	0	0	0	0	1822

La table 4.6 donne un exemple de nombre de transitions, noté $N_{x',x}$, où le système est passé de l'état x' à x après un changement d'état.

TABLE 4.6 – Nombre d'observations des transitions entre les états

État précédent	État suivant			total
	ok	dégradé	panne	
ok	0	1822	178	2000
dégradé	0	0	1822	1822

Présentons maintenant les estimations de Q^{sys} du MGD et du MGDHW, de F^{sys} du MGD, effectuées par la méthode de l'estimateur de *maximum a posteriori*, et l'estimation des paramètres α et β pour le MGDHW obtenues par la méthode de l'estimateur de *maximum a posteriori* et par une régression linéaire. Q^{ini} peut être défini *a priori*.

MGD

Commençons par traiter le modèle reposant sur un MGD. Dans ce cas, il y a deux LPC à estimer : la LPC des temps de séjour, notée \hat{F}_{MGD} et la matrice de transition naturelle, notée \hat{Q}_{MGD}^{sys} .

Les expressions de ces estimations par la méthode de l'estimateur de *maximum a posteriori* sont définies par :

$$\hat{Q}^{sys}(i, j) = \frac{N_{i,j} + \delta_{i < j}}{\sum_{j'} N_{i,j'} + \delta_{i < j}} \quad (4.16)$$

et

$$\hat{F}(i, k) = \frac{N_{i,k} + 1}{\sum_{k'} N_{i,k'} + 1} \quad (4.17)$$

En effet, il est certain que les probabilités de passer d'un état à un état "supérieur" sont nulles, car le système ne peut se réparer lui-même, donc que les $\hat{Q}^{sys}(i, j)$ seront nécessairement nuls pour les $i \geq j$.

Concernant la table de probabilité conditionnelle du temps de séjour après changement d'état, pour éviter d'obtenir des estimations de probabilités nulles pour des réalisations possibles, un *a priori* de Dirichlet uniforme est utilisé.

La table 4.7 illustre l'estimation de \hat{Q}^{sys} obtenue à partir des observations données par la table 4.6, et la table 4.8 illustre l'estimation \hat{F}^{sys} obtenir à partir des observations enregistrées dans la table 4.5.

TABLE 4.7 – Estimation de la loi de probabilité conditionnelle de transition naturelle du système

		\hat{Q}^{sys}		
		X_t		
X_{t-1}		ok	dégradé	panne
ok		0	0.9116	0.0894
dégradé		0	0	1
panne		0	0	1

TABLE 4.8 – Estimation de la loi de probabilité des temps de séjour s'il y a une transition

		S_t										
X_t		1	2	3	4	5	6	7	8	9	10	11
ok		0.0075	0.0776	0.1855	0.2541	0.2138	0.1576	0.0666	0.0303	0.0055	0.0005	0.0010
deg		0.0376	0.3404	0.4359	0.1615	0.0202	0.0016	0.00055	0.00055	0.00055	0.00055	0.00055

MGDHW

L'apprentissage du MGDHW consiste en l'estimation de la matrice de transition naturelle, notée \hat{Q}^{sys} et l'estimation des paramètres $\hat{\alpha}$ et $\hat{\beta}$ caractérisant les lois de Weibull suivies par les temps de séjour lorsqu'il y a une transition 3.5.

L'estimation \hat{Q}^{sys} de Q^{sys} s'effectue de la même manière que pour le MGD (4.16), par la méthode de l'estimateur de *maximum a posteriori*, avec un *a priori* qui est un dirac, pour assurer que les probabilités qui doivent être nulle soient bien estimées ainsi :

$$\hat{Q}^{sys}(i, j) = \frac{N_{i,j} + \delta_{i < j}}{\sum_{j'} N_{i,j'} + \delta_{i < j}} \quad (4.18)$$

L'estimation des paramètres α et β des lois de Weibull commence par l'estimation de $P(S \geq s)$ pour chacun des s par la fréquence d'observations des cas où $S \geq s$, en utilisant la méthode de l'estimateur du *maximum a posteriori* avec un *a priori* de Dirichlet égal à 0.1. L'*a priori* est choisi petit pour ne pas fausser les résultats puisqu'un cumul est effectué.

$$\widehat{P(S \geq s)} = \frac{\sum_{k=s}^{S_{max}} N_{i,k} + 0.1}{\sum_{k'} N_{i,k'} + 0.1} \quad (4.19)$$

or

$$P(S \geq s) = e^{-\left(\frac{s}{\alpha}\right)^\beta} \quad (4.20)$$

c'est à dire

$$\log\left(\log\left(\frac{1}{\widehat{P(S \geq s)}}\right)\right) = \beta \log(s) - \beta \log(\alpha) \quad (4.21)$$

Une régression linéaire entre les $\log(-\log(\widehat{P(S \geq s)}))$ et les $\log(s)$ donne une estimation $\hat{\beta}$ de β et une estimation de $\beta \log(\alpha)$ ce qui conduit à une estimation $\hat{\alpha}$ de α .

Il existe plusieurs tests d'adéquation pour vérifier la pertinence de l'ajustement aux données d'une loi de Weibull.

Le test choisi ici consiste à calculer

$$\frac{1}{N} \sum |\log(\frac{Y_{REX}}{Y_W})|$$

Plus la valeur obtenue est petite, et plus le test est bon.

La table 4.9 illustre l'estimation de $P(S \geq s)$ obtenue à partir des observations de la table 4.5.

TABLE 4.9 – Estimation de $P(S \geq s)$

état	Temps de séjour										
	1	2	3	4	5	6	7	8	9	10	11
ok	1	0.9925	0.9150	0.7295	0.4754	0.2616	0.1039	0.0373	0.0070	0.0015	0.0010
dégradé	1	0.9624	0.6220	0.1860	0.0245	0.0044	0.0028	0.0022	0.0017	0.0011	0.0006

La table 4.10 illustre le résultat de la régression linéaire entre les $\log(s)$ et les $\log(-\log(P(\widehat{S} \geq s)))$, issues de l'exemple présenté dans la table 4.5.

L'apprentissage a permis d'ajuster un MGDHW aux données, ainsi qu'un MGD. Il est maintenant intéressant d'étudier la qualité des apprentissages, en examinant l'impact du nombre de trajectoires considérées dans l'apprentissage et l'impact de la valeur du α_{ok} initial.

4.3.2 Qualité d'un apprentissage

Chacun des deux modèles ayant un nombre différent de paramètres, le nombre K de trajectoires considérées dans l'apprentissage peut avoir un impact sur la qualité et la vitesse de ce dernier.

Pour mesurer la qualité de l'apprentissage du MGDHW, les paramètres de Weibull appris vont être comparés avec les paramètres originaux, en comparant les densités des Weibull. Dans le cas du MGD, la table \hat{F}^{sys} va être comparée à la discrétisation des Weibull initiales. Pour cela, il est nécessaire d'introduire une mesure, comme par exemple la divergence de Kullback-Leibler [KULLBACK 1952].

Cette dernière fournit une mesure non-symétrique de la similarité de deux distributions de probabilité P et Q. Elle peut être interprétée comme étant la perte d'information si P est modélisé au moyen de Q. En conséquence, plus $D_{KL(P||Q)}$ est petit, et plus P et Q sont similaires.

TABLE 4.10 – Paramètre de Weibull et test d'adéquation

α	β	test d'adéquation
5.9137	3.8934	0.3514

Pour deux distributions de probabilités discrètes P et Q la divergence de Kullback–Leibler de Q par rapport à P est définie par l'équation 4.22 :

$$D_{KL}(P||Q) = \sum_k P(k) \log \frac{P(k)}{Q(k)} \quad (4.22)$$

sous condition que $Q(k) = 0 \Rightarrow P(k) = 0$

Pour des distributions P et Q continues, on utilise une intégrale

$$D_{KL}(P||Q) = \int p(s) \log \frac{p(s)}{q(s)} ds \quad (4.23)$$

Cas continu

L'objectif ici est de comparer les distributions $\mathcal{W}(\hat{\alpha}, \hat{\beta})$ et $\mathcal{W}(\alpha, \beta)$ pour juger de la qualité de l'apprentissage, $\mathcal{W}(\alpha, \beta)$ étant modélisée au moyen de $\mathcal{W}(\hat{\alpha}, \hat{\beta})$.

Pour des distributions de Weibull, BAUCKHAGE montre dans BAUCKHAGE 2013 que la divergence de Kullback-Leibler s'écrit comme suit :

$$D_{KL}(\mathcal{W}(\alpha, \beta)||\mathcal{W}(\hat{\alpha}, \hat{\beta})) = \log\left(\frac{\beta}{\alpha^{\beta}}\right) - \log\left(\frac{\hat{\beta}}{\hat{\alpha}^{\hat{\beta}}}\right) + (\beta - \hat{\beta}) * \left(\log(\alpha) - \frac{\gamma}{\beta}\right) + \left(\frac{\alpha}{\hat{\alpha}}\right)^{\hat{\beta}} * \Gamma\left(\frac{\hat{\beta}}{\beta} + 1\right) - 1 \quad (4.24)$$

avec $\gamma = 0.5772$ la constante d'Euler-Mascheroni et Γ la fonction gamma.

Cas discret

Pour évaluer la qualité de l'apprentissage de la table de probabilité conditionnelle \hat{F}^{sys} , il convient d'abord de calculer la table théorique F^{sys} du MGD correspondant au MGDHW. La variable des temps de séjour, notée S , suit dans le MGDHW une loi de Weibull $\mathcal{W}(\alpha, \beta)$. Pour obtenir F^{sys} , il faudrait discrétiser les distributions $\mathcal{W}(\alpha, \beta)$ associées à chacun des états. Le support d'une distribution de Weibull étant infini, il convient de déterminer une borne. Or les formules qui servent à comparer deux tables de probabilité nécessitent que ces tables soient de même dimension, \hat{F}^{sys} doit donc être comparée à une matrice F^{sys} de même taille. F^{sys} doit donc être construite de la même taille que \hat{F}^{sys} . La troncature de la Weibull entraîne nécessairement une perte d'information, mais \hat{F}^{sys} est telle que tous les temps de séjour des K trajectoires sont pris en comptes dans \hat{F}^{sys} . La perte d'information est donc minimisée.

Soit F^{sys} la table contenant les discrétisations des distributions de Weibull de chacun des états, les discrétisations étant effectuées avec une borne égale à la taille

de \hat{F}^{sys} .

Montrons comment comparer les lignes des tables F^{sys} et \hat{F}^{sys} , pour chacun des états.

Il s'agit donc de comparer $P(S_t|S_{t-1} = 1, X_t = i)$ et son estimation pour chacun des états i .

La divergence de Kullback Leibler s'écrit alors

$$D_{KL}(F^{sys}(i, :)||\hat{F}^{sys}(i, :)) = \sum_k F^{sys}(i, k) \log \frac{F^{sys}(i, k)}{\hat{F}^{sys}(i, k)} \quad (4.25)$$

\hat{F}^{sys} est estimé au moyen de l'estimateur par *maximum a posteriori* avec *a priori* de Dirichlet (cf. 4.17), donc tous les $\hat{F}^{sys}(i, k)$ sont non nuls.

Les divergences

$$D_{KL}(\mathcal{W}(\alpha, \beta), \mathcal{W}(\hat{\alpha}, \hat{\beta}))$$

et

$$D_{KL}(F^{sys}(i, :), \hat{F}^{sys}(i, :))$$

permettent donc de juger respectivement de la qualité de l'apprentissage du MGDHW et de celle du MGD.

Nous avons donc un MGDHW, et un MGD, appris à partir de $\mathcal{D}_{K,d}^{\alpha_{ok}}$. Appelons-les MGDHW $_{K,d}^{\alpha_{ok}}$ et MGD $_{K,d}^{\alpha_{ok}}$.

La partie suivante présente les résultats de ces évaluations d'apprentissage, ainsi que les temps de calcul, et l'espace de stockage utilisé par les modèles, par rapport au nombre de trajectoires et à la taille de F^{sys} .

4.3.3 Résultats : comparaisons des qualités d'apprentissage

Précision

La figure 4.5 présente l'évolution de la divergence de Kullback-Leibler de l'estimation de la distribution de Weibull associée à l'état "ok" du MGDHW en fonction du nombre de trajectoires considérées dans l'apprentissage et de la valeur de α_{ok} . La figure 4.6 présente l'évolution de la divergence de Kullback-Leibler de la table de probabilité conditionnelle du temps de séjour dans l'état "ok" en fonction du nombre de trajectoires considérées dans l'apprentissage et de la valeur de α_{ok} , qui est directement proportionnelle à la valeur de la taille de F^{sys} .

Il est visible sur la figure 4.6 que le choix de α_{ok} , donc le choix de la taille de F^{sys} , a un impact sur la qualité de l'apprentissage fonction du nombre de trajectoires

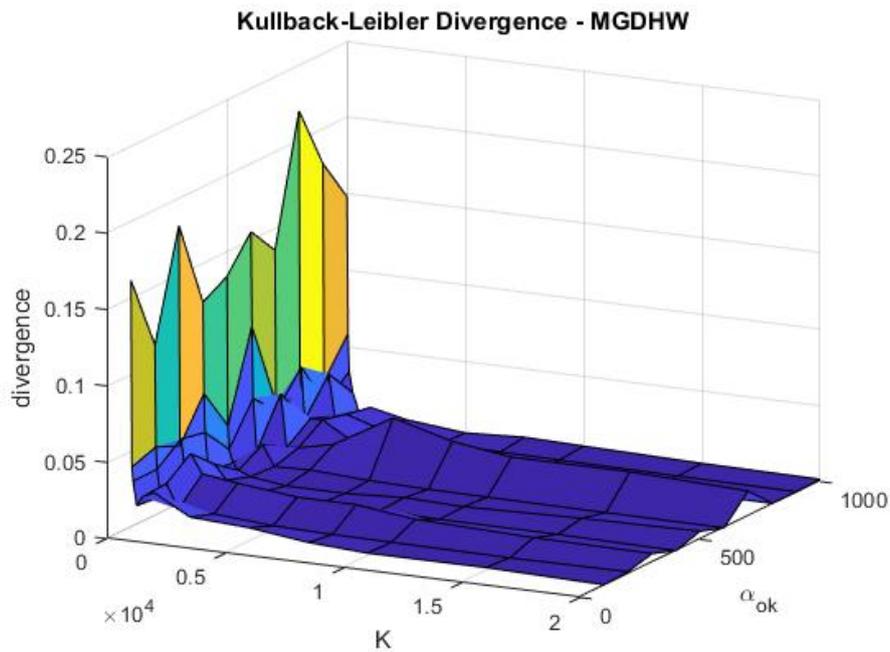


FIGURE 4.5 – Évolution de la divergence de Kullback-Leibler entre les distributions de Weibull associées à l'état "ok" apprise et théorique, en fonction du nombre de trajectoires apprises et du paramètre α_{ok}

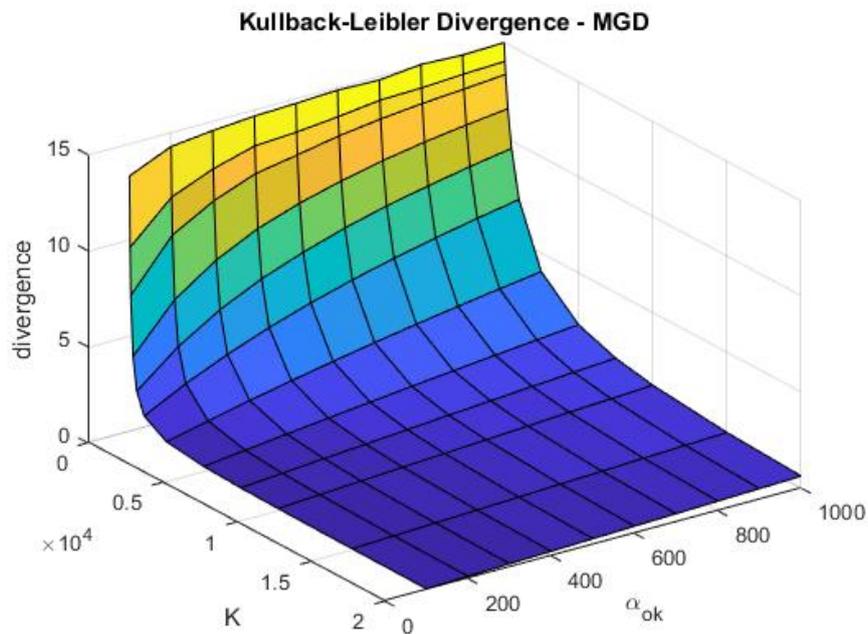


FIGURE 4.6 – Évolution de la divergence de Kullback-Leibler entre les tables de probabilité conditionnelle de la durée de séjour dans l'état "ok" apprise et théorique en fonction du nombre de trajectoires apprises et du paramètre α_{ok}

appries dans le cas d'un MGD. Pour un grand α_{ok} (*i. e.* une grande taille de F), un grand nombre de trajectoires est ainsi nécessaire pour minimiser la divergence de Kullback-Leibler des tables de probabilités conditionnelles. De plus, pour un grand α_{ok} , quel que soit le nombre de trajectoires considérés ici, la divergence de Kullback-Leibler est nettement supérieure à celle obtenue avec le MGDHW. Pour un grand α_{ok} , le MGDHW est mieux appris que le MGD. Il faudrait envisager un nombre encore plus important de trajectoires pour approcher la qualité d'apprentissage du MGDHW avec un MGD. Or plus il faut de trajectoires d'apprentissage, et plus il faut de temps pour les générer (que ce soit la réalité ou des simulations), d'espace de stockage, et de temps de calcul d'apprentissage.

Sur la figure 4.5, on remarque que le choix d' α_{ok} n'a pas d'impact sur la divergence de Kullback-Leibler et ce quel que soit le nombre de trajectoires apprises dans le cas d'un MGDHW. Un grand α_{ok} ne nécessite pas plus de trajectoires qu'un petit α_{ok} , et la divergence de Kullback-Leibler est inférieure à 0.05, ce qui atteste d'une bonne qualité d'apprentissage.

Pour un grand α_{ok} , un petit nombre de trajectoires considérées dans l'apprentissage des paramètres du MGDHW suffit alors pour une bonne qualité, alors que le MGD en nécessite un nombre nettement plus important.

Sur cet aspect, l'approche par MGDHW semble donc avantageuse.

Temps de calcul

Le temps de calcul de l'apprentissage du MGD (figure 4.8) et celui de l'apprentissage du MGDHW (figure 4.7) sont quasiment identiques. En effet, l'apprentissage du MGDHW nécessite les mêmes étapes de comptage et de stockage d'occurrences que le MGD.

Pour un petit nombre de trajectoires, la valeur de α , c'est à dire la taille de F^{sys} pour le MDG, a peu d'impact. Lorsqu'il y a peu de trajectoires, les paramètres sont moins bien appris, mais le calcul est d'autant plus rapide. Par contre, plus le nombre K de trajectoires considérées dans l'apprentissage augmente, plus la valeur de α a d'impact, et ce qu'il s'agisse du MGD ou du MGDHW.

En effet, lorsque α_{ok} est élevé, les temps de séjour dans l'état courant vont prendre plus de valeurs différentes qu'avec un α_{ok} inférieur, et de même, plus K est grand, et plus il y a de chance de trouver au moins une occurrence pour un temps de séjour donné. L'étape de comptage des occurrences de chacune des valeurs prises par le temps de séjour est aussi longue dans les deux situations.

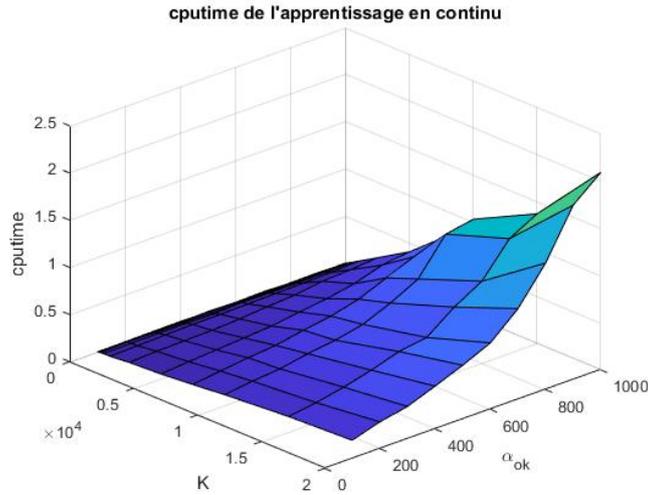


FIGURE 4.7 – Temps de calcul apprentissage MGDHW

Stockage

Intéressons-nous maintenant à la comparaison des espaces de stockage utilisés par le MGD et pour le MGDHW.

Les figures 4.9 et 4.10 présentent les mémoires occupées respectivement par le stockage du MGDHW et par le stockage du MGD en fonction de la valeur de α et du nombre K de trajectoires considérées dans l'apprentissage.

Le volume occupé par le MGDHW ne varie pas, les lois de Weibull ne contenant toujours que deux paramètres.

Par contre, la mémoire prise par le MGD augmente quadratiquement lorsque la valeur de α augmente, c'est à dire lorsque la taille de F^{sys} augmente. Pour un $\alpha = 400$, le MGD va occuper 1000 fois plus de place que le MGDHW. Pour un $\alpha = 1000$, le MGD va occuper plus de 6000 fois plus de place que le MGDHW.

L'inflexion pour des petits nombres de trajectoires considérées s'explique par le fait que la table F^{sys} est apprise avec moins de précision vu qu'il y a moins de trajectoires, il y a beaucoup d'élément de $\Omega_S = \{1, \dots, N_S\}$ qui sont sans occurrence. La table \hat{F}^{sys} est dite creuse, elle occupe moins de place en mémoire.

4.4 Comparaison des inférences

Plusieurs MGDHW et MGD ayant été appris pour différentes valeurs de α_{ok} et à partir de différentes sélections de trajectoires de $\mathcal{D}^{\alpha_{ok}}$, l'algorithme mis en place dans la section 3.2 peut être expérimenté sur les MGDHW, et les temps de calcul et les résultats obtenus peuvent être comparés avec ceux obtenus via l'algorithme

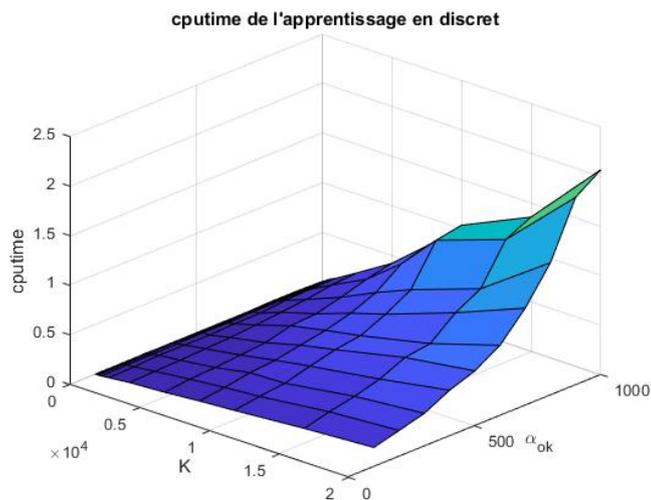


FIGURE 4.8 – Temps de calcul apprentissage MGD

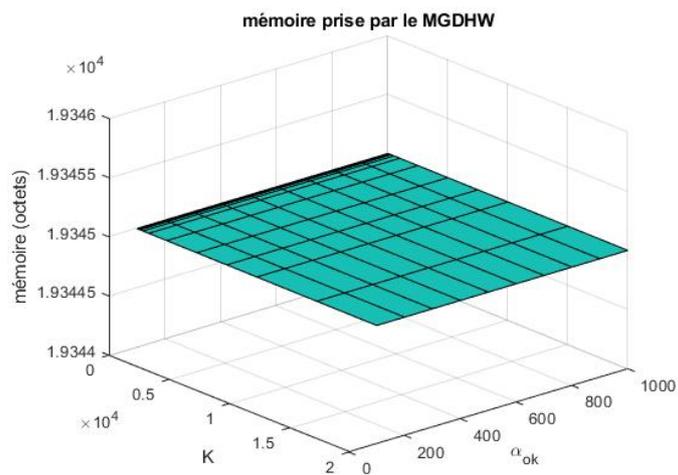


FIGURE 4.9 – Stockage MGDHW

d'élimination des variables appliqué aux MGD.

L'objectif est d'étudier l'impact de la taille de F^{sys} et de α_{ok} sur l'inférence, respectivement dans un MGDHW et dans un MGD. Pour que les comparaisons soient pertinentes, il est nécessaire de considérer un MGDHW et un MGD les mieux appris possibles quant à la précision. Pour chacun des α_{ok} , un MGD et un MGDHW sont sélectionnés parmi ceux appris à partir de la totalité des trajectoires générées, c'est-à-dire la base $\mathcal{D}_{K^{tot},1}^{\alpha_{ok}}$.

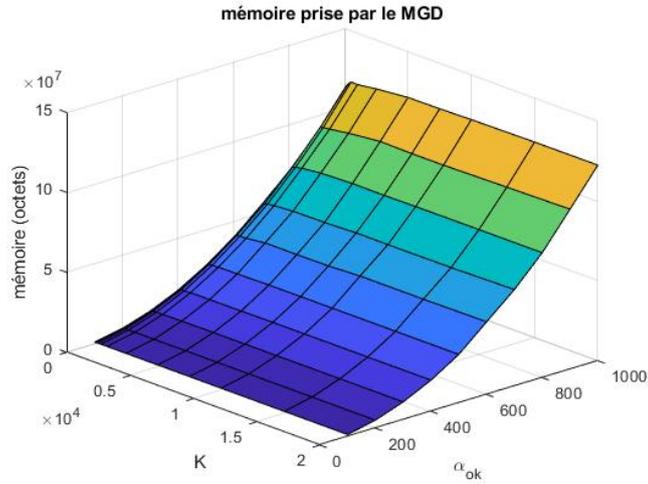


FIGURE 4.10 – Stockage MGD

Algorithm 8: Phase d'inférence sur $\text{MGDHW}_{K^{\text{tot}},1}^{\alpha_{ok}}$

Entrées :

1. $\text{MGDHW}_{K^{\text{tot}},1}^{\alpha_{ok}}$

Sorties : $\hat{R}_{\text{MGDHW}}^{(\alpha_{ok})}$ Inférence sur le $\text{MGDHW}_{K^{\text{tot}},1}^{\alpha_{ok}}$ (cf. section 3.2) ;

Temps de calcul de l'inférence ;

 $r(R^{(\alpha_{ok})}, \hat{R}_{\text{MGDHW}}^{(\alpha_{ok})})$;

4.4.1 Phase d'inférence

L'objectif étant de valider la pertinence de l'algorithme d'inférence développé dans la section 3.2, c'est le calcul d'inférence consistant en l'estimation de la fiabilité qui est choisi. En effet, la fiabilité théorique est calculable avec précision dans le cas d'un système à trois états de fonctionnement, ce qui permettra de comparer les estimations avec la valeur réelle.

L'algorithme d'inférence développé dans cette thèse et présenté dans la section

Algorithm 9: Phase d'inférence sur $\text{MGD}_{K^{\text{tot}},1}^{\alpha_{ok}}$

Entrées :

1. $\text{MGD}_{K^{\text{tot}},1}^{\alpha_{ok}}$

Sorties : $\hat{R}_{\text{MGD}}^{(\alpha_{ok})}$ Inférence sur le $\text{MGD}_{K^{\text{tot}},1}^{\alpha_{ok}}$ (cf. section 2.2.5) ;

Temps de calcul de l'inférence ;

 $r(R^{(\alpha_{ok})}, \hat{R}_{\text{MGD}}^{(\alpha_{ok})})$;

3.2 a été utilisé pour calculer l'estimation de fiabilité dans le modèle du MGDHW.

L'estimation de la fiabilité en utilisant la modélisation par un MGD a été réalisée en utilisant l'algorithme d'inférence d'élimination de variables présenté dans la section 2.2.5. D'autres algorithmes auraient été plus performants, tel que celui développé par DONAT dans [DONAT 2009], mais l'objectif de la thèse était surtout de montrer que l'inférence est possible dans un modèle graphique de durée hybride, et l'algorithme d'élimination de variables était plus simple à mettre en oeuvre.

Les fiabilités calculées par les deux approches peuvent maintenant être comparées à la fiabilité théorique, et les durées des calculs sont retenues, pour être comparés entre elles ensuite. Ces éléments sont regroupés dans les algorithmes 8 et 9.

4.4.2 Comparaison des temps de traitement

La figure 4.11 présente l'évolution des temps de calcul d'inférence pour un (courbe marron) et pour un MGDHW (courbe bleue) en fonction de la valeur de α_{ok} , qui est proportionnel à la taille de F^{sys} du MGD.

Pour un "petit" α_{ok} : $\alpha_{ok} = 200$, ce qui correspond à $S_{max} \simeq 230$, les temps de calcul sont proches de 7 minutes (avec un processeur de 3,70 GHz).

Lorsque α_{ok} augmente, et donc que la taille de l'espace d'état de S augmente, le temps de calcul de l'inférence dans le MGDH est multiplié par 6, il atteint 50 minutes pour un $\alpha_{ok} = 700$, tandis que dans ce cas, l'inférence n'a pas pu être effectuée dans le MGD sur notre machine (64 Go de RAM). La dernière inférence effectuée dans le MGD, pour $\alpha_{ok} = 600$ *i. e.* $S_{max} = 680$ a duré plus de 8 heures.

La comparaison des temps de traitement des estimations de la fiabilité avec, d'une part, une modélisation par un MGD et, d'autre part, par une modélisation par un MGDHW (cf figure 4.11), met en valeur que la modélisation par un MGD n'est pas adaptée pour de grandes valeurs de α_{ok} , valeurs qui ont peu d'impact dans le cas de la modélisation par un MGDHW. Pour $\alpha_{ok} = 700$, *i. e.* une taille de F autour de 800 dans notre exemple, notre machine n'a pas assez de mémoire pour estimer la fiabilité avec une modélisation par un MGD.

Ce résultat tend donc à montrer que les MGDHW apparaissent comme une solution parfaitement adaptée dans ce cas.

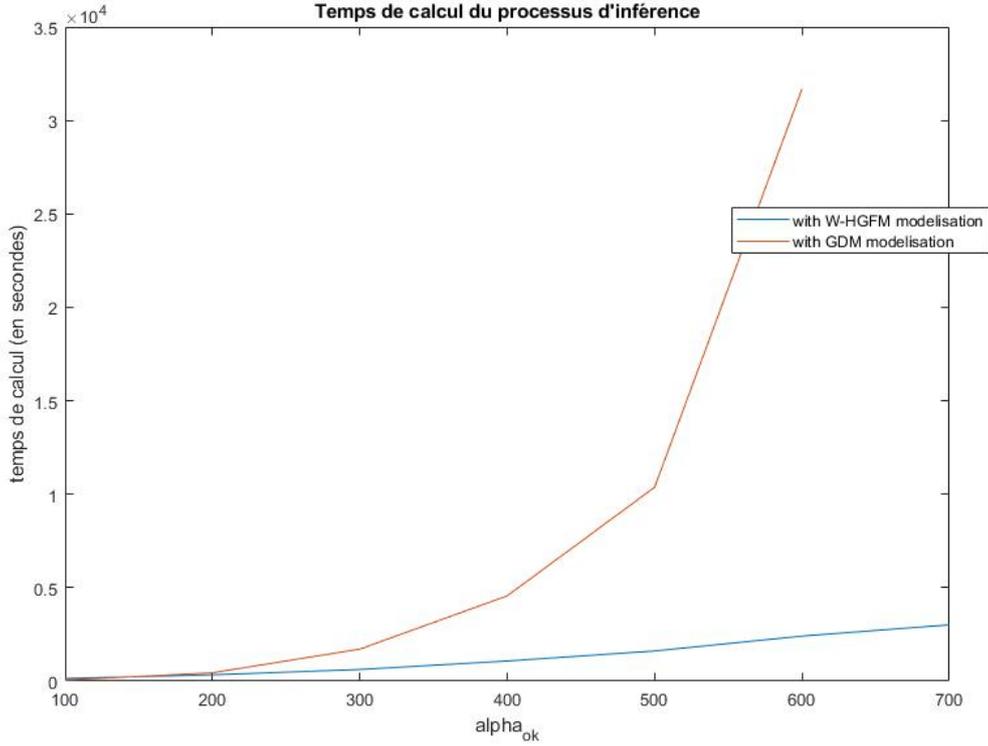


FIGURE 4.11 – Temps de traitement de l'estimation de la fiabilité

4.4.3 Calcul théorique de la fiabilité

L'algorithme 10 présente l'étape de calcul de la fiabilité théorique dans le cas d'un système qui comporte deux états de fonctionnement "ok" et "deg".

Algorithm 10: Calcul de la fiabilité théorique

Entrées : α_{ok} , β_{ok} , α_{deg} , β_{deg} , Q^{ini} , Q^{sys}

Sorties : R

$$R(t) \leftarrow Q^{ini}(ok)e^{-(\frac{t-1}{\alpha_{ok}})\beta_{ok}} + Q^{ini}(deg)e^{-(\frac{t-1}{\alpha_{deg}})\beta_{deg}} + Q^{ini}(ok)Q^{sys}(ok, deg) \sum_{j=1}^{t-1} (e^{-(\frac{j-1}{\alpha_{ok}})\beta_{ok}} - e^{-(\frac{j}{\alpha_{ok}})\beta_{ok}}) e^{-(\frac{t-1-j}{\alpha_{deg}})\beta_{deg}}$$

La suite de cette partie consiste en la démonstration de la formule présentée dans l'algorithme 10.

$$R(t) = P(X_t = ok) + P(X_t = deg) \quad (4.26)$$

La calcul de $P(X_t = ok)$ s'écrit comme suit :

$$\begin{aligned}
 P(X_t = ok) &= P(S_1 > t - 1, X_1 = ok) \\
 &= P(S_1 > t - 1 | X_1 = ok) P(X_1 = ok) \\
 &= (1 - F_{W_{\alpha_{ok}, \beta_{ok}}}(t - 1)) P(X_1 = ok) \\
 &= e^{-\left(\frac{t-1}{\alpha_{ok}}\right)^{\beta_{ok}}} Q^{ini}(ok)
 \end{aligned} \tag{4.27}$$

Développons $P(X_t = deg)$:

$$\begin{aligned}
 &P(X_t = deg) \\
 &= P(X_t = deg, X_1 = deg) \\
 &+ \sum_{j=1}^{t-2} P(X_1 = ok, j - 1 < S_1 \leq j, X_j = ok, S_j \leq 1, X_{j+1} = deg, S_{j+1} > t - j - 1, X_t = deg) \\
 &+ P(X_1 = ok, t - 2 < S_1 \leq t - 1, X_{t-1} = ok, S_{t-1} \leq 1, X_t = deg) \\
 &= P(S_1 > t - 1 | X_1 = deg) P(X_1 = deg) + \sum_{j=1}^{t-2} P(S_{j+1} > t - j - 1 | X_{j+1} = deg, S_j \leq 1) \\
 &\quad P(X_{j+1} = deg | X_j = ok, S_j \leq 1) P(j - 1 < S_1 \leq j | X_1 = ok) P(X_1 = ok) \\
 &+ P(X_t = deg | X_{t-1} = ok, S_{t-1} \leq 1) P(t - 2 < S_1 \leq t - 1 | X_1 = ok) P(X_1 = ok) \\
 &= e^{-\left(\frac{t-1}{\alpha_{deg}}\right)^{\beta_{deg}}} Q^{ini}(deg) + \sum_{j=1}^{t-2} e^{-\left(\frac{t-j-1}{\alpha_{deg}}\right)^{\beta_{deg}}} Q^{sys}(ok, deg) (e^{-\left(\frac{j-1}{\alpha_{ok}}\right)^{\beta_{ok}}} - e^{-\left(\frac{j}{\alpha_{ok}}\right)^{\beta_{ok}}}) Q^{ini}(ok) \\
 &+ Q^{sys}(ok, deg) (e^{-\left(\frac{t-2}{\alpha_{ok}}\right)^{\beta_{ok}}} - e^{-\left(\frac{t-1}{\alpha_{ok}}\right)^{\beta_{ok}}}) Q^{ini}(ok) \\
 &= Q^{ini}(deg) e^{-\left(\frac{t-1}{\alpha_{deg}}\right)^{\beta_{deg}}} + Q^{ini}(ok) Q^{sys}(ok, deg) \sum_{j=1}^{t-1} e^{-\left(\frac{t-j-1}{\alpha_{deg}}\right)^{\beta_{deg}}} (e^{-\left(\frac{j-1}{\alpha_{ok}}\right)^{\beta_{ok}}} - e^{-\left(\frac{j}{\alpha_{ok}}\right)^{\beta_{ok}}}) \tag{4.28}
 \end{aligned}$$

Les résultats 4.27 et 4.28 sont replacés dans la formule 4.26 pour donner l'expression de la fiabilité théorique du système décrit en 4.1.1.

$$\begin{aligned}
 R(t) &= Q^{ini}(ok) e^{-\left(\frac{t-1}{\alpha_{ok}}\right)^{\beta_{ok}}} + Q^{ini}(deg) e^{-\left(\frac{t-1}{\alpha_{deg}}\right)^{\beta_{deg}}} \\
 &+ Q^{ini}(ok) Q^{sys}(ok, deg) \sum_{j=1}^{t-1} e^{-\left(\frac{t-j-1}{\alpha_{deg}}\right)^{\beta_{deg}}} (e^{-\left(\frac{j-1}{\alpha_{ok}}\right)^{\beta_{ok}}} - e^{-\left(\frac{j}{\alpha_{ok}}\right)^{\beta_{ok}}}) \tag{4.29}
 \end{aligned}$$

4.4.4 Précision

À titre d'exemple, la figure 4.12 représente la courbe de fiabilité estimée avec le MGDHW, celle estimée avec le MGD, et la fiabilité théorique pour $\alpha_{ok} = 600$. Les estimations de la fiabilité sont très proches de la fiabilité théorique.

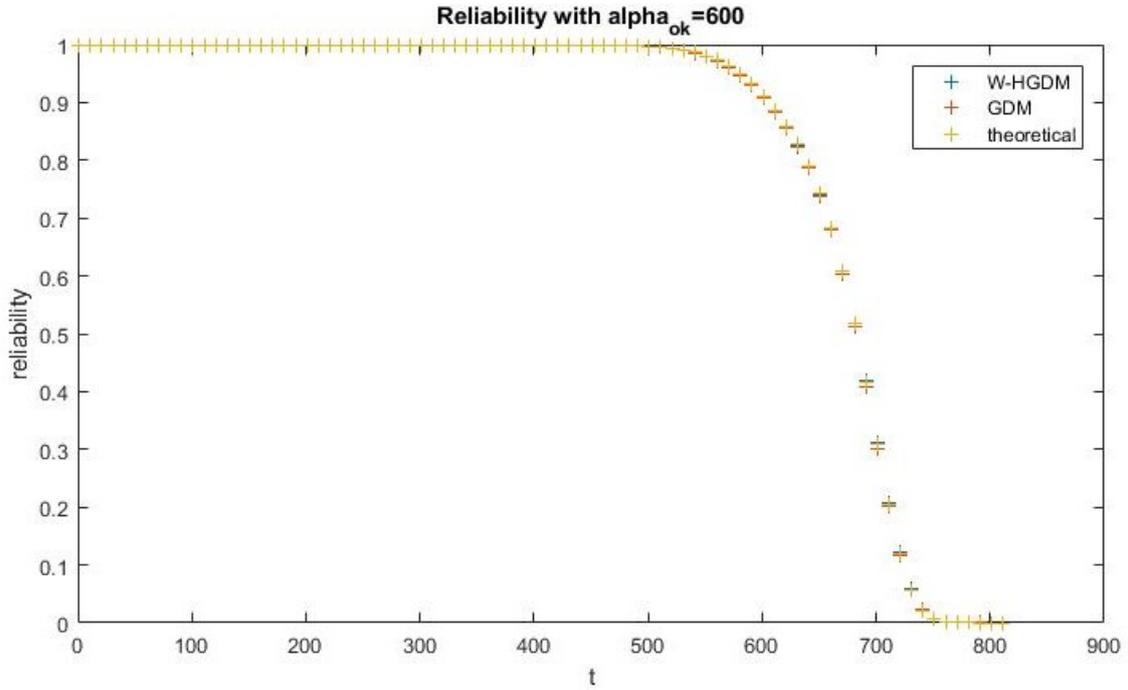


FIGURE 4.12 – Fiabilité par rapport aux modélisations

Pour évaluer la qualité de l'inférence, il est possible de comparer l'estimation \hat{R} de la fiabilité à la valeur théorique donnée par la formule 4.29 en utilisant le coefficient de corrélation (formule 4.30).

$$r = \frac{\sum_t (R(t) - \bar{R})(\hat{R}(t) - \bar{\hat{R}})}{\sqrt{(\sum_t R(t) - \bar{R})^2 \sum_t (\hat{R}(t) - \bar{\hat{R}})^2}} \quad (4.30)$$

La table 4.11 présente les valeurs des coefficients de corrélation entre $R^{(\alpha_{ok})}$ et $\hat{R}_{MGD}^{(\alpha_{ok})}$ et entre $R^{(\alpha_{ok})}$ et $\hat{R}_{MGDHW}^{(\alpha_{ok})}$ en fonction de la valeur de α_{ok} .

Le coefficient de corrélation est très proche de 1 dans tous les cas. Il est de plus en plus faible pour le MGD lorsque α_{ok} augmente, ce qui vient du fait que l'apprentissage de la table F_{sys} était de moins en moins bonne qualité lorsque α_{ok} augmente.

L'estimation de la fiabilité via le MGDHW est ainsi tout aussi satisfaisante que celle estimée avec le MGD quand elles sont toutes deux comparées par rapport à la fiabilité théorique. Toutefois, elle a nécessité un temps de calcul environ dix fois moins important.

TABLE 4.11 – Coefficients de corrélation de la fiabilité estimée avec le MGD et avec le MGDHW

r		modèle	
		MGD	MGDHW
α_{ok}	100	0.99999	0.99999
	200	0.99999	0.99999
	300	0.99998	0.99999
	400	0.99998	0.99999
	500	0.99997	0.99999
	600	0.99995	0.99999
	700	0.99995	0.99999
	800	/	0.99999
	900	/	0.99999
	1000	/	0.99999

4.5 Conclusion

Le protocole mis en place a ainsi permis de mettre en évidence que le modèle graphique de durée hybride Weibull original présenté au chapitre 3 permet de dépasser les limites imposées par l'approche standard du MGD (en terme de capacité de stockage et de temps de calcul). Le MGDHW a en effet une complexité en mémoire constante tandis que le MGD a une complexité en mémoire quadratique. Le calcul d'inférence est également plus performant avec un MGDHW qu'avec un MGD : le temps de calcul explose dans le cadre du MGD lorsque la taille de F^{sys} augmente, alors que pour des paramètres de Weibull équivalents, l'inférence est toujours calculable avec un MGDHW.

La comparaison des résultats de l'algorithme mis en place pour le MGDHW ont toutefois été effectuées avec l'algorithme d'élimination des variables dans le MGD, étant lui-même basé sur cette méthode. Il serait intéressant de comparer les résultats obtenus avec une modélisation par un MGDHW avec les résultats obtenus avec un MGD en utilisant l'algorithme développé par DONAT dans [DONAT 2009].

Il n'a également pas été testé ici l'intégration du MGDHW dans un réseau plus complexe, tel que le modèle VirMaLab, présenté dans la section 2.1.4, pour évaluer réellement l'algorithme d'inférence.

Conclusions générales et perspectives

Conclusions

Dans cette thèse, nous nous sommes intéressés à l'utilisation des réseaux bayésiens dynamiques dans le cadre de la modélisation de la dégradation de systèmes à espaces d'états discret. Des travaux précédents [FOULLIARON 2015], ayant utilisé pour du pronostic la méthode classique, basée sur les modèles graphiques de durée [DONAT 2009], en ont montré les limites, en matière de de capacité de stockage et de temps de calcul.

Nous nous sommes donc intéressés à la mise en place d'un modèle original s'appuyant sur les modèles graphiques de durée, et permettant à la variable des temps de séjour d'être continue, et plus spécifiquement de suivre une loi de Weibull. Il a été choisi de s'intéresser à ce type de loi en raison de sa pertinence démontrée en analyse de fiabilité, par exemple dans le cas de systèmes dans lesquels la défaillance d'un seul composant suffit à provoquer la panne de tout le système.

Un nouveau modèle hybride, que nous avons appelé modèle graphique de durée hybride Weibull, a donc été proposé dans cette thèse, ainsi qu'un algorithme d'inférence spécifique pour effectuer des calculs de probabilités dans ce réseau hybride.

Cet algorithme élimine successivement les variables discrètes et les variables continues, par une sommation ou une intégration. La loi de Weibull n'intervient finalement que dans le calcul d'intégration, sous la forme de sa fonction de répartition. Ainsi, l'algorithme proposé pourrait fonctionner avec n'importe quelle distribution, du moment que la fonction de répartition de la loi choisie peut être évaluée analytiquement, que le nombre de paramètres est faible, et que son apprentissage est "aisé".

La procédure d'inférence et la pertinence du MGDHW par rapport au MGD ont été validées en s'appuyant sur un protocole mis en place. Ne disposant pas de données réelles, la première étape a été de constituer des bases de données, permettant d'obtenir différentes tailles d'espaces d'état de la variable de temps de séjour dans le MGD. Les comparaisons des apprentissages et des estimations de fiabilités calculées

avec les algorithmes d'inférence ont mis en évidence que le MGD requiert un nombre beaucoup plus important que le MGDHW de séquences d'états dans la base d'apprentissage pour apprendre correctement les paramètres décrivant la dynamique de dégradation du système étudié. Si les estimations de fiabilité via le MGDHW ou via le MGD sont très proches de la fiabilité théorique dans les deux cas, le calcul de fiabilité via l'inférence entraîne des temps de calculs beaucoup plus importants dans le cas du MGD. Par ailleurs, l'inférence n'est même plus possible pour les MGD dans le cas d'espaces d'état du temps de séjour de cardinal élevé. Au contraire, pour les mêmes profondeurs de la variable de temps de séjour, l'inférence demeure possible pour le MGDHW. Les résultats de la section 4.4 ont montré que si le temps de calcul augmente de manière très significative au sein du MGD (jusqu'à atteindre la saturation des machines), il reste tout à fait raisonnable pour le MGDHW (en dessous d'une heure). Pour toutes ces raisons, les MGDHW apparaissent comme une solution intéressante et pertinente dans le cas de variables de temps de séjour très élevée pour lesquelles une discrétisation (nécessaire avec un MGD) peut conduire à une explosion de la représentation spatiale. Par ailleurs, la généralisation à d'autres lois "bien adaptées" des travaux proposés dans cette thèse ouvre la porte à de nouveaux sujets de recherche.

Perspectives

Ainsi, puisqu'il a été montré que l'approche n'est pas forcément limitée aux seules lois de Weibull, malgré le choix initial, une première perspective serait d'utiliser d'autres distributions au sein d'un modèle graphique de durée hybride en déployant un même type d'inférence que celui développé dans cette thèse. La liste des "contraintes" imposées aux lois candidates pour être intégrées à un modèle graphique de durée hybride ouvre tout de même le champ des perspectives à de nombreuses lois classiques.

Une autre perspective est d'étendre ces travaux à ceux de FOULLIARON sur le pronostic avec un MGD. Dans le modèle développé, appelé modèle graphique de durée conditionnelle (MGDC), les distributions des temps de séjour des différents états ne sont plus nécessairement indépendantes, comme c'était le cas dans un simple modèle graphique de durée. Dans un MGDC, le temps de séjour dépend de la dynamique, du profil du système. Les temps de séjour sont ainsi caractérisés par des mélanges de Weibull. S'ils ont montré de bons comportements dans l'estimation de la durée de vie résiduelle, ces travaux se sont trouvés limités par le caractère discret de la variable "temps de séjour", qui conduisait à des complexités calculatoires

trop élevées. Ainsi, étendre les travaux de cette thèse au MGDC permettrait d'augmenter les capacités de pronostic, nécessaires à l'établissement d'une maintenance prévisionnelle.

Dans cette approche, il faudrait tout d'abord s'intéresser à l'intégration des travaux de cette thèse au sein d'architectures de réseaux bayésiens dynamiques plus complexes que le simple modèle graphique de durée utilisé ici, telles que le modèle VirMaLab, qui a été présenté dans la section 2.1.4.

Bibliographie

- ABDEL-HAMEED, Mohamed (1975), « A Gamma Wear Process », *in* : *IEEE Transactions on Reliability* R-24.2, p. 152–153, ISSN : 0018-9529.
- ARULAMPALAM, M. Sanjeev, Simon MASKELL et Neil GORDON (2002), « A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking », *in* : *IEEE Transactions on signal processing* 50, p. 174–188.
- AVEN, Terje et Uwe JENSEN (1999), *Stochastic Models in Reliability*, Applications of mathematics, Springer, ISBN : 9780387986333.
- BACH, Francis R. et Michael I. JORDAN (2005), « Discriminative training of hidden Markov models for multiple pitch tracking, , 2005 », *in* : *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, t. 5, p. 489–492.
- BARBU, Vlad et Nikolaos LIMNIOS (2004), « Discrete Time Semi-Markov Processes for Reliability and Survival Analysis — A Nonparametric Estimation Approach », *in* : *Parametric and Semiparametric Models with Applications to Reliability, Survival Analysis, and Quality of Life*, sous la dir. de N. BALAKRISHNAN, M. S. NIKULIN, M. MESBAH et N. LIMNIOS, Boston, MA : Birkhäuser Boston, p. 487–502, ISBN : 978-0-8176-8206-4.
- BAUCKHAGE, Christian (2013), « Computing the Kullback-Leibler Divergence between two Weibull Distributions », *in* : *arXiv preprint arXiv :1310.3713*.
- BECKER, Anne, Dan GEIGER et Alejandro A. SCHÄFFER (1998), « Automatic selection of loop breakers for genetic linkage analysis. », *in* : *Human heredity* 48 1, p. 49–60.
- BELLOT, David (2002), « Data Fusion with Bayesian network for dynamical systems modelling and an application to telemedicine », Theses, Université Henri Poincaré - Nancy I.
- BEN MRAD, Ali (2015), « Probabilistic evidence in bayesian networks », Theses, Université de Valenciennes et du Hainaut-Cambresis.
- BEN SALEM, Abdeljabbar, Alexandre MULLER et Philippe WEBER (2006), *Dynamic Bayesian Networks in System Reliability Analysis*.
- BERTHOLON, Henri (2001), « Une modélisation du vieillissement », thèse de doct., Université Joseph Fourier, Grenoble.

- BOUDALI, Hichem et Joanne Bechta DUGAN (2005), « A discrete-time Bayesian network reliability modeling and analysis framework », *in* : *Reliability Engineering and System Safety* 87, p. 337–349.
- BOUEJLA, Amal (2014), « Contribution of Bayesian networks to the prevention of the risk of piracy against Oil Offshore Fields », Theses, Ecole Nationale Supérieure des Mines de Paris.
- BOUILLAUT, Laurent, Patrice AKNIN, Roland DONAT et Sandrine BONDEUX (2011), « VirMaLab - A generic approach for optimizing maintenance policies of complex systems », *in* : *WCRR 2011 - 9th World Congress on Railway Research*, WCRR 2011 - 9th World Congress on Railway Research, Lille, France, 22-/05/2011 - 26/05/2011, Lille, France : WCRR, 11p.
- BRACQUEMOND, Cyril et Olivier GAUDOIN (2003), « A survey on discrete lifetime distributions », *in* : 10.01, p. 69–98, eprint : <https://doi.org/10.1142/S0218539303001007>.
- BROY, Perrine, Hassane CHARAIBI et Roland DONAT (2011), « Using Dynamic Bayesian Networks to solve a dynamic reliability problem », *in* : *Advances in Safety, Reliability and Risk Management – Bérenguer, Grall and Guedes Soares (eds)*, 335–341.
- CARROLL, Kevin J. (2003), « On the use and utility of the Weibull model in the analysis of survival data », *in* : *Controlled Clinical Trials* 24.6, p. 682–701, ISSN : 0197-2456.
- COBB, Barry R. et Prakash P. SHENOY (2006), « Inference in hybrid Bayesian networks with mixtures of truncated exponentials », *in* : *International Journal of Approximate Reasoning* 41.3, p. 257–286, ISSN : 0888-613X.
- COCOZZA-THIVENT, Christiane (1997), *Processus stochastiques et fiabilité des systèmes*, Mathématiques et Applications, Springer Berlin Heidelberg.
- COOPER, Gregory F. (1990), « The computational complexity of probabilistic inference using bayesian belief networks », *in* : *Artificial Intelligence* 42.2, p. 393–405, ISSN : 0004-3702.
- COWELL, Robert G., Alexander DAWID, Steffen LAURITZEN et David SPIEGELHALTER (1999), *Probabilistic Networks and Expert Systems*.
- COX, David (1972), « Regression Models and Life Tables », *in* : *Journal of the Royal Statistic Society B.34*, p. 187–202.
- DECHTER, Rina (1999), « Bucket elimination : A unifying framework for reasoning », *in* : *Artificial Intelligence* 113.1, p. 41–85, ISSN : 0004-3702.

- DEMPSTER, Arthur, Natalie LAIRD et Donald B. RUBIN (1977), « Maximum Likelihood From Incomplete Data Via The EM algorithm », *in* : *Journal of the Royal Statistical Society. Series B (Methodological)* 39, p. 1–38.
- DEVIREN, Murat, Khalid DAOUDI et Kamel SMAÏLI (2004), « Une nouvelle approche de modélisation du langage par des réseaux Bayésiens dynamiques », *in* : *XXVes Journées d'Etudes sur la Parole - JEP-TALN-RECITAL 2004*, Colloque avec actes et comité de lecture. internationale., Fès, Maroc.
- DONAT, Roland (2009), « Reliability and Maintenance Modelling based on Probabilistic Graphical Models - Case study on Rail Break Prevention », Theses, INSA de Rouen.
- DONAT, Roland, Philippe LERAY, Laurent BOUILLAUT et Patrice AKNIN (2010), « A Dynamic Bayesian Network to Represent Discrete Duration Models », *in* : *Neurocomputing* 73.4-6, p. 570–577, ISSN : 0925-2312.
- FISHER, Ronald A. (1922), « On the Mathematical Foundations of Theoretical Statistics », *in* : *Phil. Trans. Roy. Soc. Lond.* A222, p. 309–368.
- FOULLIARON, Josquin (2015), « Utilisation des modèles graphiques probabilistes pour la mise en place d'une politique de maintenance à base de pronostic », Theses, Université Paris-Est.
- FRANCIS, Galton (1879), « The geometric mean, in vital and social statistics », *in* : *Proceedings of the Royal Society*, 365–367.
- FRANÇOIS, Olivier (2006), « De l'identification de structure de réseaux bayésiens à la reconnaissance de formes à partir d'informations complètes ou incomplètes », thèse de doct., Institut National des Sciences Appliquées de Rouen.
- GILKS, W.R., S. RICHARDSON et D. SPIEGELHALTER (1995), *Markov Chain Monte Carlo in Practice*, Chapman & Hall/CRC Interdisciplinary Statistics, Taylor & Francis, ISBN : 9780412055515.
- GROSS, Jonathan L., Jay YELLEN et Ping ZHANG (2004), *Handbook of Graph Theory, Second Edition*, 2nd, Chapman & Hall/CRC.
- HASTINGS, Wilfred K. (1970), « Monte Carlo sampling methods using Markov chains and their applications », *in* : *Biometrika* 57.1, p. 97–109, eprint : <http://biomet.oxfordjournals.org/cgi/reprint/57/1/97.pdf>.
- HECKERMAN, David (1995), *A Tutorial on Learning With Bayesian Networks*, rapp. tech., p. 57.
- HENRION, Max (1988), « Propagating Uncertainty in Bayesian Networks by Probabilistic Logic Sampling », *in* : *Uncertainty in Artificial Intelligence*, sous la dir. de John F. LEMMER et Laveen N. KANAL, t. 5, Machine Intelligence and Pattern Recognition, North-Holland, p. 149–163.

- HENRY (1842), *Catastrophe ferroviaire de Meudon*, <http://www.amaepf.fr/recits-polaires/316-15-durville-necro>.
- JAAKKOLA, Tommi S. et Michael I. JORDAN (1999), « Variational Probabilistic Inference and the QMR-DT Network », *in* : *J. Artif. Int. Res.* 10.1, p. 291–322, ISSN : 1076-9757.
- JENSEN, Finn V. (1996), *Introduction to Bayesian Networks*, 1st, Secaucus, NJ, USA : Springer-Verlag New York, Inc., ISBN : 0387915028.
- JIANG, Shao-Fei, Chun FU et Chunming ZHANG (2011), « A Hybrid Data-fusion System Using Modal Data and Probabilistic Neural Network for Damage Detection », *in* : *Adv. Eng. Softw.* 42.6, p. 368–374, ISSN : 0965-9978.
- JONQUIÈRES, Stéphanie (2000), « Application des réseaux Bayésiens à la reconnaissance active d’objets 3D : contribution à la saisie », Thèse de doctorat dirigée par Devy, Michel Informatique Toulouse, INPT 2000, thèse de doct., 170 p.
- JORDAN, Michael I., Zoubin GHAMRANI et et AL. (1999), « An introduction to variational methods for graphical models », *in* : *Machine learning*, MIT Press, p. 183–233.
- KADIE, Carl M., David HOVEL et Eric HORVITZ (2001), *MSBNx : A Component-Centric Toolkit for Modeling and Inference with bayesian Networks*, Microsoft Research Technical Report MSR-TR-2001-67.
- KAPUR, Kailash C. et Leonard R. LAMBERSON (1977), *Reliability in engineering design*, English, Wiley New York, xvi, 586 p. : ISBN : 0471511919.
- KAY, Richard (1977), « Proportional Hazard Regression Models and the Analysis of Censored Survival Data », *in* : *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 26.3, p. 227–237, ISSN : 00359254, 14679876.
- KOLLER, Daphne, Uri LERNER et Dragomir ANGELOV (1999), « A General Algorithm for Approximate Inference and Its Application to Hybrid Bayes Nets », *in* : UAI’99, p. 324–333.
- KRISTENSEN, Kristian et Ilse A. RASMUSSEN (2002), « The use of a Bayesian network in the design of a decision support system for growing malting barley without use of pesticides », *in* : *Computers and Electronics in Agriculture* 33.3, p. 197–217, ISSN : 0168-1699.
- KULLBACK, S. (1952), « An Application of Information Theory to Multivariate Analysis », *in* : *Ann. Math. Statist.* 23.1, p. 88–102.
- LANTERNIER, Brice, Dominique CHARPENTIER et Patrick LYONNET (2006), « Modélisation de taux de défaillance en mécanique », *in* : *Colloque de maîtrise des risques et sûreté de fonctionnement "Risques et performances"*, Lille, France, p. 4.

- LAPLACE, Pierre-Simo (1836), *Exposition du système du monde*, Exposition du système du monde, Bachelier.
- LAURITZEN, Steffen L. (1992), « Propagation of Probabilities, Means and Variances in Mixed Graphical Association Models », *in* : *Journal of the American Statistical Association* 87, p. 1098–1108.
- LAURITZEN, Steffen L. et David J. SPIEGELHALTER (1988), « Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems », *in* : *Journal of the Royal Statistical Society. Series B (Methodological)* 50.2, p. 157–224, ISSN : 00359246.
- LAWLESS, Jerald F. (2011), *Statistical Models and Methods for Lifetime Data*, Wiley Series in Probability and Statistics, Wiley, ISBN : 9781118031254.
- LIMNIOS, Nikolaos et Ghorghé OPRISAN (2001), *Semi-Markov Processes and Reliability*.
- MARKOV, Andreï (1971), « Extension of the Limit Theorems of Probability Theory to a Sum of Variables Connected in a Chain », *in* : *Dynamic Probabilistic Systems (Volume I : Markov Models)*, sous la dir. de R. HOWARD, New York City : John Wiley & Sons, Inc., chap. Appendix B, p. 552–577.
- MCCULLOCH, Warren S. et Walter PITTS (1943), « A logical calculus of the ideas immanent in nervous activity », *in* : *The bulletin of mathematical biophysics* 5.4, p. 115–133, ISSN : 1522-9602.
- MORAL, Serafín, Rafael RUMÍ et Antonio SALMERÓN (2001), « Mixtures of Truncated Exponentials in Hybrid Bayesian Networks », *in* : *Proceedings of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, ECSQARU '01, London, UK, UK : Springer-Verlag, p. 156–167, ISBN : 3-540-42464-4.
- MURPHY, Kevin P. (1999), « A Variational Approximation for Bayesian Networks with Discrete and Continuous Latent Variables », *in* : *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, UAI'99, Stockholm, Sweden : Morgan Kaufmann Publishers Inc., p. 457–466, ISBN : 1-55860-614-9.
- MURPHY, Kevin Patrick (2002), « Dynamic Bayesian Networks : Representation, Inference and Learning », thèse de doct., University of California, Berkeley.
- NATKIN, Stéphane O. (1980), « Les réseaux de Petri Stochastiques et leur applications à l'évaluation des systèmes informatiques », thèse de doct., CNAM-Paris.
- NAÏM, Patrick, Pierre-Henri WUILLEMIN, Philippe LERAY, Olivier POURRET et Anna BECKER (2007), *Réseaux Bayésiens*.

- NEAL, Radford et Geoffrey HINTON (1998), « A View Of The Em Algorithm That Justifies Incremental, Sparse, And Other Variants », *in* : *Learning in graphical models* 89.
- NEAPOLITAN, Richard E. (2003), *Learning Bayesian Networks*, Upper Saddle River, NJ, USA : Prentice-Hall, Inc., ISBN : 0130125342.
- NELSON, Wayne (2004), *Applied Life Data Analysis*, Wiley Series in Probability and Statistics, Wiley, ISBN : 9780471644620.
- NOWLAN, F. Stanley et Howard F. HEAP (1978), *Reliability-centered Maintenance*, Dolby Access Press.
- O’GORMAN, Bryan, Ryan BABBUSH, Alejandro PERDOMO-ORTIZ, Alan ASPURUGUZIK et Vadim SMELYANSKIY (2015), « Bayesian network structure learning using quantum annealing », *in* : *The European Physical Journal Special Topics* 224.1, p. 163–188, ISSN : 1951-6401.
- PAGÈS, Alain et Michel GONDRAN (1980), *Fiabilité des systèmes*, Collection de la Direction des études et recherches d’Electricité de France, Eyrolles, ISBN : 9782212015829.
- PEARL, Judea (1985), *Bayesian Networks : A Model of Self-activated Memory for Evidential Reasoning*, Report, UCLA, Computer Science Department.
- PEREZ, Sandra (2010), « Modélisation des pathologies respiratoires des écoliers niçois par les Réseaux Bayésiens », *in* : *First edition of International Colloquium Health and Space*, Frédéric Audard, Aix-Marseille University, Marion Borderon, Aix-Marseille University, Vincent Laperrière, Aix-Marseille University, Sébastien Oliveau, Aix-Marseille University, Sandra Perez, University of Nice Sophia Antipolis, Marseille, France.
- PETRI, Carl Adam (1962), « Kommunikation mit Automaten », thèse de doct., Darmstadt University of Technology, Germany.
- PHAM, Hoang (2006), « System Reliability Concepts », *in* : *System Software Reliability*, London : Springer London, p. 9–75.
- PINAUD, Bruno, Cédric BAUDRIT, Mariette SICARD, Pierre-Henri WUILLEMIN et Nathalie PERROT (2008), « Validation et enrichissement interactifs d’un apprentissage automatique des paramètres d’un réseau bayésien dynamique appliqué aux procédés alimentaires », *in* : *JFRB 2008 - 4èmes Journées Francophones sur les Réseaux Bayésiens*, document de 12 pages en pdf, Lyon, France.
- POISSON, Siméon-Denis (1837), *Recherches sur la probabilité des jugements en matière criminelle et en matière civile : précédées des règles générales du calcul des probabilités*, Bachelier.

- POLLINO, Carmel A., Owen WOODBERRY, Ann NICHOLSON, Kevin KORB et Barry T. HART (2007), « Parameterisation and evaluation of a Bayesian network for use in an ecological risk assessment », *in* : *Environmental Modelling & Software* 22.8, Bayesian networks in water resource modelling and management, p. 1140–1152, ISSN : 1364-8152.
- RAO, Malempati M. (1984), *Probability Theory with Applications*, Probability and mathematical statistics, Elsevier Science, ISBN : 9780080925363.
- RAUSAND, Marvin et Arnljot HØYLAND (2004), *System Reliability Theory : Models, Statistical Methods, and Applications*, Wiley Series in Probability and Statistics, Wiley, ISBN : 9780471471332.
- RINGLER, J. (1982), « Une modélisation bayésienne du taux de défaillance en fiabilité », *in* : *Microelectronics Reliability* 22.3, p. 385–392, ISSN : 0026-2714.
- ROBERT, Christian P. (1994), *The Bayesian Choice : A Decision-Theoretic Motivation*, Springer Texts in Statistics, Springer New York, ISBN : 9781475743142.
- ROSS, Sheldon M. (1996), *Stochastic processes*, Wiley series in probability and statistics : Probability and statistics, Wiley, ISBN : 9780471120629.
- RUTHERFORD, Ernest et Frederick SODDY (1903), « LX. Radioactive Change », *in* : *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 5.29, p. 576–591.
- SAHAMI, Mehran, Susan DUMAIS, David HECKERMAN et Eric HORVITZ (1998), « A Bayesian Approach to Filtering Junk E-Mail », *in* : *Learning for Text Categorization : Papers from the 1998 Workshop*, Madison, Wisconsin : AAAI Technical Report WS-98-05.
- SCANAGATTA, Mauro, Cassio P de CAMPOS, Giorgio CORANI et Marco ZAFFALON (2015), « Learning Bayesian Networks with Thousands of Variables », *in* : *Advances in Neural Information Processing Systems 28*, sous la dir. de C. CORTES, N. D. LAWRENCE, D. D. LEE, M. SUGIYAMA et R. GARNETT, Curran Associates, Inc., p. 1864–1872.
- SINGPUREWALLA, Nozer D. et Mao SHI SONG (1988), « Reliability analysis using Weibull lifetime data and expert opinion », *in* : 37, p. 340–347.
- VILLEMEUR, Alain (1988), *Sûreté de fonctionnement des systèmes industriels : fiabilité, facteurs humains, informatisation*, Collection de la Direction des études et recherches d'Électricité de France, Eyrolles, ISBN : 9782212016154.
- WATSON, H. A. (1961), « Method of inadvertent launch control analysis », *in* : *Launch Control Safety Study*, Bell Telephone Laboratories, Section VII, Vol. 1, Murray Hill, New Jersey USA.

- WEBER, Philippe et Lionel JOUFFE (2003), « Reliability modelling with dynamic bayesian networks », *in* : IFAC, p. 57–62.
- WEBER, Philippe, Paul MUNTEANU et Lionel JOUFFE (2004), « Dynamic Bayesian Networks Modelling the Dependability of Systems with Degradations and Exogenous Constraints », *in* : *IFAC Proceedings Volumes 37.4*, 11th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2004), Salvador, Brazil, 5-7 April 2004, p. 207 –212, ISSN : 1474-6670.
- WEIBULL, Waloddi (1951), « A statistical distribution function of wide applicability », *in* : *Journal of applied mechanics* 18.3, p. 293–297.
- WIENER, Norbert (1923), *Differential-space*.
- YCARD, Bernard (2004), *Notions de fiabilité*, Cahier de Mathématiques Appliquées n° 13.
- ZAHRA, Fatma Zohra, Saliha KHOUAS-OUKID et Yasmina ASSOUL-SEMMAR (2013), « Dynamic Bayesian Networks in Dynamic Reliability and Proposition of a Generic Method for Dynamic Reliability Estimation », *in* : *Modeling Approaches and Algorithms for Advanced Computer Applications*, sous la dir. d'Abdelmalek AMINE, Ait Mohamed OTMANE et Ladjel BELLATRECHE, Cham : Springer International Publishing, p. 409–418, ISBN : 978-3-319-00560-7.
- ZIO, Enrico (2007), *An Introduction to the Basics of Reliability and Risk Analysis*, Series on quality, reliability & engineering statistics, World Scientific, ISBN : 9789812707413.
- ZWINGELSTEIN, Gilles (1996), *La maintenance basée sur la fiabilité : guide pratique d'application de la RCM*, Collection Diagnostic et maintenance, Hermès, ISBN : 9782866015459.

Liste des figures

1.1	Catastrophe ferroviaire de Meudon, [HENRY 1842]	5
1.2	Courbe en baignoire	8
1.3	Politiques de maintenance	9
2.1	Approche générique pour développer des modèles de maintenance . .	18
2.2	Approche VirMaLab	19
2.3	Exemple de réseau bayésien	21
2.4	Réseau Bayésien Dynamique à trois tranches de temps	22
2.5	Réseau Bayésien Dynamique	23
2.6	Modèle graphique de durée	26
2.7	Modèle graphique de durée	26
2.8	Modèle graphique de durée	32
2.9	Réseau bayésien	34
2.10	Élimination de la variable X_4	34
2.11	Élimination de la variable X_2	35
2.12	Élimination de la variable X_1	35
2.13	Explosion du temps de calcul d'inférence dans le MGD	38
3.1	Modèle graphique de durée Hybride-Weibull	40
3.2	Modèle graphique de durée Hybride-Weibull	43
4.1	Exemple de trajectoire	51
4.2	Modèle graphique de durée Hybride-Weibull	52
4.3	Protocole expérimental	57
4.4	Exemple de trajectoires	63
4.5	Évolution de la divergence de Kullback-Leibler entre les distributions de Weibull associées à l'état "ok" apprise et théorique, en fonction du nombre de trajectoires apprises et du paramètre α_{ok}	71
4.6	Évolution de la divergence de Kullback-Leibler entre les tables de probabilité conditionnelle de la durée de séjour dans l'état "ok" apprise et théorique en fonction du nombre de trajectoires apprises et du paramètre α_{ok}	71
4.7	Temps de calcul apprentissage MGDHW	73

LISTE DES FIGURES

4.8	Temps de calcul apprentissage MGD	74
4.9	Stockage MGDHW	74
4.10	Stockage MGD	75
4.11	Temps de traitement de l'estimation de la fiabilité	77
4.12	Fiabilité par rapport aux modélisations	79

Liste des tables

2.1	Exemple de table de probabilité conditionnelle de capteur	31
3.1	Table de probabilité conditionnelle de transition naturelle d'un système à trois états de fonctionnement sans réparation	42
4.1	Paramètres de la Weibull caractérisant la LPC des temps de séjour après transition	56
4.2	LPC de l'état initial	57
4.3	LPC de transition naturelle	57
4.4	Quatre trajectoires simulées du système de production. Dans ce contexte le nombre de trajectoires K est égal à 4	63
4.5	Nombre d'observations des temps de séjour allant de 1 à S_{max} trimestres pour chaque état (excepté l'état absorbant panne) dans les $K = 2000$ trajectoires disponibles.	65
4.6	Nombre d'observations des transitions entre les états	65
4.7	Estimation de la loi de probabilité conditionnelle de transition naturelle du système	66
4.8	Estimation de la loi de probabilité des temps de séjour s'il y a une transition	66
4.9	Estimation de $P(S \geq s)$	68
4.10	Paramètre de Weibull et test d'adéquation	68
4.11	Coefficients de corrélation de la fiabilité estimée avec le MGD et avec le MGDHW	80

Liste des algorithmes

1	Élimination de variable	36
2	Protocole expérimental	58
3	Création de $\mathcal{D}^{\alpha_{ok}}$: Simulation de K^{tot} trajectoires d'un système modélisé par un MGDHW.	59
4	Choix de la profondeur T	59
5	Simulation d'une trajectoire d'un système représenté par un MGDHW	62
6	Phase d'apprentissage : apprentissage du $MGDHW_{K,d}^{\alpha_{ok}}$ sur la base $\mathcal{D}_{K,d}^{\alpha_{ok}}$	64
7	Phase d'apprentissage : apprentissage du $MGD_{K,d}^{\alpha_{ok}}$ sur la base $\mathcal{D}_{K,d}^{\alpha_{ok}}$. .	64
8	Phase d'inférence sur $MGDHW_{K^{tot},1}^{\alpha_{ok}}$	75
9	Phase d'inférence sur $MGD_{K^{tot},1}^{\alpha_{ok}}$	75
10	Calcul de la fiabilité théorique	77