



HAL
open science

Diagnostic en ligne des systèmes à événements discrets complexes : approche mixte logique/probabiliste

Dang-Trinh Nguyen

► **To cite this version:**

Dang-Trinh Nguyen. Diagnostic en ligne des systèmes à événements discrets complexes : approche mixte logique/probabiliste. Automatique / Robotique. Université Grenoble Alpes, 2015. Français. NNT : 2015GREAT067 . tel-01227260

HAL Id: tel-01227260

<https://theses.hal.science/tel-01227260>

Submitted on 10 Nov 2015

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

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

Spécialité : **Automatique Productique**

Arrêté ministériel : 7 août 2006

Présentée par

Dang Trinh NGUYEN

Thèse dirigée par **Eric ZAMAÏ**

préparée au sein du **Laboratoire GSCOP**
dans l'**École Doctorale EEATS**

Diagnostic en ligne des systèmes à événements discrets complexes : approche mixte Logique/Probabiliste.

Thèse soutenue publiquement le **15 Octobre 2015**,
devant le jury composé de :

M. Jean-Marie FLAUS

Professeur, Université Joseph Fourier, Président

M. Pascal BERRUET

Professeur, Université de Bretagne-Sud, Rapporteur

M. Jean-François PETIN

Professeur, Université de Lorraine, Nancy, Rapporteur

M. Sébastien HENRY

Maître de conférences, Université Lyon 1 IUT, Villeurbanne, Examineur

M. Quoc-Bao DUONG

Ingénieur, A.I.P PRIMECA Dauphine Savoie, Grenoble-INP, Examineur

M. Muhammad Kashif SHAHZAD

Chercheur - Post-doc, G-SCOP, Grenoble-INP, Invité

M. Eric ZAMAÏ

Maître de conférences HDR, Grenoble-INP, Directeur de thèse



Avant-Propos

Le travail présenté dans ce mémoire a été préparé dans l'équipe Gestion et Conduite des Systèmes de Production (GCSP) au laboratoire G-SCOP, dans le cadre du programme Européen AREAS. Je tiens à remercier tout d'abord Messieurs Yanick Frein et François Villeneuve, directeurs successif du Laboratoire G-SCOP et Madame Christelle Grandvallet, responsable des projets européens et internationaux de m'avoir accueilli et permis d'effectuer mes travaux de recherche tout au long de trois années de ma thèse.

Je tiens à adresser mes remerciements à mon directeur de thèse, Monsieur Eric Zamaï, Maître de Conférences à l'Institut National Polytechnique de Grenoble, tout d'abord pour avoir accepté de m'encadrer et de diriger ces travaux de recherche, et ensuite pour sa disponibilité, ses conseils précieux durant ces trois années de thèse. J'ai eu une grande chance, d'avoir pu travailler avec Eric, qui m'a aidé à acquérir des compétences non seulement dans le domaine de la recherche, mais également dans ceux de la gestion et de l'organisation de mes travaux.

J'aimerais exprimer toute ma gratitude aux membres de mon jury. Tout d'abord, je voudrais remercier Monsieur Jean-Marie Flaus, Professeur à l'Université de Grenoble, pour avoir accepté de présider mon jury. Je remercie particulièrement Messieurs Pascal Berruet, Professeur à l'Université de Bretagne-Sud et Jean-François Pétin, Professeur à l'Université de Lorraine pour m'avoir fait le privilège d'être rapporteurs de ma thèse et surtout pour leurs critiques constructives. Je tiens également à remercier Monsieur Sébastien Henry, Maître de Conférences à l'Université Lyon 1 et Monsieur Quoc-Bao Duong, Ingénieur à l'A.I.P PRIMECA Dauphine-Savoie pour avoir accepté la charge d'être examinateurs de ces travaux. Je remercie également Monsieur Muhammad Kashif Shahzad, Chercheur au laboratoire G-SCOP, non seulement pour les différents échanges scientifiques que nous avons pu avoir, pour sa lecture attentive du mémoire et pour sa participation à ce jury.

Tout au long de cette thèse, de nombreux chercheurs et enseignants-chercheurs m'ont fait l'honneur d'échanger sur mes travaux au travers des réunions de l'équipe GCSP, des journées du laboratoire, ou encore au travers des travaux collaboratifs menés autour du projet Européen INTERGRATE. Pour tout cela, je voudrais remercier Monsieur Michel Tollenaere, Monsieur Stéphane Hubac, Madame Maria Di Mascolo,...

Je souhaite également exprimer mes remerciements à l'ensemble du personnel des services informatique et administratif du laboratoire G-SCOP pour avoir participé à mon intégration au sein du laboratoire et pour la gestion des missions et pour l'organisation de ma soutenance de thèse. Pour tout cela, merci à Marie-Joséphé, Fadila, Christine, Souad, Myriam, Jean-Yves, Sébastien, ...

Je voudrais aussi remercier mes collègues Asma, Anis, Tran et Vo pour leur aide, leur amitié, leurs précieux conseils et pour les discussions qui ont largement contribué à l'écriture de ce mémoire.

Mes remerciements s'adressent enfin tout particulièrement à ma famille, mes parents, mes beaux-parents pour leur amour et leur soutien sans faille, ma femme et mon petit fils qui m'ont accompagné dans tous les bonheurs et toutes les difficultés de la vie. Ils m'apportent le courage et la force pour poursuivre l'aventure de ma thèse et ma vie future.

Table des matières

Introduction générale	5
Partie I Problématique	7
Chapitre 1 Contexte général	9
1 Introduction	9
2 Systèmes Automatisés de Production (SAP)	9
2.1 Définition générale	9
2.2 Le Système de pilotage	10
2.3 Les chaînes fonctionnelles	12
2.3.1 La chaîne d'action	12
2.3.2 La chaîne d'acquisition	12
2.3.3 Le module de contrôle/commande local	13
2.3.4 Observabilité des chaînes fonctionnelles	13
3 Les aléas de fonctionnement	14
3.1 Les défaillances d'un SAP	14
3.2 Les sources de défaillances étudiées	15
4 Diagnostic de défaillance	16
4.1 La réactivité du système de pilotage	17
4.2 Les méthodes de diagnostic	18
4.2.1 Méthodes basées sur la connaissance	18
4.2.2 Méthodes de traitement de données	19
4.2.3 Méthodes à base de modèles	20
4.2.4 Discussion	21
5 Conclusion	21
Chapitre 2 Problématique	23
1 Introduction	23
2 Mécanisme de propagation de défaillances	23
2.1 Description de la propagation de défaillances	23
2.2 Caractérisation des processus de propagation de défaillances	24
2.3 Contrainte temporelle	25
3 Localisation de défaillances issues d'une propagation à base d'un modèle	26
3.1 Approches déterministes	26

3.2	Approches probabilistes	27
3.3	Approches mixtes	28
4	Conclusion	29
Chapitre 3 Positionnement du travail		31
1	Introduction	31
2	Cahier des charges	31
3	Approches retenues pour l'hybridation	32
3.1	Approche de diagnostic logique à base de modèle	32
3.1.1	Description du modèle historique	32
3.1.2	Modèle d'opération et variables d'état	34
3.1.3	Avantages/Inconvénients	36
3.2	Les approches Bayésiennes	38
3.2.1	Définition principale	38
3.2.2	Avantages/Inconvénients	38
4	Principes et originalité de l'approche proposée	40
5	Conclusion	40
Partie II Approche proposée		41
Chapitre 4 Extension de la modélisation		43
1	Introduction	43
2	Modèle historique étendu	43
2.1	Modèle d'opération	43
2.2	Modèles d'opérations et évolutions	44
3	Algorithmes de génération en ligne du modèle historique étendu	47
3.1	Génération dynamique du modèle historique	48
3.2	Gestion des opérations simultanées avant réduction	51
3.3	Mécanisme de réduction dynamique de la taille du modèle	54
4	Le mécanisme de la propagation de la suspicion	56
5	Propriétés du modèle historique étendu	58
5.1	Graphe causal orienté	58
5.2	Nœuds de Convergence	59
5.3	Divergence	60
5.4	Interactions simultanées	60
5.5	Hiérarchie et Hétérarchie	61
6	Conclusion	62
Chapitre 5 Approche Mixte Logique et Probabiliste		63
1	Introduction	63
2	Approche générale proposée	63
3	Transformation du modèle logique en réseau Bayésien	64
3.1	Identification automatique de la structure du RB	64
3.2	Transformation de la structure du modèle	64
3.3	Dégrouperment des nœuds	66

3.3.1	Relations internes entre $\{H, E, R, P\}$	66
3.3.2	Sous-graphe opération	67
3.3.3	Algorithme de dégroupement des nœuds	70
4	Calcul de probabilités	72
4.1	Classe de Réseau Bayésien	72
4.2	Loi de probabilité du RB	72
4.3	Apprentissage des lois de probabilité	73
4.4	Calcul des probabilités	73
5	Conclusion	73
 Partie III Exemple d'application		 75
Chapitre 6 Présentation du cas d'étude		77
1	Introduction	77
2	Système de production de Semi-Conducteur	77
2.1	Généralités	77
2.2	Process de fabrication	78
2.3	Process de production et maîtrise de la qualité de fabrication	79
3	Procédures de contrôle dans le domaine du semi-conducteur	81
3.1	Contrôle Run to run	83
3.2	Contrôle FDC	83
3.3	Contrôle SPC	83
4	Focus sur le processus étudié	84
5	Description des opérations	84
6	Pré-traitement des informations statistiques	85
7	Environnement de développement	87
8	Conclusion	87
 Chapitre 7 Application du diagnostic hybride		 89
1	Introduction	89
2	Scénario 1 : Multi-routes et multi-produits	91
3	Scénario 2 : cas des opérations simultanées	97
4	Scénario 3 : Multi-produits et multi-défaillances	99
5	Discussion	101
6	Conclusion	102
 Conclusion générale		 105
 Bibliographie		 109
 Annexe A Notions de graphe		 117
 Annexe B Notions de Probabilités		 119

Annexe C Description des opérations du cas d'application

121

Introduction générale

Jusqu'à il y a peu de temps encore, les systèmes industriels de production étaient décrits généralement comme « de plus en plus complexes et fortement impactés par une concurrence particulièrement forte ». L'évolution des technologies produits mises sur le marché change encore la donne créant de nouveaux contextes de production. Depuis une courte dizaine d'années, ces évolutions technologiques croissent de manière exponentielle à l'échelle humaine ; que dire de l'évolution des téléphones portables depuis ces dix dernières années ? Pouvait-on imaginer ne serait-ce qu'un instant des potentialités offertes aujourd'hui ? Pouvait-on également imaginer la durée de vie de ces nouvelles technologies ? Quelques mois ? Une technologie en efface une autre à un rythme effréné. Mais comment produire dans un tel contexte ? Comment un site de production peut-il maintenir ses taux de rendements sans avoir même le temps de fiabiliser son système de production avant qu'une nouvelle technologie n'arrive ? Les systèmes industriels de production d'aujourd'hui sont ainsi confrontés à la diversité croissante des technologies à produire qui ont des cycles de vie de plus en plus courts et de surcroît sur des petits volumes de production ; c'est désormais la décade placée sous le signe du « High-Mix Products and Low Volume ».

Il s'agit donc désormais pour le système de production de fonctionner dans un environnement stressé par le produit à forte variabilité et court cycle de vie, environnement dans lequel il faut maintenir des temps de cycle de production courts, tout en assurant la qualité des produits fabriqués le tout à des coûts maîtrisés. Le stress inhérent à ce type de nouveaux environnements ne peut être qu'un terrain favorable à l'augmentation des dérives produits. Pour faire face à ces dérives, l'introduction de boucles de contrôle au travers d'équipement de métrologie généralement coûteux est souvent plébiscitée. Ces installations sont efficaces, mais si elles sont généralisées en sortie de chaque équipement de production, les coûts en investissement et en temps de production en seraient largement impactés. Raison pour laquelle ils sont positionnés suite à plusieurs steps de fabrication. Dans un tel scénario, le principal défi consiste à aider les opérateurs d'exploitation à diagnostiquer les causes d'une dérive détectée afin de mettre en place, rapidement, un plan de maintenance corrective adapté permettant non seulement de corriger la dérive mais de remettre le système en production. Dans ce domaine, de nombreux travaux ont apporté leur pierre à l'édifice. Ces approches vont des méthodes basées sur la connaissance (Shahzad et al., 2012; Li et al., 2013; Shahrabi et Shojaei, 2014), à celles basées sur le traitement de données (Khomfoi et al., 2007; Tran et al., 2009), sans ignorer celles à base de modèles (Deschamps, 2007; Duong et al., 2012a; Weber et al., 2012). Elles adressent chacune des facettes particulières du domaine et amènent des réponses intéressantes soit sur le plan du temps de calcul, soit sur celui de la garantie de performance du résultat obtenu soit encore sur celui de la généricité de l'approche. Naturellement, chacune s'accompagne aussi d'un ensemble d'inconvénients comme pour certaines l'énumération exhaustive des dé-

faillances impossible à envisager pour un système complexe, le temps de calcul conséquent lorsque les données sont importantes, le manque de robustesse à une évolution rapide des gammes de produits, etc. Aucune en fait ne se prête bien à ce nouveau contexte de production dans lequel variabilité produit, réactivité et complexité sont au cœur des rendements de l'entreprise.

Nos travaux de recherche se placent précisément dans ce nouveau contexte de production. L'idée générale que nous défendons ici consiste à hybrider une approche de diagnostic logique avec une approche probabiliste de diagnostic. La première amenant des solutions en terme de réduction de l'espace des causes à rechercher en cas de défaillances, la seconde permettant à l'opérateur d'exploitation de naviguer au cœur de ces causes qui seront affectées d'un score caractérisant une probabilité de pertinence.

Ce mémoire est organisé en trois parties et sept chapitres dont les thèmes sont donnés ci-après :

La première partie présente de manière générale la problématique à laquelle nous nous intéressons. Ainsi après avoir exposé le contexte général des systèmes automatisés de production réactifs aux aléas de fonctionnement, nous positionnons nos apports au niveau temps réel de coordination des équipements de production, en particulier au sein d'une approche diagnostique. Sur cette base, nous dévoilerons le cahier des charges des travaux développés dans ce manuscrit, ainsi qu'une étude des principales approches ayant apporté une contribution au domaine du diagnostic.

La partie II expose notre contribution. Ainsi, après avoir proposé une extension des modèles de diagnostic proposés par M. Eric Deschamps, nous introduisons et détaillons le principe au cœur de nos apports, à savoir l'hybridation entre une approche de diagnostic logique et une approche Bayésienne. L'approche générale décrite consiste à générer en ligne un modèle d'historique étendu à nos besoins, modèle qui est réduit en permanence aux opérations à l'origine potentielle d'une future défaillance. Lorsqu'une défaillance est détectée, un algorithme permet de propager la suspicion sur l'ensemble des opérations du modèle obtenu afin d'en isoler les candidats liés à cette situation de défaillance. Le modèle résultant est ensuite utilisé pour construire la structure graphique du modèle Bayésien, s'affranchissant ainsi de la phase coûteuse en temps d'apprentissage. Les lois de probabilité sont ensuite calculées en s'appuyant sur les bases de données de l'entreprise. Le modèle Bayésien est ensuite questionné pour établir les probabilités de pertinence des causes. L'ensemble de l'approche est décrit tant sur les aspects modèles que algorithmiques.

La partie III développe un exemple d'application des mécanismes proposés sur la base d'un atelier largement inspiré d'un cas d'étude industriel issu du domaine du semi-conducteur. Après avoir présenté de manière générale la partie opérative ainsi que son architecture de pilotage, nous proposons d'appliquer notre approche qui est décrite non seulement en termes de modélisation mais également de programmes que nous avons réalisés sous Matlab. Fort de ces modèles et programmes, nous testons notre approche sur trois scénarii et terminons par une analyse sur les performances temporelles et de consommation mémoire mettant en exergue l'intérêt de nos apports.

Première partie

Problématique

Chapitre 1

Contexte général

1 Introduction

Dans le cadre de ce premier chapitre, nous nous proposons de positionner le contexte général de notre étude. Celle-ci prend place au sein des Systèmes Automatisés de Production (SAP) en environnement stressé par le flux de produit et où le nombre de machines de production et le nombre d'opérations des gammes considérées sont importants. Fort d'une présentation de la structure même de pilotage d'un SAP, nous convergerons rapidement sur la problématique générale auquel ces travaux visent à apporter une pierre supplémentaire, à savoir l'aide à la localisation des causes à l'origine possible des dérives produits détectées en phase de métrologie. Pour cette raison, nous consacrerons la section quatre de ce chapitre à une présentation générale des approches de diagnostic.

2 Systèmes Automatisés de Production (SAP)

2.1 Définition générale

Le contexte de notre travail de recherche prend place au cœur des Systèmes Automatisés de Production (SAP). Ces systèmes se caractérisent par une automatisation partielle voire complète de certaines fonctions comme celles de transformation, d'inspection, de transport ou encore de traitement et de stockage de l'information (Papadopoulos et al., 2009).

L'intérêt des SAPs réside dans leur capacité à réduire les temps de production des produits (AFNOR, 1991) et à augmenter la flexibilité de la structure de production (Berruet, 1998). Ceci permet de pouvoir fabriquer un produit de différentes façons en présence d'incertitudes et de s'adapter assez rapidement à de nouvelles demandes (Deschamps, 2007) de production. La structure générale d'un SAP se présente généralement décomposée en trois parties (Perrin et al., 2004) tel que présenté dans la figure 1.1 :

- Le flux de produits représente la matière première introduite dans le SAP, illustre sa transformation, assemblage et enfin, en sortie, le produit fini.
- La partie opérative se décompose, comme nous le verrons plus loin, de chaînes fonctionnelles (d'actions et d'acquisitions) (Henry et al., 2012). Les chaînes d'actions regroupent l'ensemble des pré-actionneurs, actionneurs et effecteurs qui interagissent sur le flux de produits pour modifier son état au sens le plus large (état physique et position spatiale).

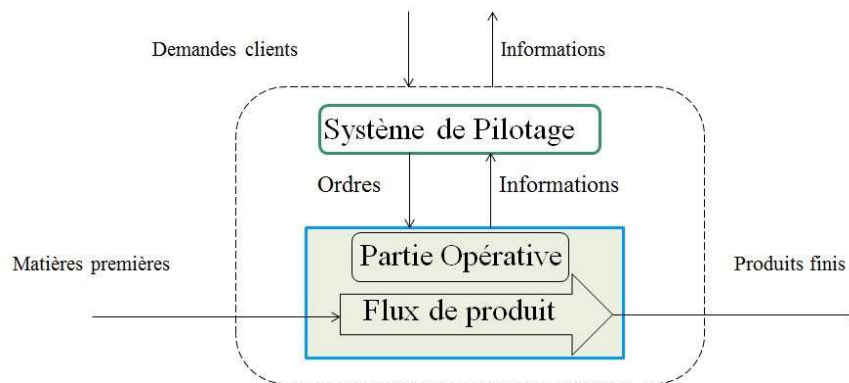


FIGURE 1.1 – Structure générale d'un SAP

Les chaînes d'acquisition permettent de mesurer des grandeurs physiques (vitesse, position, température, taille du produit...) afin de remonter de l'information au système de commande, non seulement pour lever les indéterminismes restant quant à l'arrivée d'un produit devant un poste de transformation que pour vérifier, en phase de transformation, que les ordres électriques émis vers les pré-actionneurs, se transforment bien en actions effectives sur le flux de produits.

- Le système de pilotage permet de contrôler les activités des chaînes d'actions en surveillant les informations remontées par les chaînes d'acquisition afin de satisfaire, dans les meilleures conditions possibles, les demandes de production spécifiées en termes de recettes, de quantités, de délais de fabrication...(Deschamps, 2007).

Afin de répondre aux demandes des clients, qui dans certains domaines de la production (i.e. industrie du semi-conducteur) peuvent varier fortement avec de petits volumes et courte durée de vie, le SAP a naturellement été conçu pour être flexible, non seulement sur sa composante physique (redondance en particulier) mais également sur celle de la logique d'exploitation et donc de son système de pilotage (Berruet, 1998). Ceci permet d'une part de pouvoir fabriquer un produit de différentes façon afin de respecter les délais de production et ce même en présence d'aléas de fonctionnement de la partie opérative et d'autre part de pouvoir rapidement s'adapter à de nouvelles demandes.

2.2 Le Système de pilotage

L'organisation interne d'un système de pilotage peut varier selon plusieurs critères : complexité, variabilité, connexité, etc...En fonction de ces critères, et donc des besoins, plusieurs architectures de pilotage ont été proposées dans la littérature (Scheer, 2012) :

- pilotage centralisé,
- pilotage hiérarchisé,
- pilotage coordonné,
- pilotage distribué,
- pilotage distribué supervisé.

Parmi ces systèmes, il est souvent privilégié (Groover, 2007)(Scheer, 2012) une structure hiérarchique de pilotage permettant une organisation simple, lisible, modulaire, structurée et descendante, fortement axée sur l'intégration de systèmes de commandes à base de cal-

culateurs (Automates Programmables industriels, CNC, Ordinateurs abritant les fonctions de CAO, FAO, Ordonnancement, etc...), l'ensemble relié en réseaux de communication. Ce type de structures hiérarchiques constitue une norme appelée "Computer Integrated Manufacturing"(CIM) (Joshi et Smith, 2012).

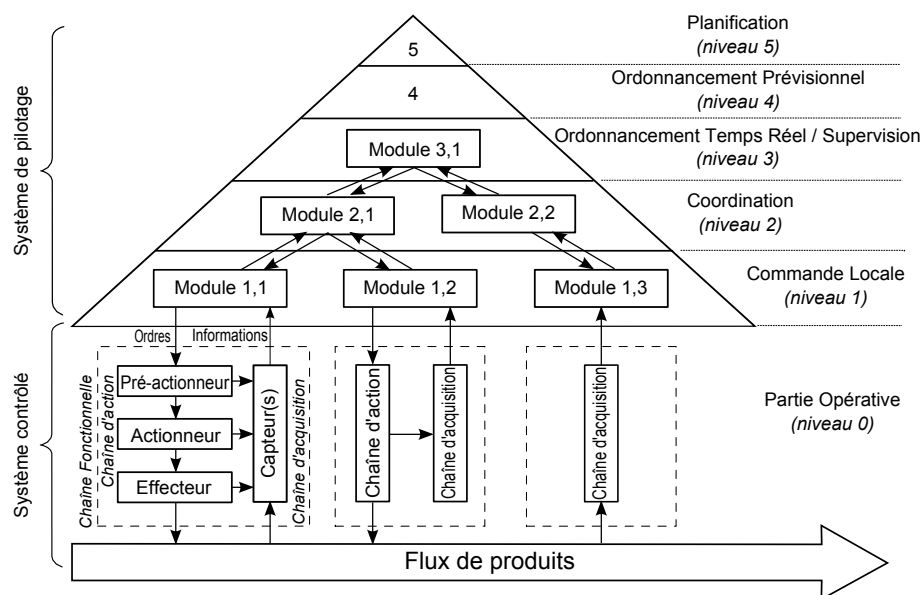


FIGURE 1.2 – Architecture CIM

La philosophie du CIM peut se résumer par la phrase suivante : lorsqu'un système est complexe, il est généralement précaunisé de le décomposer en sous-systèmes de complexité moindre. Ce principe a conduit à une architecture générique dérite dans (CIM, 1989). Nous en reprenons une représentation dans la figure (cf. Figure 1.2). Comme nous pouvons le voir, la structure de pilotage est ici décomposée selon 5 niveaux de pilotage allant de la partie opérative (dite couche d'instrumentation) à la planification en passant par le niveau de la coordination et de la supervision.

Le principe de fonctionnement d'une telle architecture présente un avantage important, son aspect générique. En effet, quelque soit le niveau considéré, il intègre des modèles mathématiques adaptés (modèle d'ordonnancement, Réseau de Petri, Grafcet, Business model, PID, ...), des contraintes propres (exclusions mutuelles, séquençement obligatoires, dates de début au plus tôt, dates de fin au plus tard, nombre de ressources limitées, etc...), des flexibilités temporelles (encore appelées marges temporelles) et donc décisionnelles. Basé sur une telle généralité, le fonctionnement reste simple : lorsqu'un module de niveau n envoie une requête à un module de niveau $n-1$, ce dernier désagrège la requête reçue en m autres requêtes qu'il envoie aux niveaux inférieurs et ainsi de suite jusqu'aux chaînes d'actions. En retour, et en fonctionnement normal, chacun des niveaux ayant généré une requête vers le bas s'attend à une remontée d'information traduisant la bonne réalisation du service demandé. En fonctionnement anormal, la remontée d'information traduit généralement une remise en cause de la demande (requête) et donc un dysfonctionnement.

Dans une telle structure, nos travaux de recherche se localisent dans les niveau temps réels du CIM, et donc en particulier dans les niveaux de coordination et de commande locale ou nous

envisageons d'apporter des solutions complémentaires en terme de diagnostic. Afin de cerner notre problématique, nous proposons au lecteur de rentrer dans le détail de la couche 0 du CIM, couche avec laquelle notre approche de diagnostic va s'interfacer.

2.3 Les chaînes fonctionnelles

Une chaîne fonctionnelle se caractérise par un agencement fonctionnel de constituants sous forme de chaînes, regroupant tous les éléments de la partie opérative, concourant à la réalisation d'une fonction opérative élémentaire (Henry et al., 2012). Elle se compose en général de trois parties : une chaîne d'action, une chaîne d'acquisition et un module de contrôle/commande, cf. figure 1.3.

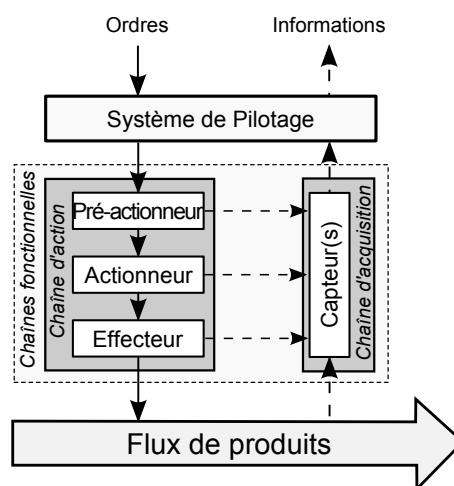


FIGURE 1.3 – Élément de base d'une partie opérative

2.3.1 La chaîne d'action

Une chaîne d'action est chargée de transformer le flux de produit en fonction de l'ordre reçu. Elle est constituée de trois éléments (Perrin et al., 2004), voir figure 1.3 :

Pré-actionneur : il est chargé de transformer l'énergie électrique produite par le système de pilotage en énergie mécanique, électrique, pneumatique ou hydraulique.

Actionneur : c'est un élément moteur de la chaîne d'action. Alimenté en énergie de puissance par son pré-actionneur, il fait fonctionner l'effecteur.

Effecteur : c'est l'élément de la chaîne d'action en contact avec le produit. Il est chargé de sa transformation physique ou spatiale.

2.3.2 La chaîne d'acquisition

La chaîne d'acquisition est chargée de remonter des informations relatives aux grandeurs physiques mesurées et mesurables. Ces mesures peuvent dans certains cas concerner le produit lui-même (positionnement, couleur, géométrie, ...) ou encore son environnement direct (température, humidité, PH, Pression, ...). Ces informations sont le reflet direct ou indirect de l'état du processus en cours d'exécution par la chaîne d'action.

Les principaux constituants d'une chaîne d'acquisition sont (Perrin et al., 2004) :

- une chaîne d'acquisition d'information(s) sur l'état de l'équipement et l'état du produit (métrologie). Elle fournit les informations nécessaires pour la conduite du procédé en observant les déplacements des actionneurs ou le résultat de leurs actions sur le procédé. Ces informations peuvent être de nature très diverses : présence de matière d'œuvre, positions, pressions, courants électriques, températures, débits, codes, etc... Sur le plan technologique, elle s'appuie sur des capteurs.
- une chaîne d'acquisition d'informations extérieure (issue de l'opérateur ; l'opérateur restant à ce jour « le meilleur des capteurs » !).

Sur la base de ces informations collectées, une chaîne fonctionnelle sera en mesure de formaliser un compte-rendu d'exécution qui sera remonté au niveau coordination. Ainsi, au niveau supérieur, ces compte-rendus seront considérés :

- Normaux : compte-rendu (CR) de bonne exécution suite à la demande de service (ordre).
- Anormaux : compte-rendu anormal (CRA) traduisant un dysfonctionnement de la partie opérative pouvant résulter de multiples causes comme nous le verrons plus loin. Ce compte-rendu pourra éventuellement être accompagné d'informations issues d'un diagnostic local effectué au niveau de la chaîne fonctionnelle.

2.3.3 Le module de contrôle/commande local

Un module de contrôle/commande local permet de transformer les requêtes en signaux électriques (carte entrées/sorties d'un automate ou d'un calculateur) exploitables par les pré-actionneurs et d'informer le module de pilotage directement supérieur de la bonne exécution ou non du travail qui lui a été demandé. Sur le plan de la logique, de tels modules s'appuient sur des calculateurs pouvant aller du simple micro-contrôleur à des automates programmables industriels puissants, intégrant des fonctions de régulation, d'asservissement et de pilotage de points de consigne. En fonction de la puissance de ces calculateurs, des fonctions basiques de traitement de défaillances peuvent être embarquées, comme la gestion réactive de situations d'urgence, ou de diagnostic de dysfonctionnement.

2.3.4 Observabilité des chaînes fonctionnelles

Comme nous l'avons vu plus haut, une chaîne d'acquisition permet d'acquérir et de remonter de l'information témoignant de l'état d'exécution de la chaîne d'action sur le produit. Les informations sont issues principalement de capteurs physiques, et leur pertinence (informations) dépend non seulement du nombre de capteurs implantés mais également de leur positionnement dans la chaîne d'action (pré-actionneur, actionneur et/ou effecteur). La position du capteur sur la chaîne d'action a un impact direct sur la confiance accordée à l'interprétation de l'information vis à vis de l'effet que doit avoir la chaîne fonctionnelle (Duong et al., 2012b).

Comme nous pouvons le voir dans cette figure, en fonction de la technologie disponible sur le marché et de l'hostilité de l'environnement relatif (humidité, température, dépôts, radiations, projections, pression, etc), les capteurs ne sont malheureusement pas toujours positionnés à l'interface directe de l'effecteur et du flux produits. Il en résulte souvent une observabilité indirecte de l'effet sur le produit ne permettant uniquement que d'approcher la réalité de la situation ; raison d'ailleurs pour laquelle, des équipements spécifiques de métrologie sont

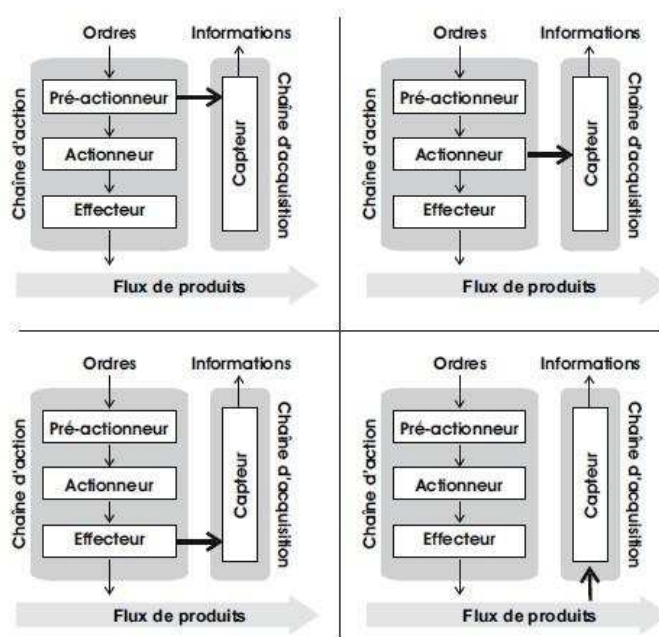


FIGURE 1.4 – Placement des capteurs

positionnés dans les systèmes de production, afin de mesurer, avec retard et dans un milieu spécifique protégé que l'effet réalisé par un équipement de production correspond bien à celui attendu.

De fait, la constitution des comptes rendus émis par les modules de contrôle-commande locaux étant basés sur de telles informations souvent imprécises et incomplètes, la confiance qui peut leur être accordée reste donc variable ; un compte rendu attestant de la bonne exécution d'un service peut s'avérer faux (Duong et al., 2012b). Comme nous allons le voir dans la section suivante, c'est de cette problématique d'observabilité des chaînes fonctionnelles que prend naissance la problématique de la propagation des défaillances dans un système de production.

3 Les aléas de fonctionnement

Nos recherches étant centrées sur la problématique du diagnostic, nous proposons ici au lecteur de présenter davantage dans le détail le concept d'aléas de fonctionnement. Nous commencerons par préciser le concept de défaillances.

3.1 Les défaillances d'un SAP

Une défaillance d'un comportement du système est définie comme une incapacité à réaliser un service (AFNOR, 1991). Une défaillance peut conduire à un défaut sur le produit ou sur le composant. Un défaut est défini par l'écart entre la caractéristique observée et la caractéristique souhaitée (Villemeur, 1992). Une défaillance peut résulter de l'occurrence

d'autres défaillances. Une défaillance première caractérise la source de défaillance.

Les défaillances premières sont généralement classées en deux groupes : externes et internes au système considéré. Les défaillances premières externes peuvent prendre origine dans :

- la commande du client (changement de spécification sur le produit, diminution du délai de production, augmentation des cadences de production),
- la matière première (qui peut ne pas correspondre aux spécifications attendues).
- l'environnement du SAP (opérateurs humains, énergie, etc...).

Les défaillances internes (Atoui et al., 2014) sont généralement dues à :

- la partie matérielle du système de commande (panne du calculateur, erreurs de communication),
- la Partie opérative elle même (panne d'un capteur ou d'un actionneur),
- la partie logicielle (lois de commande) suite à des erreurs de conception, de programmation, etc....

Intéressons nous maintenant aux sources ou causes de ces défaillances.

3.2 Les sources de défaillances étudiées

Dans le domaine du diagnostic et de la maintenance des systèmes, les sources des défaillances sont généralement classifiées selon une approche phénoménologique (Rasmuson et Kelly, 2008). Plusieurs travaux (Ishikawa, 1990), (Doggett, 2005), (Improve, 2012) s'accordent à proposer dans le cadre des SAP une classification selon cinq causes principales clefs : l'Homme, l'équipement de production, les Méthodes ou Recettes, les Matériaux ou Produits et l'Environnement.

Parmi ces cinq facteurs clefs, nous en avons retenu, dans le cadre de notre approche, quatre à savoir l'origine humaine qui est liée au service de maintenance et d'exploitation des équipements, l'équipement lui même vis à vis de ses dérives intrinsèques, les recettes qui sont souvent qualifiées hors contexte réel de production (sans stress donc) et du produit lui même lié en particulier à ces potentielles non conformités en entrée du SAP. Nous avons volontairement exclu ici l'environnement en l'intégrant à l'unique facteur humain. Cette hypothèse sera très certainement à lever dans les travaux futurs.

- **Le facteur humain** : nous entendons bien souvent l'expression dans les environnement de production : "l'humain, un mal nécessaire". L'expression est forte certes, mais caractérise bien d'une part le fait que l'être humain est à ce jour irremplaçable dans les environnements à forte variabilité ne serait-ce que par ses capacités d'adaptation (nous le voyons tous les jours dans de nombreux domaines : contrôle aérien et aussi en atelier pour faire face à des situations nouvelles auxquelles réactivité et sécurité restent les critères majeurs). Et d'autre part qu'il est source de perturbations, soit par manque de vision globale sur ces décisions fasse à un dysfonctionnement, soit parce qu'il est faillible, volontairement ou pas, dans ses tâches (négligence dans les actions de maintenance par exemple) (Kobbacy, 2008; Proctor et Van Zandt, 2011).

- **Les équipements** : l'étude proposée par (Dhillon, 2002; Viveros et al., 2014) met en avant les phénomènes classiques d'usure et de fatigue des constituants des équipements. Ces phénomènes peuvent s'anticiper au travers d'études statistiques ou probabilistes. Cependant il peut être observé des casses ou ruptures de composants imprévues, liées à une mauvaise fabrication, un mauvais montage ou une mauvaise utilisation. (Isermann, 2006) va plus loin d'ailleurs en mettant en exergue le phénomène d'évolution rapide de la technologie, et les conséquences quant à la non maîtrise des concepteurs et fabricants d'équipement face à de telles fréquentes évolutions, engendrant des augmentations sensibles des taux de pannes observés.
- **Les recettes** : tel que nous avons pu le voir dans l'architecture du CIM, le niveau supérieur avec lequel nous nous interfaçons à la capacité de générer des Ordres de Fabrications (OF) qui caractérisent l'ensemble des ordres que doit mettre en œuvre le niveau de coordination. A chacun de ces OF est généralement associée des recettes de production (Munoz et al., 2011). Les recettes sont développées par le bureau de R& D (Erickson, 1999) et qualifiées sur des équipements dédiées avant mise en production. Les recettes se composent d'informations tels que les numéros de lots, les paramètres de réglage des équipements, les dates de début et de fin d'opérations, etc... Les recettes étant généralement qualifiées dans des environnements dédiés non stressés par le contexte réel de production, elle peuvent, en situation réelle, provoquer des défaillances (Munoz et al., 2011). Bien évidemment, plus le contexte réel de production est stressé (Hubac et al., 2010), plus ces phénomènes prennent de l'ampleur.
- **Les produits** : comme nous avons pu le voir dans la figure générique d'un SAP, tout commence par l'introduction de matières premières, qui vont être transformées pour en produire (sortie) un produit fini. Introduire donc de la matière première non conforme, est donc sujet à générer des défaillances (Improve, 2012), qui parfois peuvent s'avérer destructrices pour les équipements eux mêmes (i.e. dureté d'un fil d'acier trop importante placé dans une filière incapable de supporter ce type de contraintes) ou provoquer progressivement des usures prématurées.

Les concepts de défaillances et sources de défaillances étant désormais présentés, nous nous proposons maintenant d'introduire celui du diagnostic.

4 Diagnostic de défaillance

Le diagnostic (Kempowsky et al., 2004; Toguyeni et al., 2003) est une des fonctions qui participe à la réactivité d'un système de pilotage face à l'occurrence de défaillances. Elle a pour objectif de localiser les origines du symptôme de défaillance détecté. Afin de positionner plus précisément cette fonction au cœur du système de pilotage, nous nous proposons dans un premier temps de présenter ce que nous entendons par système réactif.

4.1 La réactivité du système de pilotage

La réactivité du système de pilotage peut être définie comme étant la capacité du système à maintenir la mission qui lui a été confiée en présence d'aléas. Dans ce contexte, de nombreuses études (Combacau et al., 2000; Niel et Craye, 2002; Berruet et al., 2005; ?; Hamari et al., 2009) ont proposé des fonctions de Supervision, Surveillance, et Commande réparties dans chaque module de la structure hiérarchique CIM, qui permettent de surveiller, superviser et commander la partie opérative. Nous rappelons dans la figure 1.5 les résultats issus de la communauté scientifique nationale du groupe de travail GT INCOS¹ (aujourd'hui devenu SED (Systèmes à Événements Discrets) du GDR-MACS².

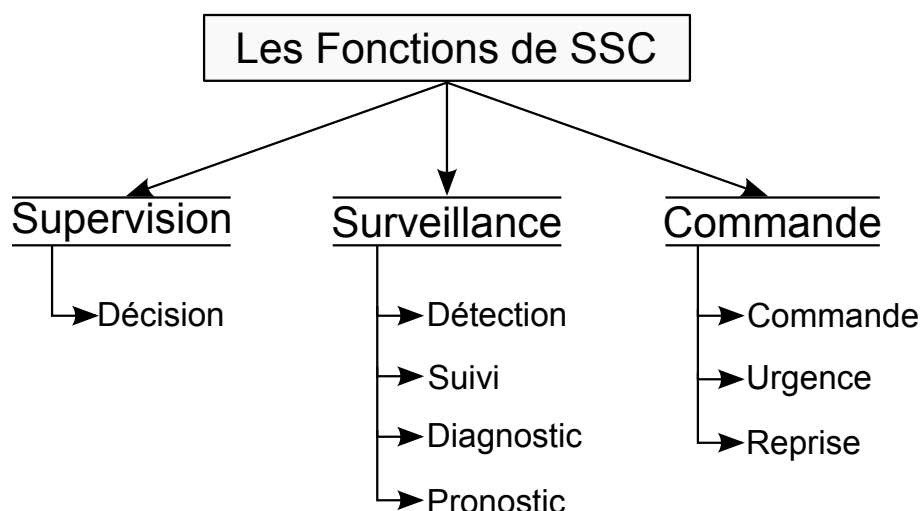


FIGURE 1.5 – Les fonctions de SSC

Les fonctions de la supervision se chargent de contrôler et de surveiller l'exécution d'une opération ou d'un travail effectué par d'autres sans rentrer dans les détails de cette exécution (Combacau et al., 2000). Le rôle des fonctions de supervision est exclusivement décisionnel. Elles déterminent un état accessible pour le retour en fonctionnement normal (Berruet, 1998) et les différentes actions correctives en capacité de modifier la configuration du procédé et de la commande (Henry, 2005).

Les fonctions de la surveillance recueillent en permanence tous les signaux en provenance du système contrôlé (les chaînes fonctionnelles et les flux de produits) et de la commande afin de vérifier en permanence si l'état réel du système considéré correspond à celui attendu. Elles intègrent des fonctions telles que la détection, le suivi, le diagnostic et le pronostic.

Les fonctions de la commande se chargent de piloter le système en exécutant une séquence d'activités de commande (ordres) pour assurer la réalisation d'un produit ou des services contribuant à atteindre un objectif fixé. La commande regroupe toutes les fonctions qui agissent directement sur le système contrôlé : la commande dans un but de production, la reprise pour assurer un retour en production normale, l'urgence pour garantir la sécurité des biens et des personnes.

1. Groupe de Travail INGénierie de la COmmande et de la Supervision des SED

2. Groupement de Recherche, Modélisation, Analyse et Conduite des Systèmes dynamiques

Notre étude se focalise au niveau du diagnostic ; nous proposons dans la section suivante de mener un état de l'art en la matière.

4.2 Les méthodes de diagnostic

Depuis quelques années, de nombreuses méthodes de diagnostic ont été proposées dans le domaine académique et industriel. Afin de diagnostiquer les défaillances dans un système, les méthodes se basent généralement sur la connaissance disponible du système. Cette connaissance dépend des techniques de surveillance du système et d'une référence décrivant le fonctionnement normal et le fonctionnement anormal (Ribot et al., 2009). Sur cette base, les méthodes de diagnostic sont classiquement regroupées selon trois axes (Venkatasubramanian et al., 2003) : les méthodes basées sur la connaissance, les méthodes de traitement de données (méthodes qualitatives et méthodes quantitatives) et les méthodes à base de modèles, voir figure 1.6.

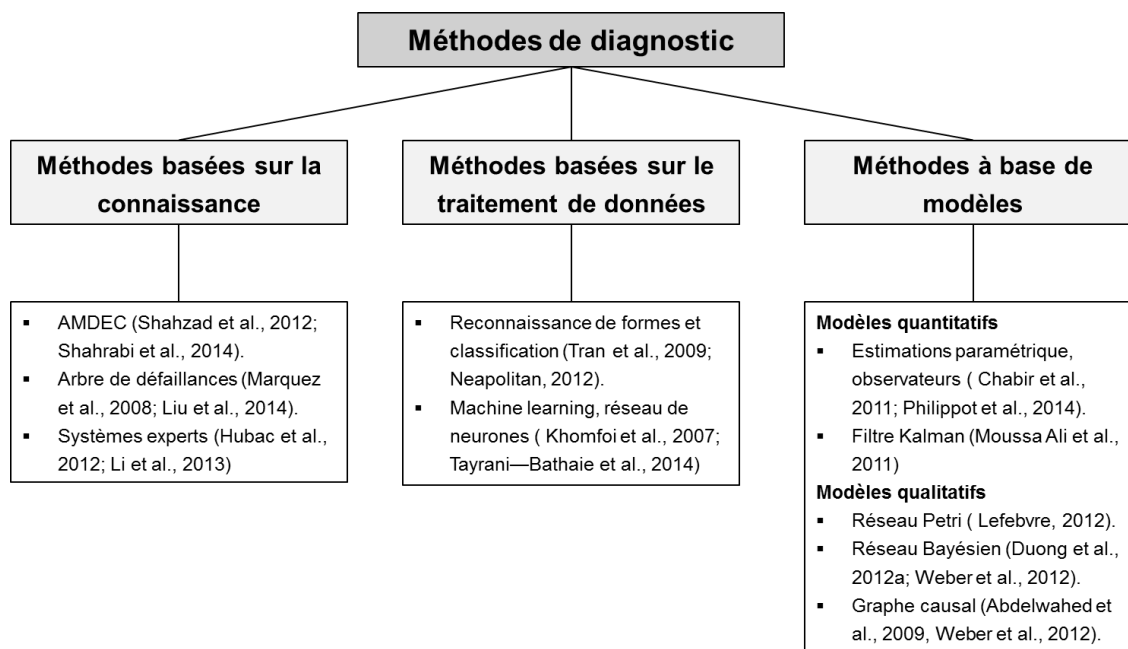


FIGURE 1.6 – La classification des méthodes de diagnostic

4.2.1 Méthodes basées sur la connaissance

Ce type d'approches s'attache à expliquer les potentielles relations causales entre les symptômes, les défaillances et les fautes en se basant sur la connaissance observée du système. La connaissance provient d'une analyse fonctionnelle et structurelle du système et permet d'identifier les causes possibles de défaillances. Ces approches s'appuient sur des méthodes telles que l'AMDEC (Shahzad et al., 2012), les systèmes experts (Sahin et al., 2012), ainsi que les arbres de défaillances (Liu et al., 2014).

Les méthodes de l'Analyse des Modes de Défaillance et de leurs Criticités (AMDEC) (Shahzad et al., 2012) permettent une analyse systématique et très complète, composant par composant, de tous les modes de défaillances possibles tout en précisant leurs effets sur le système

global. Sur la base de cette analyse, ces approches sont proposées pour modéliser les éventuelles défaillances dans un système de production. L'AMDEC est donc très répandue dans de nombreux domaines industriels, l'aéronautique en particulier, cependant elle est lourde à mettre en place pour des systèmes complexes et souvent insuffisante. En effet, il est nécessaire d'identifier a priori les défauts et/ou les dysfonctionnements pouvant apparaître, à un certain moment, dans le système supervisé ; le recensement préalable des défaillances et leurs relations éventuelles ne peut jamais être exhaustif et requiert en général une longue expérience, parfois entachée de croyances et non de connaissances. De surcroît, toute modification ou évolution du système nécessite une réécriture du tableau. Enfin, cette méthode ne peut pas traiter les cas de défaillances multiples et intégrer l'aspect fonctionnel.

Les systèmes experts sont capables de représenter l'expertise humaine existante, de prendre en compte des bases de données existantes, d'analyser de nouvelles connaissances, d'effectuer des inférences logiques et de prendre des décisions basées sur un raisonnement (Li et al., 2013). Dans la plupart des cas, les systèmes experts sont utiles pour aider au diagnostic, en se reposant sur l'apprentissage des relations entre les causes de défaillances et les effets observés. Cependant, il est à noter que ce type d'approches dépend fortement des connaissances d'expertes, et de la nature et de la qualité des bases de données. Notons également que l'apprentissage sur de vastes bases de données reste à ce jour un problème difficile.

Les arbres de défaillances sont des outils majeurs d'analyse du risque technologique (Marquez et al., 2008). Ce sont les méthodes les plus couramment utilisées dans les analyses de fiabilité, ou de sécurité des systèmes. Il s'agit de méthodes optimisées qui permettent la détermination des chemins critiques dans un système. Pour exécuter un diagnostic correct à partir des arbres de défaillances, ceux-ci doivent largement représenter toutes les relations causales du processus, c'est-à-dire qu'ils doivent être capables d'expliquer tous les scénarios de défauts possibles. D'ailleurs, les arbres de défaillances dynamiques permettent de choisir l'état de la sortie des portes dynamiques des arbres. Les conditions (Merle et al., 2010) ou les outils probabilistes (Rao et al., 2009) sont utilisées pour la détermination des états de la sortie des portes dynamiques. Pourtant, le nombre des états de chaque porte est défini avant et toujours statique. De surcroît, l'emploi de cette méthode se révèle difficile pour les systèmes fortement dépendants du temps.

4.2.2 Méthodes de traitement de données

Ces méthodes utilisent les informations disponibles émanant du système de captage. L'objectif de ces méthodes est d'associer un ensemble de mesures aux états de fonctionnement du système. Ces approches à base de données font appel aux méthodes de reconnaissance de formes qui utilisent elles mêmes des techniques d'apprentissage et de classification afin d'établir un modèle de référence. Ce dernier est fondé sur l'analyse des données issues des signaux corrélés entre eux pour la discrimination des différents modes de fonctionnement et certains modes de défaut. Les principales techniques utilisées dans ce cadre pour construire les modèles relatifs sont les réseaux de neurones et la logique floue.

Les réseaux de neurones (Khomfoi et al., 2007) exploitent des observations quantitatives et/ou qualitatives disponibles sous la forme de données historiques ou de résultats de traitement en ligne de signaux issus des capteurs. Ce sont des approches envisageables quand l'obtention d'un modèle analytique du procédé s'avère difficile, et lorsqu'un raisonnement sur

les comportements dynamiques (variables et relations) du procédé n'est pas utile. Le comportement de référence d'un signal, en général statistique, peut être représentatif de l'état normal de l'installation ou d'un défaut particulier.

La logique floue est issue de la théorie mathématique des ensembles flous qui considère des ensembles définis de manière graduelle (Tran et al., 2009). Le concept de fonction d'appartenance permet de modéliser la définition d'un sous-ensemble. La logique floue associe à une donnée un degré d'appartenance à un ensemble qui peut être différent d'un état booléen.

4.2.3 Méthodes à base de modèles

Les approches à base de modèles s'appuient sur des modèles comportementaux explicites du système soumis au diagnostic. Un grand avantage de ces approches par rapport aux approches relationnelles et de traitement de données, réside dans le fait que seule l'information du comportement normal du procédé est prise en compte par l'intermédiaire d'un modèle de référence. La précision du modèle, liée aux besoins de la surveillance et aux critères de performance du diagnostic, définit le choix de l'utilisation de modèles quantitatifs (FDI) ou qualitatifs (DX).

Les approches quantitatives (FDI). La méthodologie générale associée revient à comparer l'entrée et la sortie du système à diagnostiquer. Toutes incohérences exprimées en résidus, peuvent être utilisées à des fins de détection et de localisation. Les approches FDI les plus fréquemment utilisées sont les approches observateurs de diagnostic (Chabir et al., 2014), relations de parité, filtres de Kalman et estimation des paramètres (Moussa et Zhang, 2014). En outre, l'approche présentée dans (Philipot et al., 2014) propose d'utiliser la technique de filtre de contrôle à base d'un diagnostiqueur qui est intégré dans le système.

Les approches FDI sont capables de gérer les variables d'état d'un processus considéré pour diagnostiquer rapidement les erreurs à travers les résidus. Cependant, plusieurs facteurs tels que la complexité du système, les non-linéarités et/ou le manque de données rendent souvent très difficile, voire même impossible, de développer un modèle mathématique précis pour le système. En outre, ces approches sont limitées au traitement mono-défaillance. Si plusieurs défaillances se produisent en même temps, elles fournissent le même résultat de diagnostic. Un autre inconvénient de ces méthodes est que si un défaut n'est pas spécifiquement modélisé (nouvelle défaillance), le diagnostic ne fournit pas de résultat.

Les approches qualitatives (DX). L'approche DX est une approche qualitative basée sur la cohérence qui provient du domaine de l'intelligence artificielle. La technique du diagnostic de cohérence consiste à comparer le comportement réel du système observé et son comportement attendu tel qu'il peut être prédit grâce à des modèles de bon comportement (Ribot et al., 2009). Les travaux de (Deschamps, 2007) et (Duong et al., 2012a) s'appuient sur ces principes pour proposer une approche de diagnostic logique qui repose essentiellement sur la cohérence et le conflit entre les connaissances d'état des comportements du système. Les Réseaux de Petri (RdP) sont utilisés également dans ce cadre pour le diagnostic des SED (Lefebvre, 2012). D'ailleurs, (Mahulea et al., 2012) décrit des systèmes manufacturiers par des RdPs associés à une fonction floue sur les transitions afin de définir des intervalles sur les délais d'exécution des ordres pour localiser les états suspectés des composants du système. (Whiteley et al.,

2015) utilise le RdP temporel pour représenter le fonctionnement normal du système pour la détection. Associé à des arbres de défaillance pour le parcours des événements, il établit la localisation de la défaillance. Une approche de détection et de localisation a été proposée par (Soldani et al., 2006). Un symptôme de défaillance est détecté dès qu'il y a incohérence entre un événement reçu et les événements attendus au niveau du modèle. Le mécanisme de détection s'appuie sur le test de sensibilité des transitions du réseau de Petri. L'étape de localisation consiste à modifier la séquence des observations reçues jusqu'à la détection, pour trouver une séquence de franchissement dans le modèle permettant d'expliquer le marquage du réseau au moment de la détection. Dans les travaux menés par (Duval et al., 2012; Weber et al., 2012), nous pouvons également mettre en exergue les approches à base de Réseaux Bayésien pour le diagnostic de défaillances. Plusieurs techniques probabilistes sont utilisées, par exemple, chaîne de Markov et réduction des risques de système (Weber et al., 2008) pour améliorer les algorithmes probabilistes.

4.2.4 Discussion

Cet état de l'art montre que les méthodes basées sur la connaissance sont utiles dans le cas où un modèle analytique pour le diagnostic n'est pas disponible. Ce type d'approches permet de rapidement associer un symptôme à la faute qui en est la cause. Mais ces approches dépendent fortement de la connaissance disponible, connaissance collectée auprès des experts et opérateurs. Si la base de données est de grande taille ou si le système évolue, il devient très difficile de la maîtriser.

Les méthodes de traitement de données quant à elles s'appuient sur les seuls signaux observés issus des capteurs ; ces méthodes ne reposent pas sur une connaissance physique du système à diagnostiquer. Dépendantes non seulement du système de captage, mais également des techniques mathématiques pour un pré-traitement des signaux observés issus des capteurs, si le système devient complexe, les variables traitées sont très nombreuses. La phase de pré-traitement sera longue, complexe et dépendra de l'expérience des programmeurs.

Le choix d'une méthode de diagnostic dépend donc de la connaissance disponible relative au système considéré, de la capacité d'observation du système de captage ou de la capacité à développer des modèles qui permettent de suivre l'état réel du système. Dans le cadre des fonctions de SSC des SAPs (cf section 4.1 page 17), ces dernières sont largement organisées autour de modèles qui permettent de spécialiser les fonctionnements, d'optimiser l'intégration et la coordination des fonctions entre elles. Dans le cadre de notre approche, nous nous insérons dans cette philosophie orientée modèle pour développer encore davantage les travaux menés jusqu'à présent dans l'équipe. Ils font donc naturellement suites à ceux développés par M. Eric Deschamps et M. Duong Quoc Bao.

5 Conclusion

Nous avons présenté le contexte général du SAP dans ce chapitre. Nous avons ainsi précisé de manière détaillée les éléments centraux des architectures de pilotage les animant et nous nous sommes ensuite focalisés sur leur fonctionnement en présence d'aléas de fonctionnement. Dans ce contexte, nous nous sommes naturellement intéressés aux approches

permettant leur traitement, et en particulier à celles dédiées à la localisation de défauts, à savoir le diagnostic. Le chapitre suivant se propose de décrire la problématique à laquelle ce mémoire est dédié.

Chapitre 2

Problématique

1 Introduction

Après avoir présenté le contexte général du SAP dans lequel évoluent les travaux présentés dans ce manuscrit, ce chapitre se propose d'entrer encore plus dans le détail de la partie opérative, particulièrement, le mécanisme de la propagation de la défaillance dans un processus de production. C'est au cœur de ce processus où la problématique traitée dans ce mémoire se place. Sur cette base, la section 3 présente différentes approches aujourd'hui en capacité de traiter ce problème de propagation à savoir les approches dites déterministes, probabilistes et mixtes. Le chapitre se termine en concluant sur l'intérêt des approches mixtes qui vont être à la base de nos travaux.

2 Mécanisme de propagation de défaillances

2.1 Description de la propagation de défaillances

Parmi les SAP, nous nous plaçons volontairement dans le cadre de systèmes complexes caractérisés par un grand nombre d'équipements de production, une gamme de produits à forte variabilité dans le temps imposant des routes de production diverses et fluctuantes, et des temps complets de production longs. C'est le cas notamment de l'industrie du semi-conducteur dans lequel le processus de fabrication des processeurs dépasse largement les sept semaines de production, sur la base de plus de 300 équipements, plus de 7000 opérations élémentaires (steps) (Improve, 2012). Dans un tel contexte, et compte tenu également de la grande technicité du produit fini, des équipements de métrologie permettent de contrôler, à de nombreuses étapes du cycle de production, la conformité des opérations de transformations exécutées par les équipements de production. Malgré ce contrôle sévère, mais non systématique en sortie de chaque équipement de production (les coûts seraient prohibitifs), plus de 50% des produits sont mis au rebut. Cela montre bien le phénomène que nous évoquions dans le chapitre précédent, la non capacité de pouvoir observer tous les effets issus des effecteurs embarqués dans les équipements de production et de les détecter au moment où ils se sont produits. Les dérives de transformations non observées, finissent tôt ou tard par être détectées, avec retard, par les équipements de métrologie et peuvent également propager des défaillances aux autres équipements de production, initialement sains. C'est le cas par exemple d'une conta-

mination propagée par le produit lorsqu'une opération s'est mal effectuée. Ceci caractérise un mécanisme de propagation de défaillances porté par le produit (cf. figure 2.1).

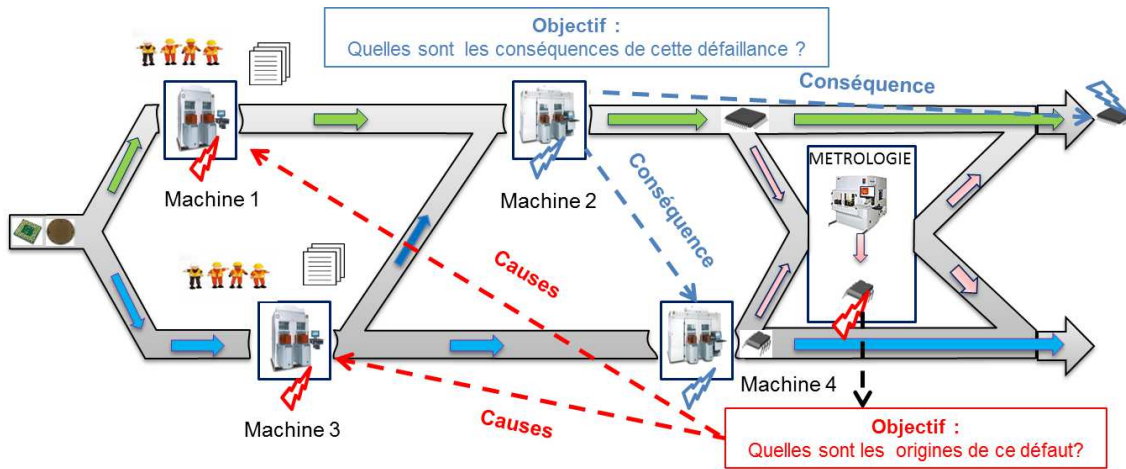


FIGURE 2.1 – Problématique de la propagation de la défaillance

Ce mécanisme nécessite au moins d'un champs d'investigation, d'une part celui de la localisation de l'origine de la défaillance observée en métrologie, et d'autre part l'analyse des conséquences de cette propagation sur les autres équipements de production. D'autres champs sont également à envisager, comme la prédiction de telles dérives. Ces travaux sont par exemple développés dans (Bouaziz et al., 2011; Weber et al., 2012; Asma et al., 2015).

Permettez nous maintenant de rentrer davantage au cœur de ce mécanisme de propagation afin de le caractériser plus finement.

2.2 Caractérisation des processus de propagation de défaillances

Compte tenu de la structure des systèmes de production auquel nous nous intéressons, il est clair que la configuration même de l'atelier et des possibilités de routage des produits impose une caractérisation des possibles lorsque l'on parle de propagation (ceci reste vrai à l'échelle d'un équipement complexe intégrant des services variés de transformation et de routage, i.e. équipement de photolithographie dans le secteur de la micro-électronique) :

Propagations multiples d'une défaillance : c'est le cas lors de l'occurrence d'une défaillance, mais non observée au moment où elle apparaît, pendant l'exécution d'un service offert par un équipement de production. Si en sortie de ce service les produits viennent à être dirigés vers plusieurs autres services, et donc via d'autres routes, la défaillance se propagera via les produits sur autant de routes potentielles (cf. figure 2.2) occasionnant ou pas (le pire, car cas où la défaillance n'est pas confinée en usine... mais probablement chez le client) une défaillance de plusieurs services aval.

Propagation de défaillances multiples : c'est le cas d'un cumul de défaillances portées initialement et unitairement par plusieurs produits passant, in fine, par un équipement ou service « goulot ». Dans un tel cas, cité dans (Bichon et al., 2011) en tant que défaillances multiples, il est fort probable encore que les défaillances cumulées entraînent à terme une défaillance d'un futur service aval avec ici une problématique de localisation rendue difficile par ce chemin concentrateur (cf. figure 2.3).

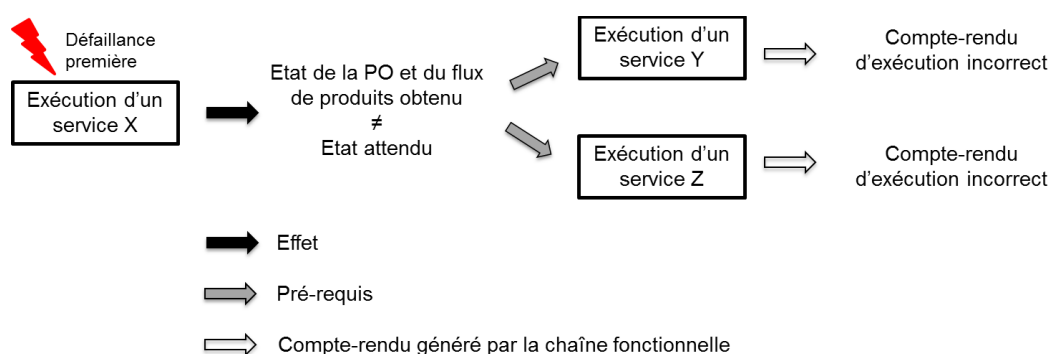


FIGURE 2.2 – Propagation multiple d'une défaillance

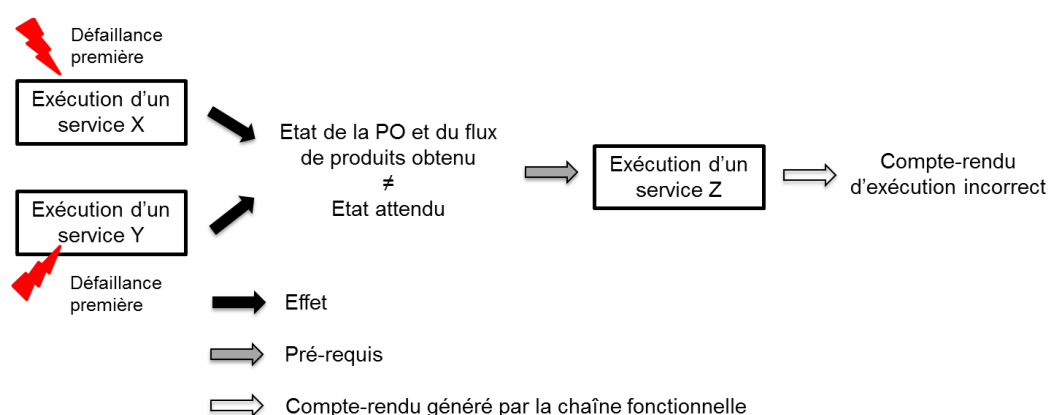


FIGURE 2.3 – Propagation de défaillances multiples

Au delà de ces différents aspects de propagation de défaillances et des solutions que nous proposons plus loin, il est important d'analyser sous quelles contraintes ces solutions doivent opérer.

2.3 Contrainte temporelle

Dans le cadre du principe de fonctionnement générique de la pyramide CIM, nous avons évoqué le concept de flexibilité décisionnelle en présence d'aléas de fonctionnement. Cette flexibilité décisionnelle a été d'une part caractérisée sous l'angle de la flexibilité physique (si le pneu de ma voiture est HS suite à explosion, et que je ne dispose pas de roue de secours, il me sera difficile de satisfaire encore à ma mission de transport et de respecter les contraintes de date due) mais également sous l'angle de la flexibilité temporelle (si je dois me rendre de Grenoble à Nancy ou Lorient sur la base d'un temps calculé au plus court et sans tenir compte d'éventuelles défaillances, au moindre aléa, la mission est immédiatement rendue défectueuse). Le processus général de réactivité à l'aléa fait intervenir plusieurs fonctions de haut niveau (Surveillance, Supervision et Commande) nécessitant chacune, dans l'hypothèse de non énumération des toutes défaillances possibles, un temps de réaction. Suite à détection d'une propagation de défaillance, il est important que le processus de diagnostic, a minima, soit exécuté dans des délais compatibles avec les marges temporelles affectées à chaque service. Ceci est illustré dans la figure 2.4.

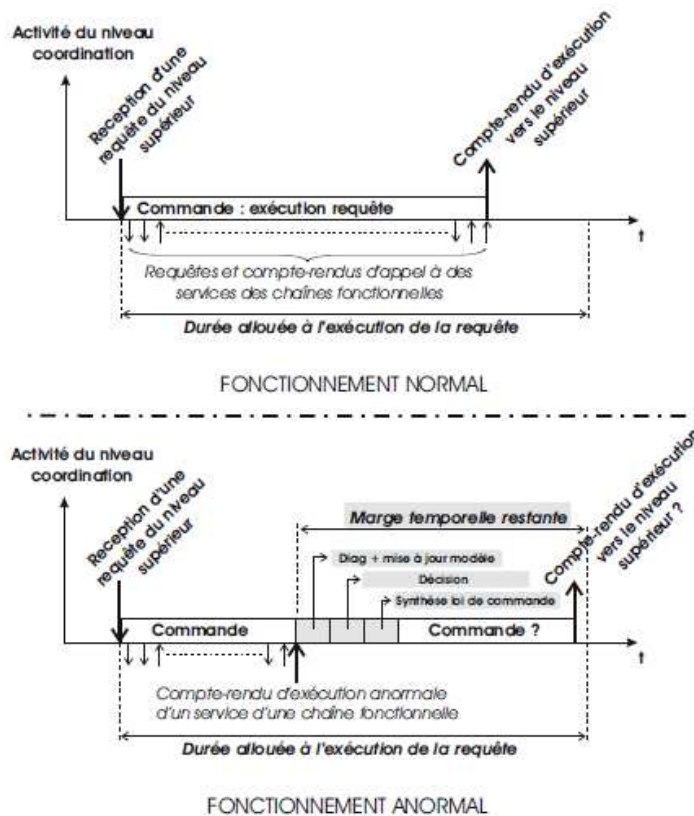


FIGURE 2.4 – Contrainte temporelle

Concepts de propagations et de contraintes temporelles étant désormais présentés, nous nous proposons de positionner notre approche de diagnostic selon ces caractéristiques.

3 Localisation de défaillances issues d'une propagation à base d'un modèle

Nous rappelons ici que dans le cadre des fonctions de SSC des SAPS (cf section 4.1 page 17), ces dernières sont largement organisées autour de modèles. Nous nous focalisons sur le développement une méthode de diagnostic à base de modèles. Parmi les approches à base de modèles présentées dans l'état de l'art, nous nous proposons ici d'évaluer les approches les plus adaptées à la problématique de la propagation de défaillances sous contrainte temporelle. Sur cette base, nous nous permettons de regrouper les approches à base de modèles selon trois types : approches déterministes, approches probabilistes et approches mixtes.

3.1 Approches déterministes

Ces approches sont basées sur les données historiques de production pour déterminer les relations cohérentes entre les comportements du système. Ces relations permettent de localiser l'ensemble des comportements non cohérents qui sont à l'origine possible d'une défaillance. Parmi ces approches, nous pouvons citer les graphes de propagation (Abdelwahed

et al., 2009), les chroniques temporelles (Strasser et Sheppard, 2011), les approches de diagnostic logique (Deschamps, 2007) et ou encore celles basées Réseau de Petri (RdP) (Mahulea et al., 2012).

Le graphe de propagation (Abdelwahed et al., 2009) utilise des relations "Et" ou "Ou" pour modéliser les liens entre les événements dans un système. Les intervalles du temps d'activité d'événements permettent de déterminer l'ensemble des événements liés à l'intervalle du temps d'une suspicion.

Une chronique temporelle est un graphe définissant des contraintes temporelles entre les dates d'occurrence d'un ensemble d'événements générés par un procédé (Philipot et al., 2014). Une chronique est associée à un produit dont l'information temporelle est recueillie en fonction de sa production. L'aspect temporel est pris en compte au travers de la spécification de contraintes temporelles entre les différents événements du procédé. Dès lors, les différentes chroniques doivent être cohérentes entre elles et la fonction de détection a alors pour rôle de reconnaître les évolutions défaillantes.

L'idée fondamentale de l'approche diagnostic logique est de comparer le fonctionnement réel du système observé par l'intermédiaire de capteurs et son fonctionnement prédit grâce aux modèles de bon comportement (Deschamps, 2007). Le résultat de cette comparaison permet d'établir un diagnostic de cohérence. Toute contradiction entre les observations et les prédictions déduites des modèles révèle nécessairement une ou plusieurs défaillances. Le modèle de diagnostic logique associé à cette approche est défini par une description comportementale et structurelle du système à diagnostiquer. La description comportementale est un ensemble de propositions logiques du premier ordre exprimant pour chaque composant la valeur des sorties en fonction des entrées. Ces relations ne devant être satisfaites que si les composants correspondants sont en fonctionnement normal. La description structurelle quant à elle décrit les liens entre les composants. Le modèle est généré en ligne sur la base des événements de commande et de capteurs.

Nous constatons ici que les approches à base de données historiques permettent, en présence d'une détection de défaillance, de lancer des inférences à des fins de diagnostic, en exploitant les modèles générés. Elles présentent cependant des inconvénients que nous listons ci-après :

- Elles dépendent fortement de la capacité à prélever de l'information auprès du processus considéré, ce qui n'est pas toujours évident.
- Si le processus d'inférence révèle plusieurs causes potentielles, elles sont toutes considérées avec la même probabilité de cause. Cela nécessite donc pour les opérateurs de maintenance, si cet ensemble est grand, un temps de raffinement sur le terrain important.

Parmi les approches qui peuvent amener des réponses intéressantes en termes de complétude des informations ou d'affectation de scores permettant à des opérateurs de maintenance d'aller visiter en priorité tel ou tel équipement, nous pouvons naturellement nous référer aux approches probabilistes.

3.2 Approches probabilistes

Parmi les approches probabilistes, il est difficile de ne pas citer celles basées réseau de neurones (Khomfoi et al., 2007) ou réseaux Bayésien (Weber et al., 2012). Elles sont toutes deux dédiées à supporter l'aide à la décision, en particulier dans le domaine de la maintenance.

En particulier, l'approche Bayésienne est très bien adaptée à la problématique du diagnostic de défaillances, puisque basée sur un modèle graphique permettant de modéliser les relations de causes à effets (idéal dans le contexte de propagations de défaillances) et dont les liens peuvent être calculés sous la forme de tables de probabilités conditionnelles (Weber et al., 2008; Neapolitan, 2012). Dans ces méthodes, la structure graphique et les règles probabilistes sont estimées à partir des données historiques observées et la connaissance experte.

Cependant, de nombreuses études telles que (Neapolitan, 2012) montrent que ces approches, s'appuyant sur des techniques d'apprentissage, sont encore complexes à mettre en œuvre et fortement dépendante de l'expert et de son expertise. L'identification structurelle du modèle Bayésien reste longue et dépend beaucoup de la taille de l'historique. L'approche est de plus peu propice aux environnements à forte variabilité, puisque nécessitant des apprentissages réguliers et longs. Dans le contexte dans lequel nous nous plaçons (multiproduits, multiroutes de production, multirecettes), ce type d'approche pour le diagnostic dynamique n'est malheureusement pas approprié.

Afin de palier à ces problèmes, depuis quelques années, des approches mixtes commencent à voir le jour afin de déterminer un "bon" compromis parmi les approches précédemment citées et profiter ainsi de leurs avantages respectifs. Nous en présentons quelques unes ci-après.

3.3 Approches mixtes

De nombreuses méthodes mixtes basées sur une hybridation entre une approche déterministe et une probabilistes telles que un arbre de défaillances & un réseau bayésien (Marquez et al., 2008), une approche de type FMEA & un réseau bayésien (Garcia et Gilabert, 2011), ou encore un arbre de défaillances & un réseau neuronal (Yang et al., 2009) existent. L'approche déterministe permet de construire la structure du modèle de diagnostic sans apprentissage. Les probabilités obtenues par les algorithmes probabilistes permettent d'évaluer les priorités dans le processus d'accompagnement de l'expert en phase de diagnostic.

Cependant, les approches basées sur les modes de défaillances trouvent leur limite dans l'énumération exhaustive de tous les modes de défaillances dans un système complexe dynamique.

A l'issue de cette section, nous avons pu déterminer que la piste des approches mixtes reste une voie particulièrement intéressante dans le cadre de la problématique de recherche des causes de propagations de défaillances dans un système complexe, dynamique et à forte variabilité. Nous retenons que la structure du modèle est fondamentale pour assurer, par inférence, une recherche des causes. Que la structure de ce modèle, si elle est obtenue par apprentissage, reste incompatible avec le contexte de forte variabilité et de réactivité. Que si elle est obtenue par une approche déterministe, elle reste à l'évidence incomplète. Enfin, que l'approche probabiliste Bayésienne s'avère être précieuse quant à sa capacité à calculer des probabilités conditionnelles (les faits observés et connus) et donc à établir des priorités dans les causes à évaluer en priorité.

4 Conclusion

Ce chapitre nous a permis de présenter notre problématique centrée sur le traitement particulier de la propagation des défaillances au sein d'une partie opérative. Ce phénomène de propagation s'explique en particulier au travers de l'absence de moyens de mesure au sein des équipements de production, moyens parfois impossible à imaginer à notre époque (problèmes résistance au milieu par exemple). Sur la base de cette problématique, le chapitre s'est attaché à présenter les différentes voies de recherche qu'il est possible d'imaginer à ce jour ; nous avons ainsi présenté des approches déterministes, probabilistes et mixtes en mettant l'accent sur leurs avantages et inconvénients. Les approches mixtes se présentent comme les plus adaptées à notre problématique alliant adaptation au contexte réactif pour celles déterministes, et adéquation à la complexité pour celles probabilistes. Le chapitre suivant va dans ce sens et positionne nos travaux de recherche.

Chapitre 3

Positionnement du travail

1 Introduction

Ce chapitre, véritable charnière de notre mémoire, s'attache à positionner notre étude. Faisant suite aux travaux menés dans notre équipe de recherche par mes prédécesseurs, M. Eric Deschamps et M. Duong Quoc Bao, notre approche reprend les principes du diagnostic logique à base de modèles pour l'étendre et l'améliorer encore sur le plan de la précision de l'aide au diagnostic. Ces extensions et améliorations sont multiples. Elles vont de l'extension des modèles initialement proposés jusqu'à l'hybridation de l'approche avec une approche Bayésienne. Ainsi, les sections suivantes s'attachent à cadrer le cahier des charges que nous nous sommes fixés puis rentrent dans le détail de l'approche que nous avons retenu en tant que fondation et amenons par la suite les bases de l'approche Bayésienne. Le chapitre se termine par une synthèse des spécificités et originalités de nos travaux.

2 Cahier des charges

Dans le contexte d'une structure de pilotage d'un système complexe de production (forte variabilité produit, petits volumes, multi-routage dynamique, nombre importants d'équipements, nombre importants d'opérations, cycle de production long), nous définissons en tant qu'objectif, la spécification et la conception d'une fonction de diagnostic dynamique, localisée dans les niveaux temps réel du CIM, à savoir au niveau coordination.

Cette fonction sera déclenchée sur l'occurrence d'un compte rendu anormal reçu du niveau immédiatement inférieur, caractérisant une défaillance dans le service demandé (remise en cause). Les causes de cette défaillance qui sont à rechercher peuvent prendre différentes origines comme un équipement, une recette de fabrication, une mauvaise intervention humaine dans le cadre d'une maintenance préventive ou encore une mauvaise spécification de la matière d'œuvre injectée dans le SAP. Parmi ces comptes rendus anormaux, nous ne nous intéresserons qu'à ceux résultant d'un échec de diagnostic au niveau local, caractérisant que l'origine de la défaillance détectée à bas niveau ne vient pas du module en question ; les autres cas ne présentant en effet aucun intérêt au niveau coordination en terme de diagnostic.

Les causes seront donc à localiser dans le passé et devront être classées par ordre de priorité dans le cas de causes incertaines, le tout en respectant des contraintes temporelles, fixées par les marges temporelles issues des Ordres de Fabrication (Date de fin au plus tôt,

date de fin au plus tard).

Comme nous l'avons déterminé dans le chapitre précédent, notre voie de recherche se place volontairement dans une démarche d'hybridation d'approches déterministe et probabiliste afin de profiter des avantages de chaque type distingué. La suite de ce chapitre se propose de détailler les deux approches que nous avons retenues pour cette hybridation.

3 Approches retenues pour l'hybridation

Nos travaux de recherche faisant suite à ceux initiés dans ce domaine par M. Eric Deschamps et M. Duong Quoc Bao au sein de l'équipe Gestion et Conduite des Systèmes de Production du laboratoire G-SCOP, nous avons naturellement proposé d'étendre l'approche existante. Elle s'appuie sur la génération en ligne d'un modèle d'historique dont la taille est maîtrisée (mécanisme d'oubli) et d'un mécanisme de propagation permettant d'exploiter le modèle résultant à des fins de diagnostic ou de pronostic. Nous proposons ici de retenir l'approche de génération en ligne de modèle comme base à la structuration dynamique d'un réseau Bayésien ; structure qui sera ensuite complétée par les tables de probabilités conditionnelles au travers de bases de données "propres" (après traitement) issues du site de production considéré (Nguyen et al., 2014a). La technique de propagation arrière sera quant à elle retenue également à des fins de restriction de l'espace des causes potentielles relatives à l'occurrence d'une défaillance. L'approche Bayésienne permettra alors, dans cet espace réduit des causes d'en déterminer les probabilités de pertinence (Nguyen et al., 2014b).

L'approche générale étant présentée, nous nous proposons maintenant de présenter davantage dans le détail les éléments essentiels à retenir des deux approches qui sont à la base de notre proposition.

3.1 Approche de diagnostic logique à base de modèle

3.1.1 Description du modèle historique

L'approche générale proposée dans le cadre des travaux de recherche de M. Eric Deschamps (Deschamps, 2007) est structurée selon deux axes temporels : une démarche hors ligne de modélisation des comportements normaux des chaînes fonctionnelles, une démarche en ligne permettant d'exploiter le modèle de fonctionnement normal généré à des fins de suivi et de diagnostic.

D'un point de vue principe de fonctionnement de l'approche, cette fonction de diagnostic est organisée comme la figure 3.1 et est présentée selon les trois points principaux suivants :

- Dans un premier temps, et sur la base d'une extension d'un outil de modélisation issu de la planification automatique Henry (2005), l'ensemble des services offerts par les différentes chaînes fonctionnelles sont à modéliser en exprimant ses services sous forme de comportements opératoires, de durée, de préconditions et de précontraintes à l'exécution du service, de conditions et de contraintes durant l'exécution, d'effets intermédiaires et finaux exprimant l'état des variables d'état des chaînes fonctionnelles pendant et après l'exécution de l'opération. Cette démarche, certes longue pour un premier système de production à considérer reste incontournable. Elle présente, comme nous le verrons par

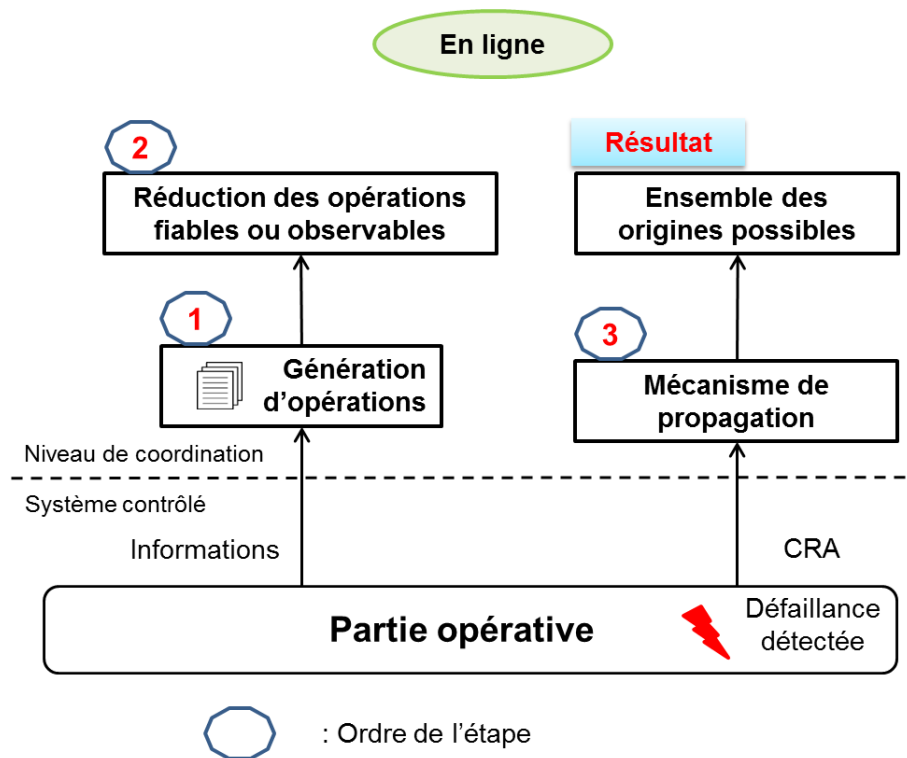


FIGURE 3.1 – Fonction de diagnostic du modèle historique

la suite de nombreux avantages, comme par exemple celui de geler des services lorsqu'ils sont devenus non opérationnels, de rajouter facilement de nouvelles descriptions correspondant à l'ajout de nouvelles chaînes fonctionnelles ou d'en modifier certaines en fonction de l'état réel de la partie opérative. Ces aspects ont d'ailleurs été mis à profit d'une approche de reconfiguration automatique exploitant les degrés de liberté restant suite à occurrence de défaillances (Henry et al., 2012). Il résulte de cette étape une collection de l'ensemble des services offerts par une partie opérative exprimé sous la forme de fiches.

- Dans un deuxième temps, et donc ici en ligne, un algorithme de génération dynamique du modèle d'historique prend le relais. Il est rythmé par les événements de commande issus des modules de coordination vers les chaînes fonctionnelles ainsi que par les événements de comptes rendus émis (Henry, 2005). Pour ce faire, à chaque émission d'une commande vers le niveau inférieur, l'algorithme parcourt les fiches opératoires pour d'une part « recopier » la fiche correspondante au service demandé et l'instancier en fonction de la demande et du contexte. Chaque fois qu'un compte rendu est reçu, les variables d'état de type effets intermédiaires et effets finaux sont affectés d'un score (couleur variant du blanc au noir) correspondant à la confiance du compte rendu reçu. Un algorithme basé sur une approche Bayésienne permet cette évaluation (Duong et al., 2012a). Lorsqu'une autre commande est générée (i.e. action suivante dans l'exécution du Grafset de commande par exemple), une nouvelle fiche est insérée dans le modèle historique et connectée aux variables d'état de sortie des fiches précédentes ; les effets finaux précédents devenant des pré-conditions et de pré-contraintes pour l'opération suivante. Lorsque le niveau de confiance atteint le score de 1/1 (le compte rendu à été

validé par métrologie), la fiche correspondante est retirée du modèle partant du principe que si le résultat attendu est confirmé, toutes les opérations précédentes se sont correctement exécutées. La taille du modèle est ainsi maîtrisée, sous réserve bien entendu d'une certaine capacité de métrologie ; sans équipement de métrologie, la taille du modèle ne cesserait de croître. La structure graphique résultant est le reflet de l'histoire du process, l'histoire ne retenant que les candidats potentiellement à suspecter si une défaillance venait à être détectée. Cependant, il peut exister des phénomènes de compensation de défaillances qui peuvent masquer une défaillance potentielle. Dans le cadre de notre approche, nous proposons une hypothèse que ces phénomènes ne sont pas pris en compte. De ce fait, l'approche de diagnostic logique suppose que si la caractéristique du produit mesuré est conforme, toutes les opérations correspondant à ce produit sont qualifiées fiables.

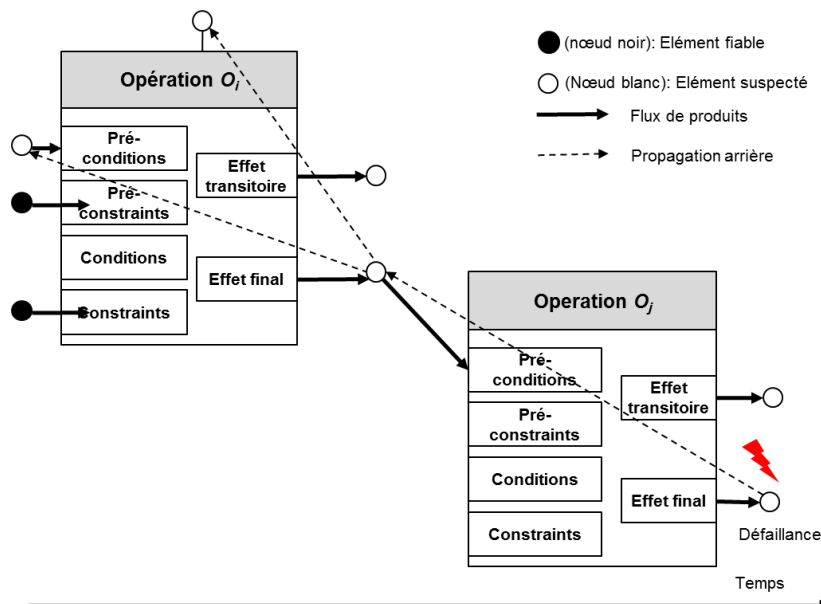


FIGURE 3.2 – Exemple du modèle historique

- Lorsqu'un compte rendu anormal est reçu au niveau coordination, un algorithme de propagation arrière de la suspicion est activé au sein du modèle historique résultant permettant ainsi de localiser l'ensemble des candidats probablement à l'origine de la défaillance. Un mécanisme de propagation avant permet quant à lui de pronostiquer les conséquences de cette même défaillance, voir figure 3.2. Pour plus de détails, le lecteur pourra se reporter à (Deschamps, 2007)

Le principe général de cette approche à base de modèles étant présenté, nous nous proposons désormais nous nous attarder sur le modèle d'opération proposé.

3.1.2 Modèle d'opération et variables d'état

La modélisation des services offerts par les équipements de production ou de métrologie s'appuie sur une extension des modèles d'opérations issus du domaine de la planification automatique (Henry, 2005). Ils prennent la forme d'une fiche très structurée que l'expert atelier doit remplir. Une fiche modélisant une chaîne fonctionnelle (CF_k) est composée de :

Opération O_i		
Chaîne fonctionnelle CF_k		
Instant de début : t_s		Disponible
Instant de fin: t_f		
Évolution chaîne fonctionnelle ec_i		
Effet Transitoire: EFT (ec_i) IC(EFT(ec_i)))	Pré-condition: Pcd(ec_i)	Pré-contrainte: PCt(ec_i)
Effet final: EfF(ec_i) IC(EfF(ec_i)))	Condition: Cd(ec_i)	Contrainte: Ct(ec_i)
Évolution de flux de produits $ea_{i,1}$		
Effet Transitoire: EFT ($ea_{i,1}$) IC(EFT($ea_{i,1}$)))	Pré-condition: Pcd($ea_{i,1}$)	Pré-contrainte: PCt($ea_{i,1}$)
Effet final: EfF($ea_{i,1}$) IC(EfF($ea_{i,1}$)))	Condition: Cd($ea_{i,1}$)	Contrainte: Ct($ea_{i,1}$)

FIGURE 3.3 – Définition du modèle d'opération

- Du_i , la durée estimée de l'opération Oa_i .
- ec_i , l'évolution de la chaîne fonctionnelle, elle composée de :
 - o $EfT(ec_i)$, l'effet transitoire.
 - o $EfF(ec_i)$, l'effet final.
 - o $PCd(ec_i)$, la pré-condition à respecter avant le lancement de l'opération pour que l'effet transitoire de la chaîne fonctionnelle soit réalisé.
 - o $Cd(ec_i)$, la condition à respecter pendant l'exécution de l'opération pour que l'effet final de la chaîne fonctionnelle soit réalisé.
 - o $PCt(ec_i)$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération.
 - o $Ct(ec_i)$, la contrainte à respecter durant l'exécution de l'opération.
 - o $IC(EfT(ec_i))$, l'indice de confiance associé à l'effet transitoire.
 - o $IC(EfF(ec_i))$, l'indice de confiance associé à l'effet final.
- $ea(i, j); j \in [1, N_i]$, les évolutions associées du flux de produits pour une opération Oa_i avec N_i évolution possible du flux de produits. L'évolution $ea(i, j)$ se compose de :
 - o $EfT(ec(i, j))$, l'effet transitoire du flux de produits.
 - o $EfF(ec(i, j))$, l'effet final du flux de produits.
 - o $PCd(ec(i, j))$, la pré-condition à respecter avant le lancement de l'opération pour que l'effet transitoire $EfT(ec(i, j))$ soit réalisé.
 - o $Cd(ec(i, j))$, la condition à respecter pendant l'exécution de l'opération pour que l'effet final $EfF(ec(i, j))$ soit réalisé.
 - o $PCt(ec(i, j))$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération.
 - o $Ct(ec_i)$, la contrainte à respecter durant toute la durée de l'exécution $ea(i, j)$.

Ces informations sont ensuite complétées pour des besoins spécifiques de diagnostic.

- les liens entre les (pré)conditions, les (pré)contraintes et les évolutions du flux de produit,

ainsi que les liens entre les disponibilités des opérations :

- o $Surv_{i,j}$ pour $j \in [1, N_i]$, les différents comportements des produits, pour une opération Os_i avec N_i surveillances possibles.
- o $VE(Surv_{i,j})$, la variable d'état surveillée.
- o *valeur1*, *valeur 2*,... les valeurs de la variable d'état qui peuvent être surveillées.
- o $PCd(Surv_{i,j})$, la pré-condition à respecter avant le lancement de l'opération pour effectuer la surveillance.
- o $Cd(Surv_{i,j})$, la condition à respecter pendant l'exécution de l'opération.
- o $PCt(Surv_{i,j})$, la pré-contrainte à respecter sur l'état du flux de produits des chaînes fonctionnelles avant le début de l'opération.
- o $Ct(Surv_{i,j})$; la contrainte à respecter pendant l'exécution de l'opération.
- les descriptions de l'opération particulière, notamment l'opération de surveillance, définissant les expressions logiques entre les variables d'état contenus dans les (pré)conditions, les (pré)contraintes et dans les effets.
- les informations permettant de déduire le contexte d'exécution d'une opération, c'est-à-dire l'ensemble $j \in J$ des évolutions attendues au lancement de l'opération.
- l'ensemble $j \notin J$ des évolutions non souhaitées du flux de produit associé à une opération. Cet ensemble se compose de :
 - o $PCd_NS(ea_{i,j})$ une expression logique correspondant à l'expression de la pré-condition de $ea_{i,j}$.
 - o $Cd_NS(ea_{i,j})$ une expression logique correspondant à l'expression de la condition de $ea_{i,j}$.
 - o $PCt_NS(ea_{i,j})$ une expression logique correspondant à l'expression de la pré-contrainte de $ea_{i,j}$.
 - o $PCt_S(ea_{i,j})$ une expression logique correspondant à l'expression de la pré-contrainte à laquelle les conjonctions fausses portant sur les tests des variables d'état ont été retirées.
 - o $Ct_NS(ea_{i,j})$ une expression logique correspondant à l'expression de la contrainte de $ea_{i,j}$.
 - o $Ct_S(ea_{i,j})$ une expression logique correspondant à l'expression de la contrainte à laquelle les conjonctions fausses portant sur les tests des variables d'état ont été retirées.

En résumé, nous avons des variables d'état exprimant des pré-requis (PCd , PCt , Cd , Ct , PCd_NS , Cd_NS , Pct_NS , Ct_NS , PCt_S ou Ct_S), des effets des opérations exécutées et des variables relatives aux opérations de surveillance ($Surv_{i,j}$), en fait des opérations de métrologie. Ces variables d'état fournissent les informations sur le comportement des opérations, des évolutions des chaînes fonctionnelles et des évolutions du flux de produits.

La section suivante analyse les avantages et les inconvénients du modèle historique de (Deschamps, 2007).

3.1.3 Avantages/Inconvénients

Par rapport aux approches déterministes, le diagnostic logique se basant sur un modèle historique proposé par (Deschamps, 2007) amène les avantages suivants :

- Basée sur un modèle du fonctionnement normal, elle contourne le problème lié à la quête in-atteignable d'exhaustivité de description des défaillances et de leurs causes : « ce qui n'est pas normal est forcément anormal ».
- Adaptée au diagnostic en temps réel pour fournir un ensemble de causes possibles de la défaillance.
- La taille du modèle est maîtrisée par le principe de métrologie.

Malgré ces avantages, des inconvénients et limitations restent présents (Nguyen et al., 2015).

Problème 1 : l'approche prend pour hypothèse très restrictive que l'origine des défaillances prend forcément comme coupable l'équipement de production. Or de nombreux travaux tel que (Bouaziz et al., 2011; Asma et al., 2015) attestent que d'autres facteurs sont à prendre en compte comme la maintenance, la matière d'œuvre, les recettes, etc... et à considérer comme potentiels responsables.

Problème 2 : le mécanisme de génération du modèle historique ni celui de propagation arrière ne permettent de gérer les interactions simultanées d'opérations exécutées en parallèle et en mode collaboratif (l'une et l'autre étant précondition et précontrainte de l'autre) cf. figure 3.4. De telles situations peuvent occasionner des phénomènes de boucles infinies dans les algorithmes de recherche.

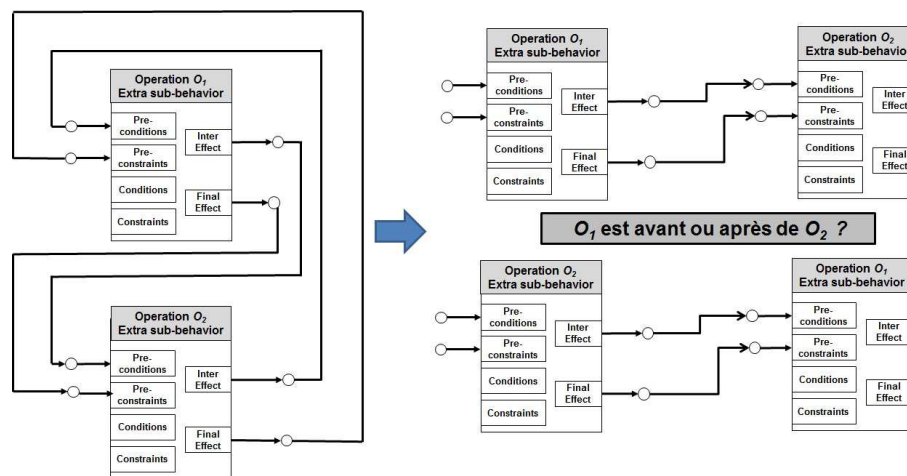


FIGURE 3.4 – Les interactions simultanées entre les opérations

Problème 3 : ensemble résultant de la recherche des causes possibles peut rester encore important et rien ne permet de guider l'opérateur humain dans le choix des causes à étudier en priorité.

Aux problèmes 1 et 2 seront apportés des améliorations des algorithmes proposés dans (Deschamps, 2007) et (Duong et al., 2012a) indépendamment de l'approche d'hybridation que nous avons suggérée précédemment. Quant au problème 3, il fait justement l'objet de tout l'intérêt de la démarche d'hybridation avec une approche probabiliste Bayésienne. Aussi, la section suivante présente les principes d'une telle approche.

3.2 Les approches Bayésiennes

3.2.1 Définition principale

Les réseaux Bayésiens figurent parmi les modèles d'analyse probabiliste. Ils proposent un formalisme mathématique et des bases théoriques solides pour le développement des modèles de systèmes complexes. Les RBs sont de plus en plus utilisés dans les domaines tels que l'analyse des risques, la sûreté de fonctionnement et la maintenance (Weber et al., 2012). Ils combinent une représentation de la connaissance sous la forme graphique (des relations de dépendance directe) et probabiliste (une incertitude sur la connaissance) (Naïm et al., 2011).

Définition 1. Un réseau bayésien (Naïm et al., 2011) est défini par :

- Un graphe acyclique orienté $G, G = (V, E)$, où V est l'ensemble des nœuds de G , et E est l'ensemble des arcs de G ,
- Un espace probabilité (Ω, Z, P) , avec Ω est un ensemble fini non-vide, Z est un ensemble de sous-espace de Ω , et P est une mesure de probabilité sur Z avec $P(\Omega) = 1$,
- Un ensemble de variables aléatoires associées aux nœuds du graphe G et défini sur (Ω, Z, P) , tel que :

$$P(V_1, V_2, \dots, V_n) = \prod_{i=1}^n P(V_i | C(V_i)) \quad (3.1)$$

Où $C(V_i)$ est l'ensemble des parents de V_i dans le graphe G .

La construction d'un réseau Bayésien est réalisée suivant trois étapes nécessaires.

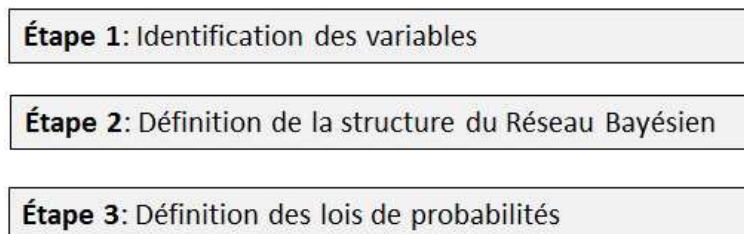


FIGURE 3.5 – La construction d'un Réseau Bayésien (Naïm et al., 2011)

3.2.2 Avantages/Inconvénients

Il existe de nombreuses représentations pour extraire des connaissances à partir des données (arbres de décision, réseaux de neurones, etc), il en est de même pour les techniques de fouilles de données (classification, régression, filtrage, etc). Parmi ces méthodes, les réseaux Bayésiens ont attiré notre attention en raison de leurs propriétés en terme de modélisation. Nous listons leurs principaux avantages (Naïm et al., 2011) :

- les RBs peuvent être utilisés pour apprendre des dépendances causales et pour modéliser des phénomènes aléatoires,
- un RB est une représentation graphique compacte et synthétique avec une facilité d'acquisition et d'utilisation de la connaissance,
- les RBs peuvent combiner plusieurs aspects à la fois, statistiques, probabilités, aide à la décision et management des connaissances,

- pour l'utilisation, leur flexibilité permet d'interroger le même modèle graphique pour des objectifs différents, tels que la pronostic ou le diagnostic,
- de surcroît, ils permettent de modéliser la connaissance par une attribution des probabilités même si les données sont de natures incertaines,
- les algorithmes dédiés au calcul offrent un outil puissant pour la fusion des données incomplètes avec la prise en compte des jugements des experts,
- pour les applications industrielles, la qualité de l'offre en matière d'outils rend les RBs de plus en plus séduisants. Nous pouvons montrer les logiciels pratiques tel que BayesiaLab (Conrady et Jouffe, 2013), Probayes (Tay et al., 2008), Matlab (Dejaeger et al., 2013), Genie smile (Millan et al., 2013), Netica (Fuster et al., 2014).

Malgré ces avantages, subsistent toujours naturellement quelques inconvénients. Les étapes 1 et 2 de la démarche Bayésienne (cf. Figure 3.5) reposent essentiellement sur la capacité et la connaissance de l'expert. S'il n'existe pas d'avis de l'expert, les méthodes d'apprentissage seront utilisées. Nous distinguons deux types d'apprentissages : paramétrique et structurel. Le premier permet d'estimer les lois de probabilités conditionnelles (paramètres) étant donnée une structure de RB, quant au second, il vise à déterminer une structure graphique optimale du modèle à partir d'un ensemble de données observées. Les méthodes d'apprentissage citées appartiennent à la famille d'apprentissage à partir de données, la deuxième famille s'appuie quant à elle sur l'acquisition de connaissances (les jugements de l'expert) (Renooij, 2001) comme illustré dans la figure 3.6. Dans le cas de l'apprentissage à partir de données, cela suppose que les données sont riches, triées et exactes, ce qui n'est pas toujours trivial en situation réelle industrielle.

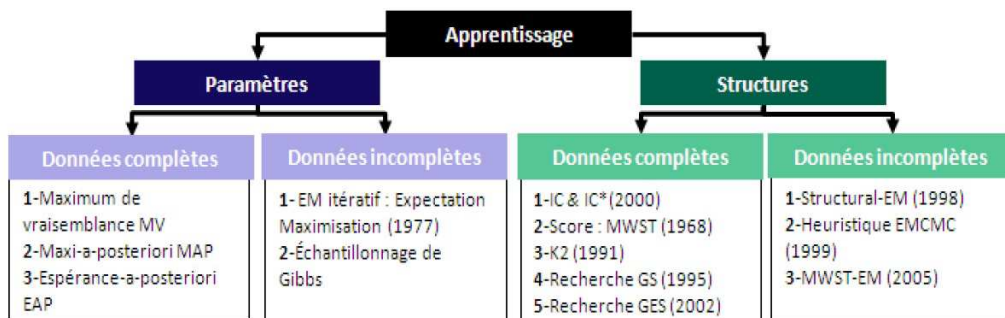


FIGURE 3.6 – Les méthodes d'apprentissage

Notons également que ce type d'approche nécessitant expertise et données pour un apprentissage souvent long ne se prête guère à des modifications fréquentes du contexte de production. Il est important de rappeler que le nombre de structures graphiques apprises croît de manière super-exponentielle en fonction du nombre de variables du réseau (par exemple, pour dix nœuds le nombre de graphes possibles est environ 4×10^{18}) (Robinson, 1997)

Une approche pure Bayésienne dans le contexte dans lequel nous avons placé notre étude est de fait à proscrire. La détermination de la structure graphique du graphe étant du ressort de l'impossible compte tenu de la forte variabilité dans laquelle nous nous sommes placés et des marges temporelles dont nous disposons.

4 Principes et originalité de l'approche proposée

L'approche d'hybridation que nous proposons dans le cadre d'une fonction de diagnostic dynamique en contexte complexe prend pour principe de mixer une approche logique et une approche probabiliste.

L'approche proposée se structure autour de quatre étapes principales :

- (i) Génération en ligne d'un modèle historique étendu à nos besoins (Problème 1 et 2 page 37) permettant jusqu'au moment de la détection d'un compte rendu anormal de présenter un modèle du passage d'un produit sur différents services supportés par des équipements de production. Ce modèle garantit que seuls les services dont les effets n'ont pu être vérifiés en métrologie sont conservés. Ils sont donc naturellement tous suspects.
- (ii) Lors de la détection d'un compte rendu anormal, le modèle est utilisé pour construire la structure graphique d'un modèle Bayésien. Ici réside une première originalité de notre approche, en proposant une démarche innovante d'identification automatique et en temps masqué de la structure d'un réseau Bayésien. Le processus d'identification des variables et de leurs liens de causalité s'en trouve donc ainsi significativement simplifié.
- (iii) Les bases de données historiques (FDC, MES, Maintenance, etc...) sont exploitées pour calculer les tables de probabilités conditionnelles sur la base du modèle graphique ajusté (optimisation des calculs résultante).
- (iv) L'interrogation du modèle Bayésien résultant est menée afin de déterminer les niveaux de suspicion de chacune des causes.

5 Conclusion

Dans ce chapitre, nous avons positionné notre approche en tant que hybridation d'une approche de diagnostic logique et probabiliste. Cette approche devra être en mesure de faire face à un contexte multi-routes de produits, multi-causes et multi-défaillance dans un environnement réactif afin de minimiser les temps d'arrêts de production. Au delà des avantages et inconvénients présentés pour chacune de ces approches nous avons mis en exergue l'intérêt de mener tout d'abord une approche de diagnostic logique, permettant, sur la base d'une simple modélisation du fonctionnement normal, de générer en ligne, un modèle des évolutions passées. Ce modèle est réduit en ligne, en fonction du niveau de captage disponible au sein des équipements. Suite à l'occurrence d'une défaillance, le modèle représentant le passé englobe forcément le "coupable". Afin de localiser ce coupable parmi les coupables potentiels nous suggérons l'usage d'une approche Bayésienne dont le modèle obtenu par diagnostic logique servira de base à la construction de sa structure graphique. La structure ainsi obtenue, en temps masqué, sera ensuite exploitée pour établir les probabilités de pertinence de chacun des coupables potentiels. Ce chapitre marque la fin de cette partie introductive et ouvre à la partie suivante centrée sur la présentation détaillée de notre approche.

Deuxième partie

Approche proposée

Chapitre 4

Extension de la modélisation

1 Introduction

Comme nous avons pu le montrer dans les chapitres précédents, le modèle historique proposé dans (Deschamps, 2007) et (Duong et al., 2012a) pour le diagnostic prend pour hypothèse forte que les équipements sont systématiquement responsables de la propagation de défaillances. Il a été prouvé, en particulier dans le cadre du projet Européen IMPROVE que d'autres causes étaient à considérer comme la cause humaine H ou encore les recettes R ; même si le Produit ne faisait pas partie des causes recherchées dans le modèle initialement proposé, il était cependant modélisé sous la forme de variables d'états. Dans le cadre de ce chapitre, nous proposons d'étendre tout d'abord la modélisation à ces nouveaux facteurs à prendre en compte. Cela nécessite de nouvelles propositions en terme de modélisation (section 1), mais également des apports significatifs dans l'extension des algorithmes initialement proposés tant pour la génération du modèle en ligne que pour son exploitation dans la phase d'isolement des opérations à suspecter. Cette partie algorithmique fait l'objet de la section 2. La section 3 expose quant à elle l'étude du modèle en vue de sa transformation sous la forme d'un graphe causal préfigurant le futur réseau Bayésien.

2 Modèle historique étendu

2.1 Modèle d'opération

Comme annoncé précédemment, le modèle historique doit désormais intégrer d'autres causes que l'équipement ou le produit. Nous proposons ainsi d'intégrer deux nouvelles causes à rechercher, le facteur humain et la recette (Nguyen et al., 2014c). Nous en donnons les définitions suivantes :

Définition 2. *Facteur Humain H_i*

H_i représentera au sein du modèle considéré le facteur humain qui aura pu avoir une influence sur le comportement de l'opération O_i exécutée sur la chaîne fonctionnelle CF_i pendant un intervalle de temps Du_i . Il s'agit ici de se donner les moyens de modéliser et de tracer l'impact d'une action de maintenance par un équipe de maintenance ou d'une action de production par un opérateur d'exploitation. H_i est un ensemble multiple de valeurs $\{H_i^1 \dots H_i^n\}$ correspondant chacune à l'identificateur de ou des opérateurs humains qui sont intervenus.

Définition 3. Facteur Recette R_i

R_i représente la Recette à laquelle appartient l'opération O_i qui sera exécutée sur la chaîne fonctionnelle CF_i pendant un intervalle de temps Du_i . R_i prendra pour valeur l'une des recettes $\{R_i^1 \dots R_i^m\}$ supportées par la chaîne fonctionnel .

Les variables H_i et R_i sont ainsi ajoutées au modèle générique d'une opération (modèle issu rappelons-le de la planification automatique) comme nous pouvons le voir dans la figure 4.1.

Opération O_i		
Chaîne fonctionnelle CF_k	Facteur humain H_i	Recette R_i
Instant de début : t_s		Disponible
Instant de fin: t_f		
Évolution chaîne fonctionnelle ec_j		
Effet Transitoire: EFT(ec_j) IC(EFT(ec_j)))	Pré-condition: Pcd(ec_j)	Pré-contrainte: PCt(ec_j)
Effet final: EfF(ec_j) IC(EfF(ec_j)))	Condition: Cd(ec_j)	Contrainte: Ct(ec_j)
Évolution de flux de produits $ea_{i,1}$		
Effet Transitoire: EFT($ea_{i,1}$) IC(EFT($ea_{i,1}$)))	Pré-condition: Pcd($ea_{i,1}$)	Pré-contrainte: PCt($ea_{i,1}$)
Effet final: EfF($ea_{i,1}$) IC(EfF($ea_{i,1}$)))	Condition: Cd($ea_{i,1}$)	Contrainte: Ct($ea_{i,1}$)

FIGURE 4.1 – Modèle d'opération étendu à la représentation du facteur humain et recettes

Ici, il est à noter que :

- i correspond au numéro de l'opération offerte par la chaîne fonctionnelle k ,
- j correspond au numéro de l'évolution du flux de produits d'une opération i .

Les bases du modèle étant désormais étendues à nos besoins, nous reprenons dans la section suivante les principes fondamentaux nécessaires à la compréhension des algorithmes qui seront présentés plus loin.

2.2 Modèles d'opérations et évolutions

Le concept d'opération, qu'il soit étendu ou non, permet de modéliser quatre catégories opératoires distinctes qui peuvent avoir un effet sur la chaîne fonctionnelle elle-même et/ou sur le produit. Ces quatre catégories sont à la base des algorithmes d'exploitations, que ce soient à des fins de génération de commandes (Henry et al., 2012) ou de diagnostic (Duong et al., 2012a). Nous nous proposons d'en synthétiser ici une présentation sous la forme des deux figures 4.2 et 4.3 ci-jointes reprises de (Henry, 2005) :

Cette analyse permet de caractériser les relations de causalité entre les grandeurs de commande et les évolutions. Il en résulte huit situations, de S1 à S8, qui sont couvertes par quatre catégories d'opérations : opérations d'action, opérations d'information, évolutions induites et évolutions requises. Les critères à la base de ce regroupement (8 situations, 4 éléments) sont d'une part la capacité du module coordination à déclencher une opération, et d'autre part à la maîtriser.

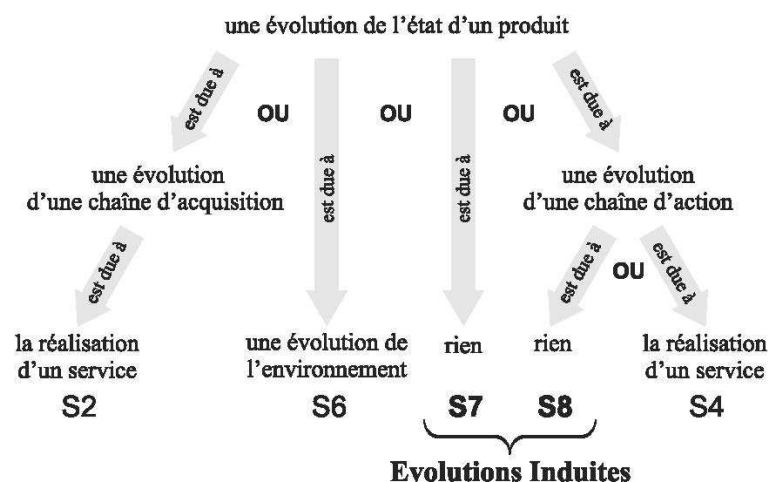


FIGURE 4.2 – Origine d'une évolution de l'état d'un produit.

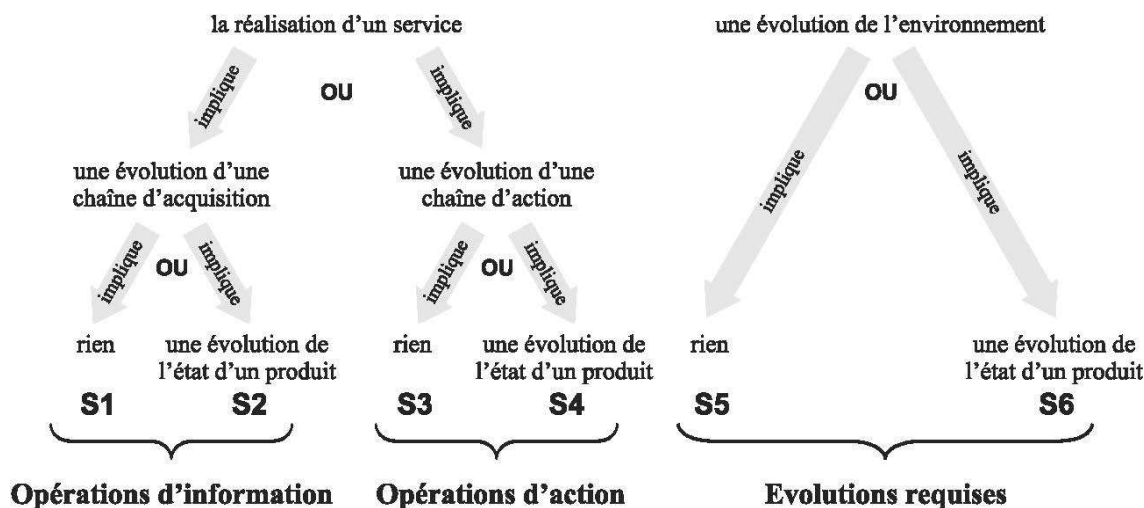


FIGURE 4.3 – Effets possibles résultants de la réalisation d'un service ou d'une évolution de l'environnement.

Ainsi, une opération d'action couvre les situations (S3 et S4), déclenchées et maîtrisées, qui sont basées sur une chaîne d'action. Les deux autres situations (S1 et S2), contrôlées et maîtrisées, basées sur une chaîne d'acquisition sont couvertes par les opérations d'information. Une évolution induite (S7 et S8) correspond à une situation non déclenchée mais maîtrisée puisque le module de coordination pilote la chaîne fonctionnelle à l'origine d'une telle évolution. Et enfin, une évolution requise (S5 et S6) est une situation non déclenchée et non maîtrisée, puisque la chaîne fonctionnelle à l'origine d'une telle évolution n'est pas pilotée par le module de coordination.

Sur la base de ces catégories d'opérations, une seconde analyse est nécessaire afin de mettre en relation la nature d'un comportement d'une opération et les trois sous-ensembles des caractéristiques des effets : "l'état physique du flux de produit", "la position spatiale du flux de produit" et "l'état des chaînes fonctionnelles". Cette analyse est retranscrite dans le tableau de la figure 4.4.

Une opération de transformation	Une opération de transitique	Une opération de préparation	Effets sur les chaînes fonctionnelles et/ou sur le flux de produit
modifie	ne peut pas modifier	ne peut pas modifier	L'état physique du flux de produit (géométrique, état de surface, etc).
peut modifier	modifie	ne peut pas modifier	La position spatiale du flux de produit.
peut modifier	peut modifier	modifie	L'état des chaînes fonctionnelles.

FIGURE 4.4 – Les comportements d'une d'opération sur les variables d'état du système

L'analyse de ce tableau conduit naturellement à définir les différents comportements que peut avoir chaque catégorie d'opérations :

- une opération d'action ou d'information peut avoir un comportement de transformation, un comportement de transitique ou un comportement de préparation ;
- une évolution induite peut avoir un comportement de transformation ou un comportement de transitique ;
- une évolution requise peut avoir un comportement de transformation, un comportement de transitique ou un comportement de préparation.

Comme nous allons le voir par la suite, les algorithmes que nous proposons, que se soient pour la génération en ligne du modèle ou pour son exploitation à des fins de localisation s'appuient fortement sur cette structuration.

La structuration des opérations étant désormais présentée, il nous reste à préciser les notations que nous avons retenues et reprises de (Deschamps, 2007) pour caractériser les évolutions et donc les effets des opérations sur la chaîne fonctionnelle et le produit.

Effet transitoire de l'état d'une chaîne fonctionnelle $EfT(ec_i)$ correspondant à l'évolution ec_i de l'opération Oa_i . Il est obtenu si la chaîne fonctionnelle est en fonctionnement normal $\lceil AN(CF_k(2x - 1))$ au lancement du service, où x est la x^{ime} opération exécutée par la chaîne fonctionnelle CF_k ; la pré-contrainte $PCT(ec_i)$ et la pré-condition $PCd(ec_i)$ sont respectées ; les pré-contraintes $PCT(ea_{i,j})$ des $ea_{i,j}$ sont réalisées et les pré-conditions $PCd(ea_{i,j})$ sont vraies. L'effet $EfT(ec_i)$ est défini tel quel :

Définition 4. *Effet transitoire de l'évolution de la chaîne fonctionnelle*

$$\lceil AN(CF_k(2x - 1)) \wedge PCd(ec_i) \wedge PCT(ec_i) \bigwedge_{\forall j \in [1, N_i]} [PCd(ea_{i,j}) \Rightarrow PCT(ea_{i,j})] \Leftrightarrow EfT(ec_i)$$

Effet final de l'état de la chaîne fonctionnelle $EfT(ec_i)$ correspondant à l'évolution ec_i de l'opération Oa_i . Il est obtenu si la chaîne fonctionnelle est en fonctionnement normal $\lceil AN(CF_k(2x))$ durant l'exécution du service ; l'effet transitoire $Eft(ec_i)$ est réalisé ; la contrainte $Ct(ec_i)$ et la condition $Cd(ec_i)$ sont respectées ; les contraintes $Ct(ea_{i,j})$ sont réalisées et les pré-conditions $PCd(ea_{i,j})$ sont vraies. L'effet $EfT(ec_i)$ est défini tel quel :

Définition 5. *Effet final de l'évolution de la chaîne fonctionnelle*

$$\lceil AN(CF_k(2x)) \wedge EfT(ec_i) \wedge Cd(ec_i) \wedge Ct(ec_i) \bigwedge_{\forall j \in [1, N_i]} [PCd(ea_{i,j}) \Rightarrow Ct(ea_{i,j})] \Leftrightarrow EfF(ec_i)$$

Effet transitoire sur l'état du flux de produits $EfT(ea_{i,j})$ pour $j \in [1, N_i]$ correspondant à l'évolution $ea_{i,j}$ de l'opération Oa_i . Il est obtenu si la chaîne fonctionnelle est en fonctionnement normal $\neg AN(CF_k(2x - 1))$ au lancement du service, la pré-contrainte $PCT(ec_i)$ et la pré-condition $PCd(ec_i)$ sont respectées ; les pré-contraintes $PCT(ea_{i,l})$ des $ea_{i,l}$ sont réalisées et les pré-conditions $PCd(ea_{i,l})$ sont vraies pour $l \in [1, N_i] \setminus j$. L'effet $EfT(ea_{i,j})$ est défini tel que :

Définition 6. *Effet transitoire de l'évolution du flux de produits*

$$[\neg AN(CF_k(2x - 1)) \wedge PCd(ea_{i,j}) \wedge PCd(ea_{i,j}) \wedge PCT(ea_{i,j}) \wedge PCd(ec_i) \wedge PCT(ec_i) \bigwedge_{\forall l \in [1, N_i] \setminus j} [PCd(ea_{i,l}) \Rightarrow PCT(ea_{i,l})]] \Leftrightarrow EfT(ea_{i,j})$$

Effet final sur l'état du flux de produits $EfF(ec_i)$ correspondant à l'évolution $ea_{i,j}$ de l'opération Oa_i . Il est obtenu si la chaîne fonctionnelle est en fonctionnement normal $\neg AN(CF_k(2x))$ durant l'exécution du service ; l'effet transitoire $EfT(ea_{i,j})$ est réalisé ; la contrainte $Ct(ec_i)$ et la condition $Cd(ec_i)$ sont respectées ; la contrainte $Ct(ea_{i,j})$ et la condition $Cd(ea_{i,j})$ sont respectées ; les contraintes $Ct(ea_{i,l})$ sont réalisées et les pré-conditions $PCd(ea_{i,j})$ sont vraies pour $l \in [1, N_i] \setminus j$. Cet effet $EfF(ea_{i,j})$ est défini tel que :

Définition 7. *Effet final de l'évolution du flux de produits*

$$[\neg AN(CF_k(2x)) \wedge EfT(ea_{i,j}) \wedge Cd(ea_{i,j}) \wedge Ct(ea_{i,j}) \wedge Cd(ec_i) \wedge Ct(ec_i) \bigwedge_{\forall l \in [1, N_i] \setminus j} [PCd(ea_{i,l}) \Rightarrow Ct(ea_{i,l})]] \Leftrightarrow EfF(ea_{i,j})$$

Ici, il est à noter que d'un point de vue dynamique :

- x correspond au numéro d'exécution des opérations de la chaîne fonctionnelle k ,
- $\neg AN(CF_k(2x - 1))$ correspond au fonctionnement normal de la chaîne fonctionnelle k au début de l'opération numéro x ,
- $\neg AN(CF_k(2x))$ correspond au fonctionnement normal de la chaîne fonctionnelle k durant l'exécution de l'opération numéro x ,
- $J_{k,x}$ correspond à la liste des indices des évolutions attendues de l'opération numéro x exécutée par la chaîne fonctionnelle k .

Ceci étant présenté, nous nous proposons dans la section suivante de découvrir les algorithmes de génération en ligne du modèle historique que nous proposons.

3 Algorithmes de génération en ligne du modèle historique étendu

Comme nous l'avons vu plus haut, le principe général de la génération en ligne de modèle historique consiste à chaque événement de production (un ordre lancé vers les chaînes fonctionnelles ou un compte rendu d'exécution remontant vers le module de coordination) de mettre à jour et suivre l'activité lancée au sein d'un modèle. Pour ce faire, et à chacune des occurrences de ces événements, il sera nécessaire de sélectionner "la bonne fiche opératoire" dans la base de fiches, d'instancier au cœur du modèle historique les entrées (Pré-conditions et Pré-contraintes de l'opération considérée) puis à la réception de du compte rendu les

sorties (effets intermédiaires et finaux). Ceci est répété naturellement à chaque occurrence d'événements. Deux algorithmes, notés 1 et 2, détaillés par la suite, permettent ainsi de générer en ligne le modèle.

Au fur et à mesure de la construction du modèle en ligne, ce dernier doit être réduit des opérations qui ont été certifiées vérifiées suite à opération de métrologie. Il s'agit ici, lorsqu'une opération de surveillance (métrologie) a été lancée de vérifier si le compte rendu correspondant permet ou non de valider l'effet escompté. Si tel est le cas, et comme proposé dans (Deschamps, 2007), un mécanisme de réduction du modèle est lancé. Cependant, comme nous l'avons souligné dans le chapitre 3 de ce mémoire, il subsiste un problème à ce jour non résolu, celui de la prise en compte d'opérations simultanées interdépendantes, l'une étant le pré-requis de l'autre. Afin de résoudre ce problème, nous avons proposé trois autres algorithmes (notés 3, 4 et 5) qui permettent de tenir compte de ces spécificités en présence d'opérations simultanées. Enfin, l'algorithme 6 permet sur cette base de réduire effectivement le modèle.

La démarche générale étant posée, les sections suivantes se proposent de détailler chacun de ces algorithmes.

3.1 Génération dynamique du modèle historique

De manière générale, l'algorithme de génération en ligne du modèle historique étendu reprend dans les grandes lignes celui proposé par (Deschamps, 2007). Il s'articule autour des points fondamentaux suivants et est formalisé dans le détail par les logigrammes des figures 4.5 et 4.6.

- Relever et instancier l'instant de démarrage t_s (respectivement de fin t_f) de l'opération.
- Instancier les pré-requis ((pré)conditions $PCd(ec_i)$, $PCd(ea_{i,j})$ et les pré-contraintes $PCT(ec_i)$, $PCT(ea_{i,j})$).
- Instancier les informations relatives aux équipes de maintenance, à la recette et au produit des variables $\{H_i, R_i, P_i\}$, si les informations sont disponibles.
- Déterminer les évolutions attendues $j \in J$ et non souhaitées $j \notin J$ afin de formaliser les évolutions d'état des chaînes fonctionnelles et/ou du flux de produits.
- Rechercher les effets finaux EfF des opérations qui ont été exécutés par la chaîne fonctionnelle CF_k .
- Rechercher les effets transitoires $\lceil EfT$ et les effets finaux $\lceil EfF$ correspondant à l'exécution de l'opération O_i à travers le flux de produits.
- Mettre à jour les variables d'état de l'opération et du flux de produits en fonction des effets réalisés (cf. compte rendu d'exécution).

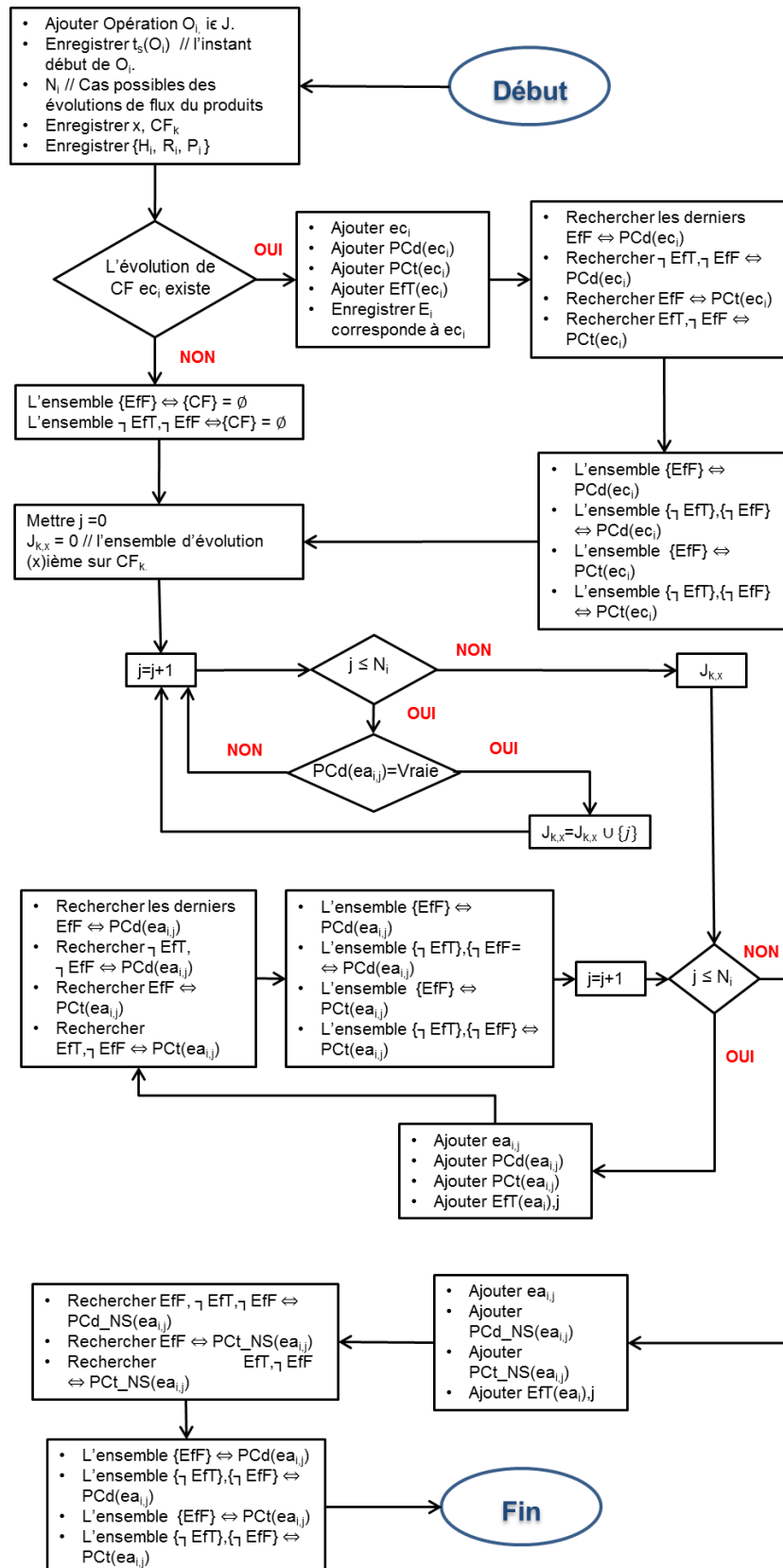


FIGURE 4.5 – **Algorithme 1** : Génération des informations de début d'opération

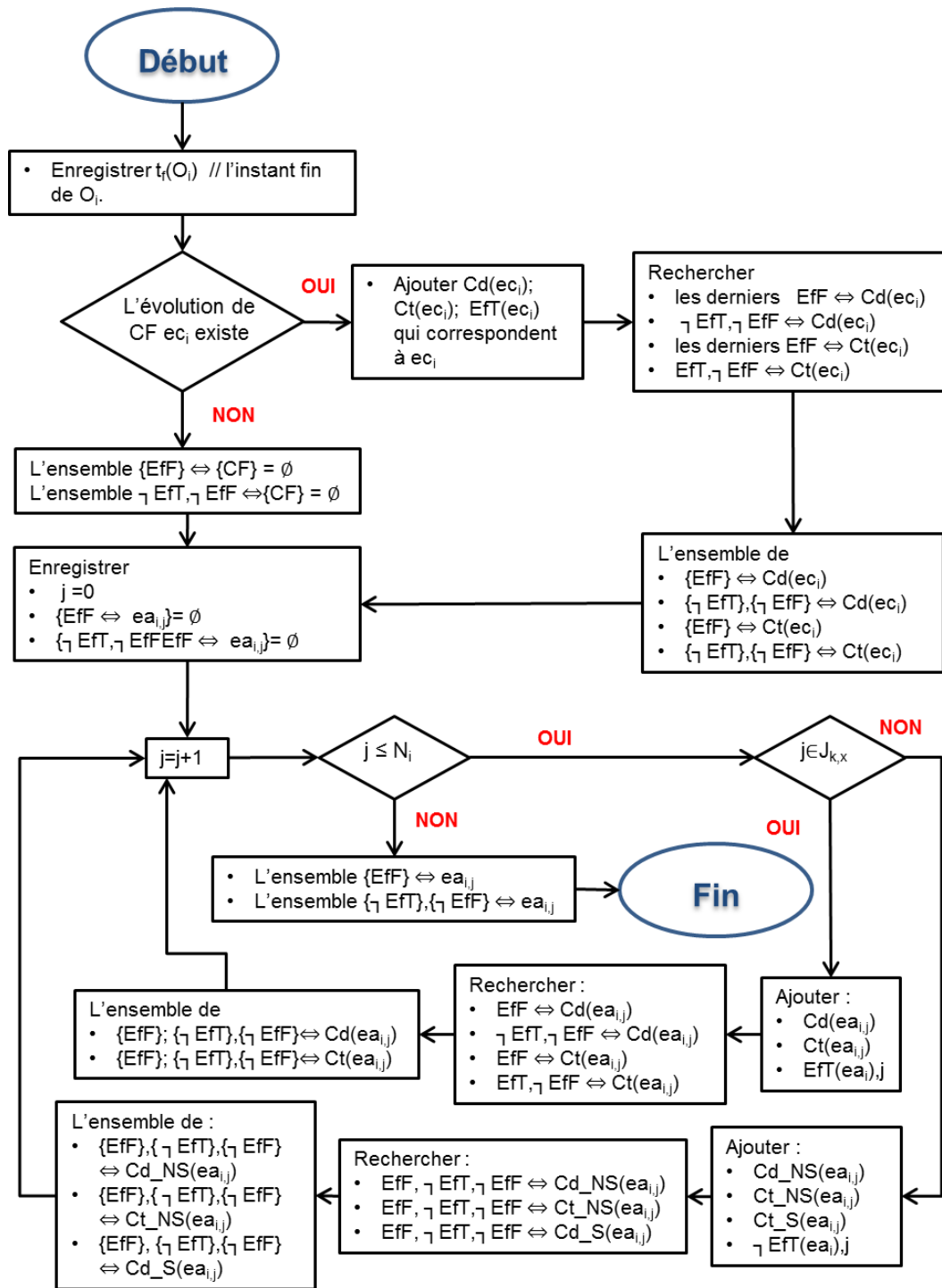


FIGURE 4.6 – **Algorithme 2** : Génération des informations de fin d’opération

A titre d’exemple, nous avons représenté dans la figure 4.7 le cas d’une opération d’extension de la tige d’un vérin au moment de son lancement.

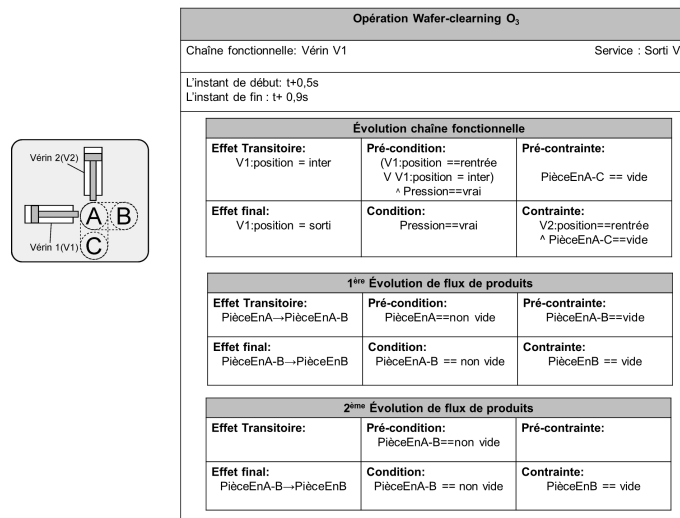


FIGURE 4.7 – Description du service sortir V1

3.2 Gestion des opérations simultanées avant réduction

Afin de résoudre le problème posé par les opérations simultanées interdépendantes, et ainsi éviter à terme, tant pour l'algorithme de réduction que pour celui de propagation des écueils de non convergence, nous proposons au sein de cette section trois algorithmes pour non seulement repérer dans le modèle historique ces singularités mais également les résoudre.

Dans un premier temps, nous allons rechercher dans le modèle obtenu, en ligne et donc pas à pas, les opérations qui présentent cette particularité ; nous allons mettre en place un mécanisme de recherche, à partir de la dernière opération O_i enregistrée dans le modèle historique, non seulement de l'opération O_j dont les effets sont les pré-requis de cette dernière opération O_i exécutée (cf. figure 4.8) mais également dont au même instant les effets de la dernière opération O_i enregistrée sont les pré-requis de l'opération O_j . Sur cette base, les algorithmes que nous proposons ici sont séparés selon trois phases. La phase 1 (Algorithme 3) recherche l'ensemble des opérations avant correspondant aux évolutions de la chaîne fonctionnel ec_i ou de flux de produits $ea_{i,j}$ ou des variables de surveillance $Surv_{i,j}$. Cet ensemble est considéré comme un ensemble des parents de l'opération O_i . La phase 2 (Algorithme 4) recherche les opérations arrières de l'opération O_i , elles sont considérées comme un ensemble des fils de l'opération O_i . La phase 3 (Algorithme 5) se base sur le recherche des intersections entre l'ensemble des parents et des fils de l'opération O_i pour nous fournir un ensemble des opérations simultanées correspondant à O_i .

L'algorithme 3 se structure autour des points suivants :

- Ajouter l'évolution de chaîne fonctionnelle ec_i et les pré-conditions, pré-contraintes et effets transitoires correspondant à ec_i .
- Rechercher les derniers effets EfF , $\lceil EfT$ et $\lceil EfF$ qui sont qualifiés de correct avec l'information de début d'opération correspondant à la chaîne fonctionnelle.
- Pour chaque dernier effet correspondant à ec_i , rechercher toutes opérations qui utilisent ce pré-requis. Nous appelons OP_1 l'ensemble de ces opérations.
- Ajouter l'évolution du flux de produits $ea_{i,j}$. Rechercher tous les effets EfF , $\lceil EfT$ et $\lceil EfF$ qui correspondent à l'évolution $ea_{i,j}$.

- Pour chaque dernier effet correspondant à $ea_{i,j}$, rechercher toutes opérations qui utilisent ce pré-requis. Nous appelons OP_2 l'ensemble de ces opérations.
- Ajouter les variables de surveillance $\{Surv_{i,j'}\}$ pour $j' \in J \setminus \{j\}$.
- Enregistrer l'opération O_j si $IC(Surv_{i,j}) = 1$. Nous appelons OP_3 l'ensemble de ces opérations.
- Générer l'ensemble $OP(O_i) = OP_1 \cup OP_2 \cup OP_3$ comme étant l'ensemble des opérations précédentes de l'opération O_i pour $i \in J$.

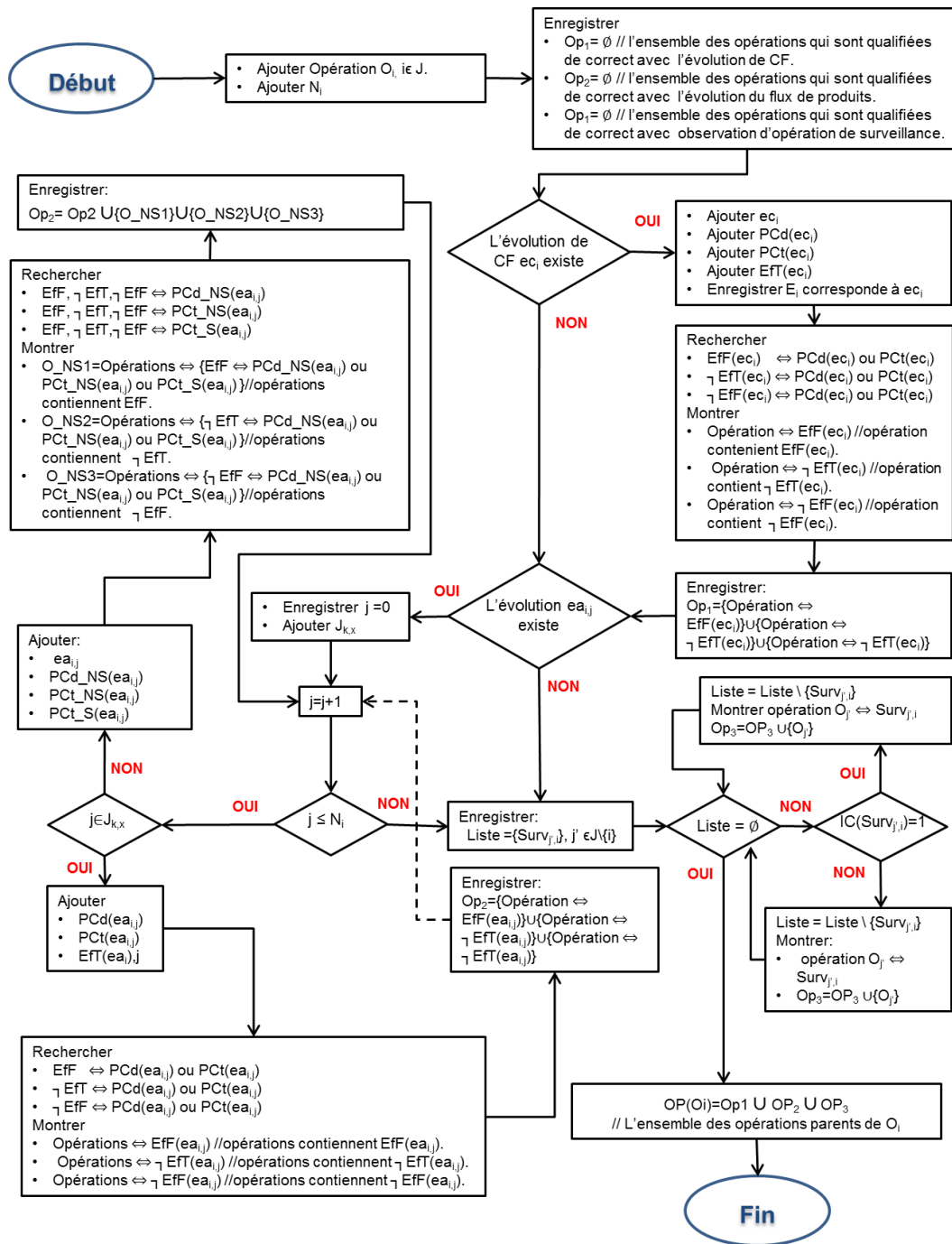


FIGURE 4.8 – **Algorithme 3** : Détermination des opérations simultanées phase 1

De la même manière, l'algorithme 4 illustré dans la figure 4.9 est proposé pour déterminer un l'ensemble OF des opérations j utilisant comme pré-requis les effets de l'opération i :

- Rechercher les $EfT(m)$ et $EfF(n)$ pour $m \in M, n \in N$.
- Rechercher tous les pré-requis portant sur $EfT(m)$ et/ou $EfF(n)$.
- Rechercher toutes les variables de surveillance $Surv_{j,i}$ relatives à $EfT(m)$ et/ou $EfF(n)$ dont $IC(Surv_{j,i}) = 1$.
- Rechercher l'ensemble $OF1$ d'opérations qui sont connectées en amont à tous les pré-requis obtenus.
- Rechercher l'ensemble $OF2$ d'opérations qui sont connectées aux variables de surveillance $Surv_{j,i}$.
- Fournir l'ensemble $OF(O_i) = OF1 \cup OF2$ contenant les opérations ultérieures de l'opération O_i .

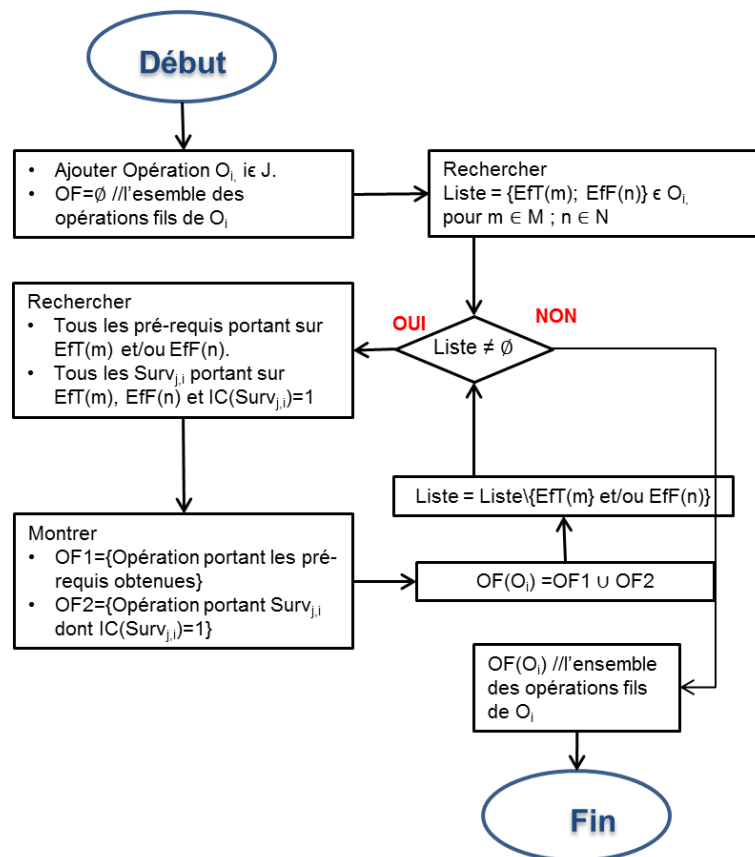


FIGURE 4.9 – **Algorithme 4** : Détermination des opérations simultanées phase 2

Une fois ce premier tri réalisé, nous pouvons lancer l'algorithme 3 qui permet de déterminer l'ensemble des opérations qui sont lancées en parallèle de l'opération O_i et qui sont interdépendantes. Selon la définition des opérations simultanées, voir page 37, l'opération O_j impacte l'opération O_i et vice-versa. L'ensemble des opérations simultanées avec $\{O_i\}$ peut donc se caractériser ainsi : $RP(O_i) = OP(O_i) \cap OF(O_i)$. Si nous intégrons O_i à cet ensemble pour exprimer toutes les opérations simultanées, nous obtenons $J_{RP,i} = RP(O_i) \cap \{O_i\}$.

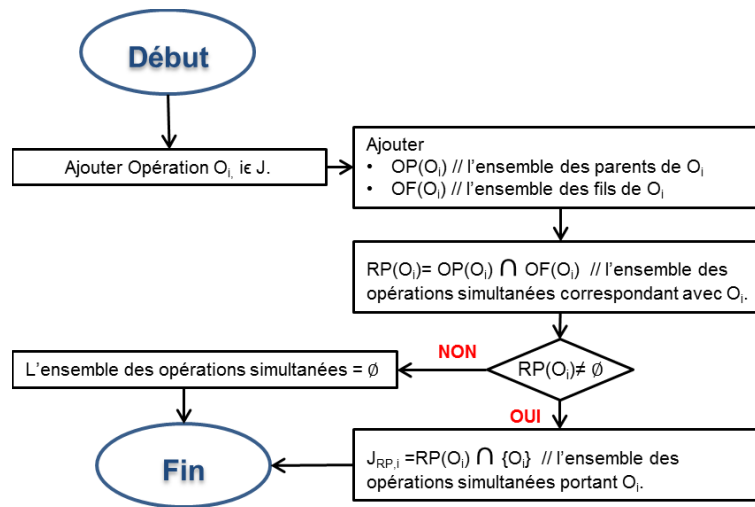


FIGURE 4.10 – **Algorithme 5** : Détermination des opérations simultanées d’une opération

Le modèle historique étant désormais doté de l’ensemble des algorithmes permettant sa génération, nous nous proposons désormais de présenter le mécanisme de réduction proposé dans (Deschamps, 2007), étendu à la gestion des opérations simultanées.

3.3 Mécanisme de réduction dynamique de la taille du modèle

De manière synthétique, le mécanisme d’oubli et de réduction du modèle pour le diagnostic se base sur un indice de confiance associé aux effets observés des opérations. Cet indice de confiance ($IC(Surv_{i,j})$) est calculé en ligne sur la base non seulement de la connaissance sur le niveau de captage des chaînes fonctionnelles associées mais également sur une approche probabiliste basée sur un apprentissage à base de données (Duong et al., 2012b). Lorsque cet indice de confiance, qui peut varier en 0 et 1, se voit attribuer le score de 1 ($IC = 1$, c’est le cas notamment lorsqu’il y a correspondance entre le service demandé par rapport à l’observation directe), alors l’effet obtenu est caractérisé de correct. Cette « confiance » est alors propagée en arrière dans le modèle (i.e. parcours de graphe) afin de valider les variables d’état qui ont conduit à ce succès. Toute opération dont les variables d’entrées et de sorties sont marquées d’un score de 1 sont alors "oubliées" et donc effacées du modèle.

D’un point de vue plus technique, les règles de réduction utilisées sont directement issues des définitions des comportements des opérations définies dans le début du chapitre. Suite à la qualification correcte de l’effet d’une opération (transitoire ou final), le modèle de son comportement décrit les pré-requis (pré-conditions et/ou conditions et/ou pré-contraintes et/ou contraintes) qui devaient être nécessairement respectés. Ces informations seront donc qualifiées de correctes. Ensuite, ces informations sont connectées également à des effets (souhaités ou non souhaités) antérieurs d’opérations. Ainsi l’utilisation de leurs modèles de comportement permet encore une fois de qualifier de corrects les pré-requis pour obtenir les effets, et ainsi de suite jusqu’à ce que tout le modèle soit parcouru.

L’algorithme s’articule autour des points suivants et est détaillé dans la figure 4.11 :

- Ajouter à la liste les effets transitoires EfT et les effets finaux EfF qui sont liés à un IC

- de 1.
- Rechercher l'opération O_i contenant EfT et/ou EfF .
 - Ajouter l'ensemble d'opérations simultanées $J_{RP,i}$ de l'opération O_i . $\forall j, j' \in J_{RP,i}, j \neq j'$ rechercher les connexions amont des pré-requis de O_j à des effets EfT , EfF , $\neg EfT$ et $\neg EfF$ de l'opération O'_j . Nous notons l'ensemble des effets transitoires simultanés $\{EfT\}_{simultané}$ et l'ensemble des effets finaux simultanés $\{EfF\}_{simultané}$.
 - Rechercher les pré-requis de l'effet le plus récent contenu dans Liste et retirer l'effet de la Liste.
 - Rechercher les connexions amont des pré-requis aux effets EfT ou EfF .
 - Ajouter les EfT ou EfF s'ils n'appartiennent pas à l'ensemble des effets transitoires simultanés $\{EfT\}_{simultané}$ ou à l'ensemble des effets finaux simultanés $\{EfF\}_{simultané}$.
 - Enregistrer l'ensemble des pré-requis, l'ensemble des EfT , EfF , $\neg EfT$ et $\neg EfF$ qui sont connectés avec $IC = 1$.

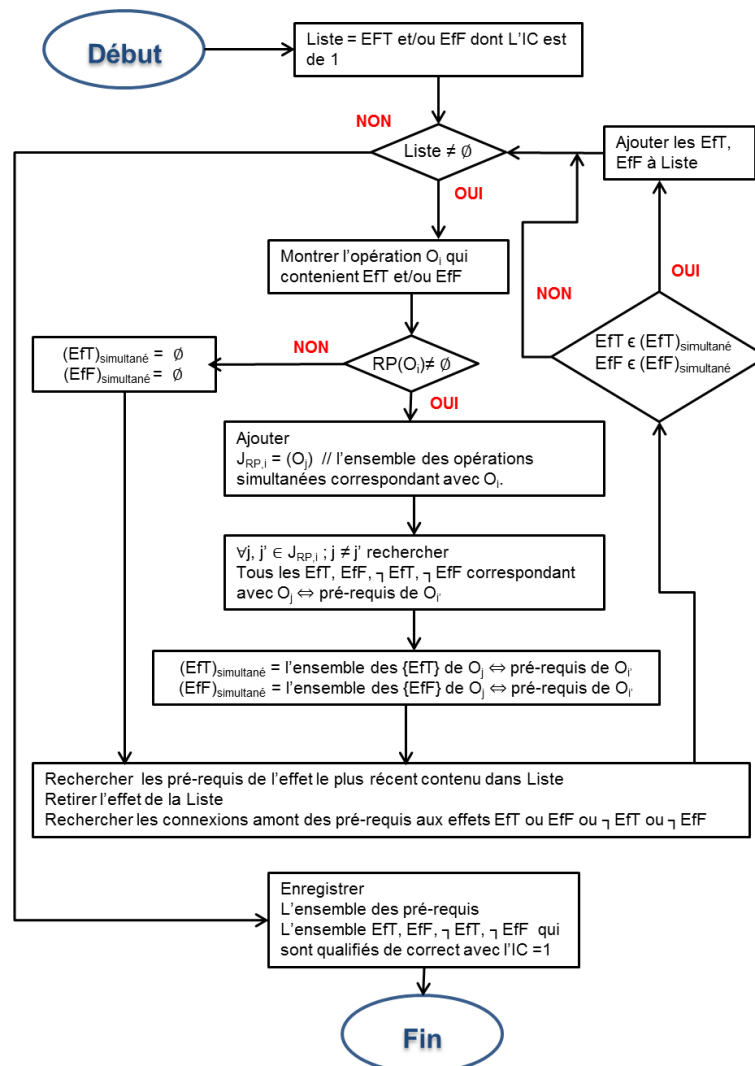


FIGURE 4.11 – Algorithme 6 : Réduction du modèle

4 Le mécanisme de la propagation de la suspicion

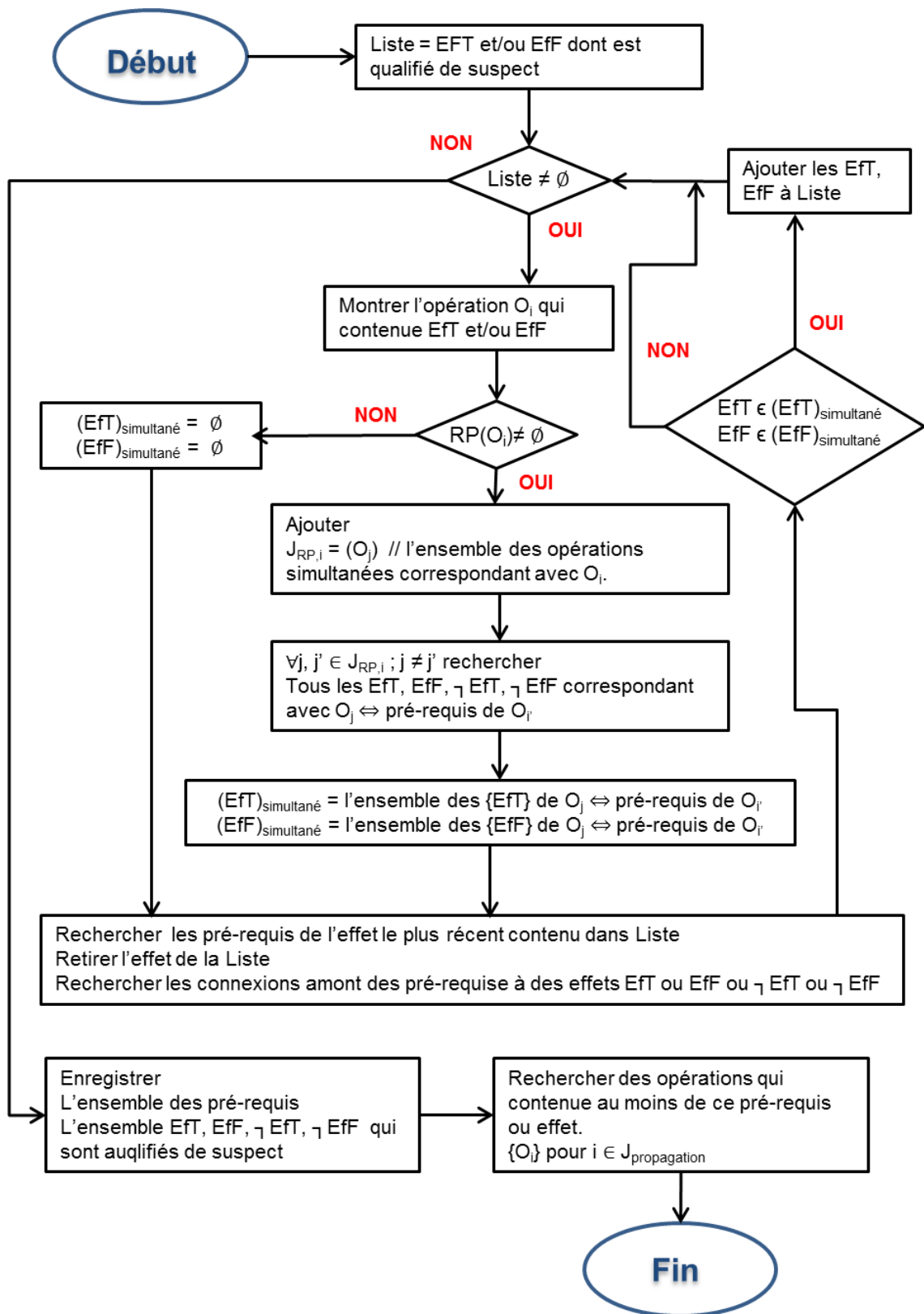
La démarche utilisée pour le diagnostic des capacités opératoires est tout à fait similaire à la recherche des conflits minimaux faite dans le domaine du diagnostic logique (Dubuisson, 2001). Un conflit minimal correspond à un ensemble de composants dont au moins un est défaillant pour expliquer une observation donnée. Nous recherchons ici la somme des conflits minimaux, correspondant à l'ensemble des opérations à suspecter.

L'obtention des conflits minimaux en diagnostic logique est assurée par un mécanisme de propagation de contraintes. Pour que la sortie d'un composant considéré soit cohérente avec ce qui est attendu, il faut que ce composant ne soit pas défaillant mais également que les composants permettant d'appliquer les entrées du composant considéré ne soient pas non plus défaillantes. Il est donc possible pour chaque sortie de composant de savoir suite à une incohérence quels sont les composants potentiellement responsables (conflit minimum).

Cependant dans le contexte dans lequel nos travaux se placent, le problème est plus complexe. La mauvaise réalisation d'un effet n'entraîne pas la remise en cause de l'ensemble des pré-requis de la réalisation de l'opération. Prenons l'exemple de la suspicion d'un effet sur une chaîne fonctionnelle finale $EfT(ec_i)$. Dans ce cas, il n'est pas nécessaire de suspecter les pré-contraintes $Pct(ea_{i,j})$ des évolutions attendues du flux de produits, puisque leur non respect n'aurait aucune conséquence sur cet effet. Une des difficultés supplémentaires réside dans la nécessité non seulement de connaître les opérations potentiellement responsables de la réception d'un CRA, mais également leurs conséquences possibles sur l'exécution d'autres opérations (pronostic, problématique non traitée dans ce mémoire). Cette recherche s'appuiera sur les modèles des comportements des opérations exécutées.

La recherche des origines d'un compte rendu anormal est ainsi assurée par une propagation arrière. Partant du compte rendu, il s'agit de remonter aux origines possibles soit en terme d'exécutions non conformes des opérations soit en terme d'évolutions non attendues de l'état des chaînes fonctionnelles ou du flux de produits.

L'algorithme, intégrant désormais la prise en compte des opérations simultanées et donc des problématiques de convergence, est détaillé dans la figure 4.12. Un exemple est donné dans la figure 4.13 permettant d'illustrer ce mécanisme de propagation de suspicion au sein d'un extrait de modèle d'historique.

FIGURE 4.12 – **Algorithme 7** : Propagation arrière via les évolutions d'opérations

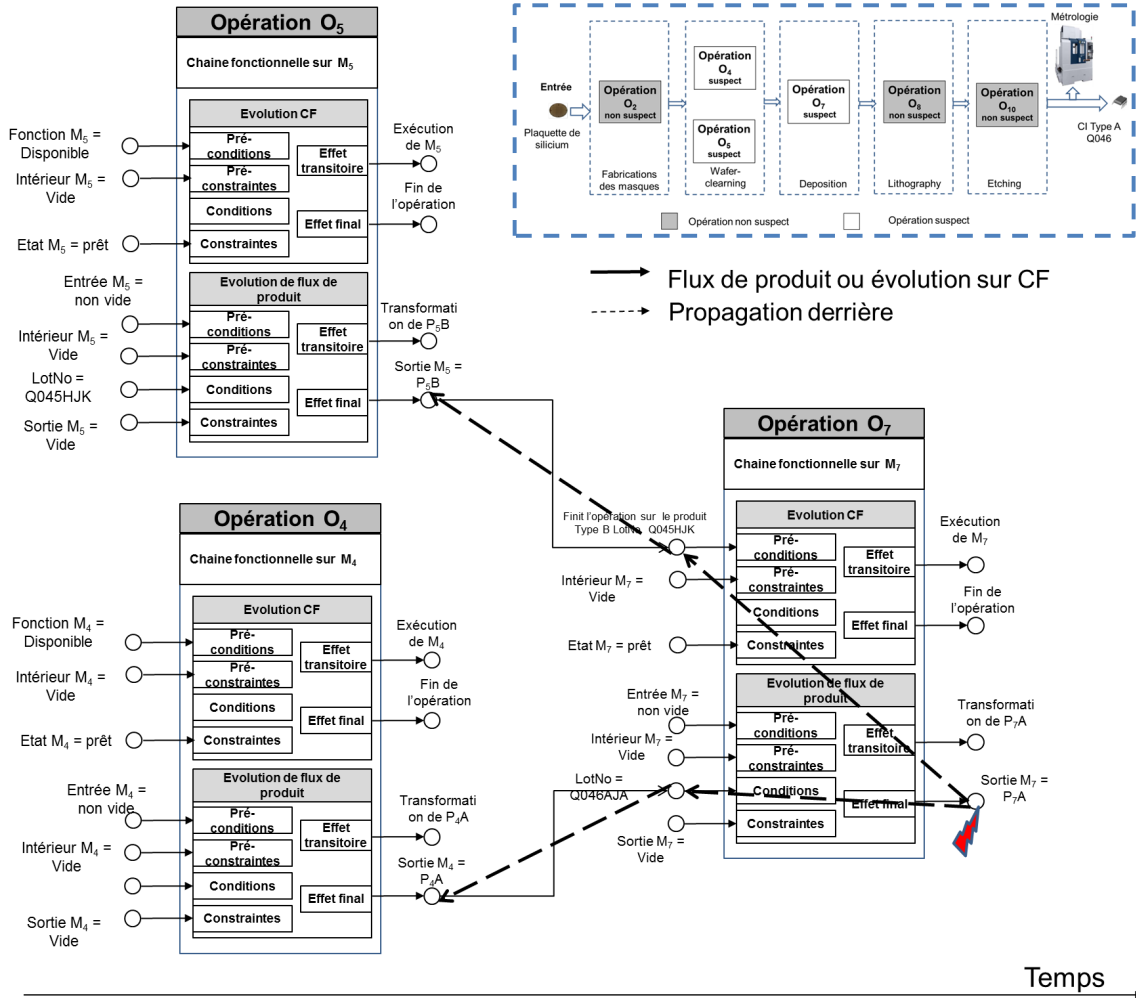


FIGURE 4.13 – Exemple de propagation de suspicion

5 Propriétés du modèle historique étendu

Comme nous l'avons annoncé en tête de ce chapitre, une des originalités de notre approche réside dans l'exploitation du modèle historique réduit aux opérations suspectes à des fins de construction de la structure graphique d'un réseau Bayésien. Cette proposition se place donc dans le domaine de la transformation de modèles. Aussi, est-il pertinent d'analyser les propriétés du modèle que nous cherchons à transformer. C'est tout l'objectif de cette section.

5.1 Graphe causal orienté

Le modèle historique réduit aux opérations suspectes est constitué d'un ensemble d'opérations partiellement ou totalement liées entre elles par des relations de causes à effets ; les effets d'une opération étant les pré-requis d'une ou plusieurs autres. Dans le contexte dans lequel nous évoluons, cet ensemble d'opérations reliées entre elles constitue un graphe causal qui n'est ni plus ni moins qu'un arbre de défaillances comme l'illustré par la figure 4.14.

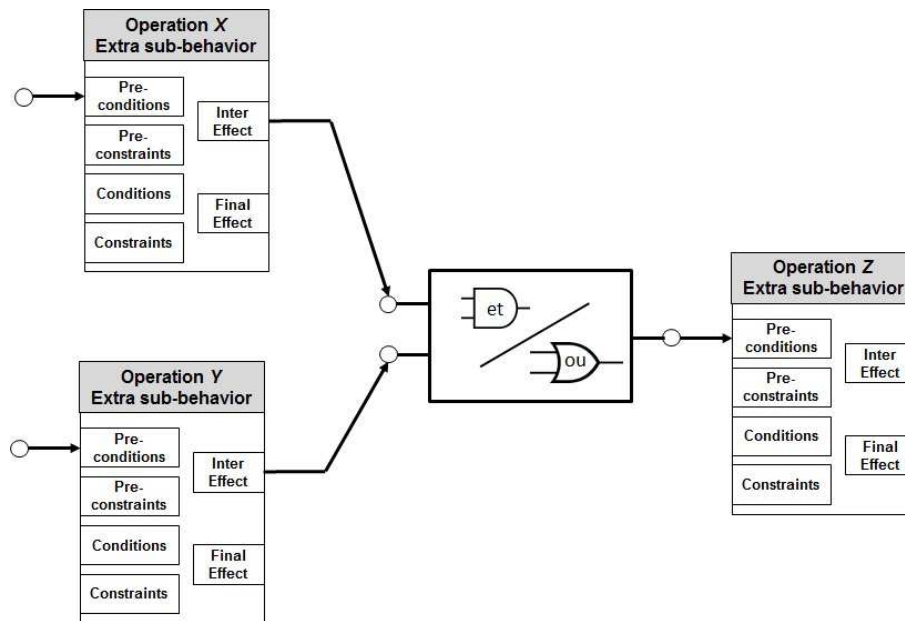


FIGURE 4.14 – Relations entre les opérations

Propriété 1. La structure du modèle d'historique réduit à ses opérations suspectes est un arbre orienté des défaillances.

5.2 Nœuds de Convergence

Un processus de production peut présenter des singularités telles que des machines ou équipement dits "goulots" (concentrateur de plusieurs flux de produits) tel que représenté dans la figure 4.15. Les modèles générés représentant l'enchaînement des opérations et la traçabilité du flux de produit, des opérations où convergent des flux de produits pourront naturellement y être modélisées. Des nœuds de convergence sont donc à considérer dans un tel graphe.

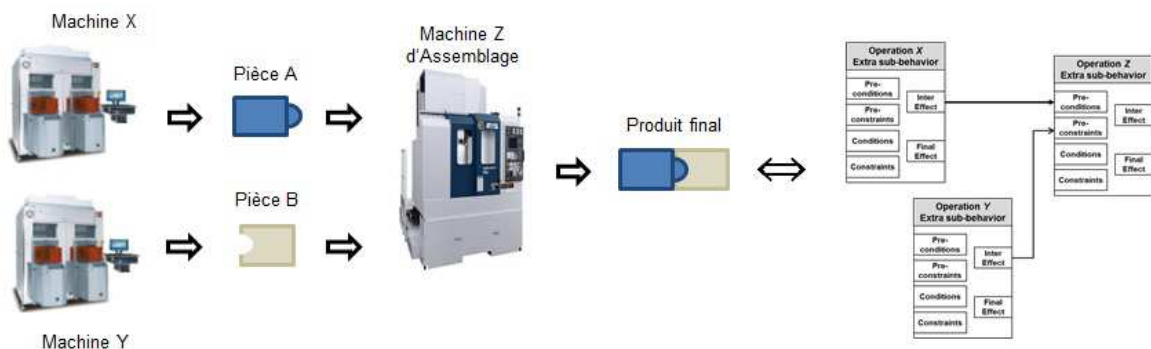


FIGURE 4.15 – Nœud convergent

Propriété 2. Le modèle d'historique réduit à ses opérations suspectes peut contenir des nœuds de convergence.

5.3 Divergence

A l'inverse, et selon le même raisonnement fait précédemment, des opérations peuvent caractériser des phénomènes de divergence. C'est notamment le cas d'une machine capable de traiter plusieurs produits à la fois, produits qui une fois transformés peuvent être orientés vers des équipements différents pour subir des transformations elles aussi différentes tel que montré dans la figure 4.16.

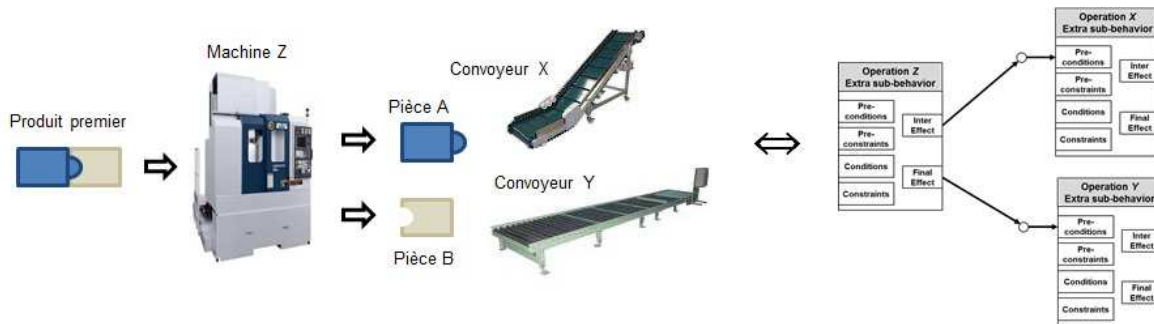


FIGURE 4.16 – Nœud divergent

Propriété 3. *Le modèle d'historique réduit à ses opérations suspectes peut contenir des nœuds exprimant une divergence.*

5.4 Interactions simultanées

Comme nous l'avons déjà discuté, certaines transformations de produits peuvent nécessiter la collaboration de plusieurs équipements simultanément. Le produit ou les produits manipulés étant dans ce cas précis considérés comme des ressources partagées, les pré-requis, que ce soient les pré-conditions (présence du produit) et les pré-contraintes (positions initiales des bras de deux robots articulés collaboratifs) de l'un sont de fait les pré-requis de l'autre... et vice-versa (cf. figure 4.17).

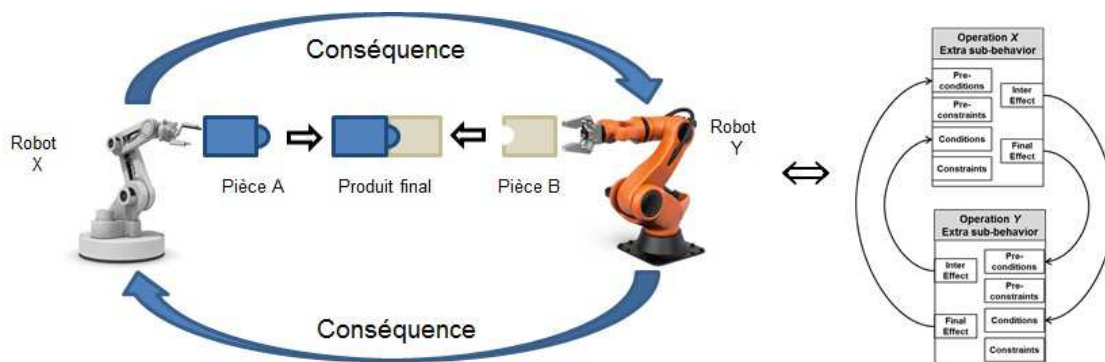


FIGURE 4.17 – Opération simultanée

Propriété 4. *Le modèle d'historique réduit à ses opérations suspectes peut contenir des opérations simultanées inter-dépendantes.*

5.5 Hiérarchie et Hétérarchie

Le résultat d'une opération terminée à l'instant t pour un produit P peut devenir le pré-requis d'une opération lancée beaucoup plus tard sur le même produit P . Entre temps, d'autres opérations ont pu être à leur tour exécutées sur ce même produit (cf. figure 4.18). De fait, le modèle d'historique peut présenter des propriétés de hiérarchie.

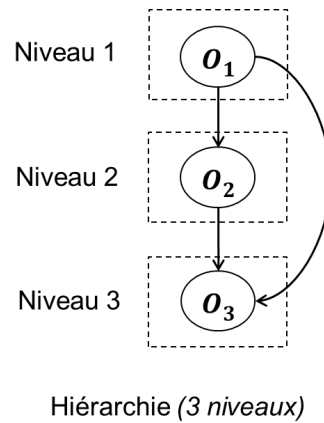
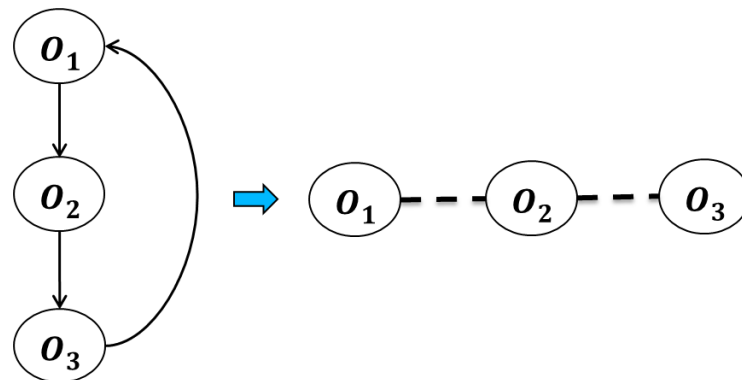


FIGURE 4.18 – Structure hiérarchique

Propriété 5. *Le modèle d'historique réduit à ses opérations suspectes peut être un graphe hiérarchique.*

Compte tenu de la propriété de simultanéité des opérations, et comme le montre la figure 4.19, le modèle d'historique peut-être un graphe causal fortement connexe et donc hétérarchique (Trentesaux et al., 2008).



Hétérarchie = graphe causal fortement connexe

FIGURE 4.19 – Structure hétérarchique

Propriété 6. *Le modèle d'historique réduit à ses opérations suspectes peut être un graphe hétérarchique.*

6 Conclusion

Préalablement à l'hybridation qui est au centre de notre approche, ce chapitre s'est proposé d'étendre l'approche de diagnostic logique à la prise en compte d'autres causes que celles généralement ciblées, à savoir les équipements. Basé sur l'expertise issue des travaux menés au sein de deux projets Européens (IMPROVE et INTEGRATE) menés en collaboration forte et étroite avec l'entreprise ST-MicroElectronics, d'autres causes sont à considérer comme l'humain que ce soit en exploitation ou maintenance, les recettes qui, développées dans un contexte non stressé, peuvent ne pas être supportées par les équipements en contexte réel de production, ou encore la matière première sur laquelle les produits sont fabriqués. L'extension à ces causes a nécessité des interventions sur les modèles eux mêmes mais également sur les algorithmes de génération en ligne de l'historique des évolutions. Au delà de ces extensions, nous avons également soulevé le problème à ce jour non résolu d'opérations simultanées, problème auquel nous avons apporté des solutions. La fin de ce chapitre a permis enfin de préciser les propriétés générales du modèle historique obtenu afin de mettre en perspective une base saine de transformation de modèles qui se trouve au cœur du chapitre suivant.

Chapitre 5

Approche Mixte Logique et Probabiliste

1 Introduction

L'idée générale que nous défendons ici consiste à nous appuyer sur le mécanisme de génération en ligne du modèle d'historique avancé des opérations exécutées réduit à celles suspectes pour identifier la structure du réseau Bayésien correspondant au modèle de diagnostic, et de mener par la suite le calcul des probabilités du modèle Bayésien résultant afin de déterminer les candidats à visiter en premier (notion de score) et ainsi contribuer à optimiser la prise de décision pour la maintenance corrective. Ce chapitre va ainsi s'attacher à proposer une démarche innovante d'hybridation d'une approche logique déterministe et d'une approche probabiliste à ces fins. La section 1 se consacrera à la présentation générale de cette approche. La section 2 quant à elle s'attachera à détailler les mécanismes que nous avons développés pour transformer un modèle d'historique logique en la structure graphique d'un réseau Bayésien. Les calculs de probabilités relatifs au modèle graphique obtenu feront l'objet de la section 3. Discussion et conclusions termineront ce chapitre.

2 Approche générale proposée

Le principe de l'hybridation que nous soumettons est présenté au sein de la figure 5.1. L'approche que nous avons nommée MLP pour Mixte Logique et Probabiliste, s'interface avec les ordres émis vers les chaînes fonctionnelles et les comptes rendus générés en retour d'une part, et d'autre part avec l'opérateur de maintenance au travers d'une sortie matérialisée par une liste de causes potentielles chacune affectée d'un score (i.e. probabilité de pertinence). Au cœur de l'approche, une première phase, logique, permet de manière dynamique et sur la base de modèles de bons fonctionnement, de caractériser un sous-ensemble logique de causes. Ce sous-ensemble, généré et managé en ligne servira à la construction de la structure graphique Bayésienne. Le processus d'identification d'une telle structure est désormais acquis avant que la défaillance ne se produise (réalisé en fait en temps masqué), garantissant ainsi un gain de temps précieux et crédible au regard de la réactivité demandée. Fort de cette structure graphique obtenue par transformation de modèles, reste à définir les tables de probabilité de chacune des variables, conditionnellement à ses causes et de mener par la suite des

inférences ; en fonction de la défaillance observée, nous allons ainsi rechercher ses causes.

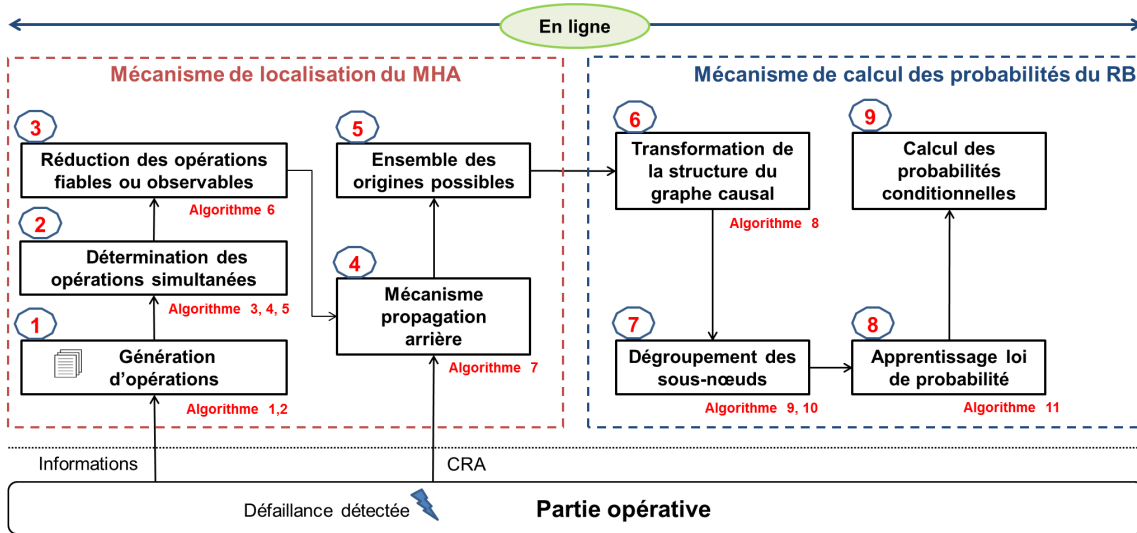


FIGURE 5.1 – Le mécanisme de diagnostic du modèle MLP

3 Transformation du modèle logique en réseau Bayésien

3.1 Identification automatique de la structure du RB

A partir du modèle historique étendu et réduit aux opérations à suspecter, le processus d'identification de la structure du RB peut être lancé. Cette identification consiste en :

- la transformation du modèle logique en un graphe orienté permettant son portage en tant que structure du RB. Les opérations deviendront des nœuds du graphe, et relations entre ces opérations formaliseront les arcs du réseau.
- Les nœuds du graphe résultant modélisant une opération, un processus de dégroupement de ces nœuds sera nécessaire afin d'atteindre explicitement le niveau de granularité souhaité.

3.2 Transformation de la structure du modèle

Définition 8. *Equipement* : une opération est exécutée par la chaîne fonctionnelle. L'équipement est défini par la partie matérielle de la chaîne fonctionnelle.

Définition 9. *Produit* : le produit d'une opération est représenté dans l'évolution du flux de produit. Une opération peut engendrier une évolution du flux de produit ou non. L'élément produit peut donc apparaître ou non dans la représentation d'une opération.

Dans l'approche proposée, une opération O_i pour $i \in J_{propagation}$ ¹ est transformée en un nœud O_i du graphe causal (cf. figure 5.2). Les variables relatives à l'équipement (E), au facteur humain (H), au produit (P) et à la recette (R) seront à cette étape masquées par l'opération ; le

1. $J_{propagation}$ est l'ensemble des opérations suspectes après de propagation

nœud opération étant considéré comme macro (i.e. macro étape d'un grafcet). Les arcs entre les nœuds du graphe causal sont directement l'image des liens entre les opérations (liens entre les effets d'une opération et les pré-requis d'une autre). La figure 5.3 présente un exemple de la transformation d'un lien en un arc.

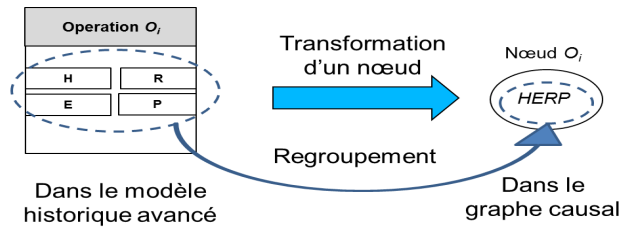


FIGURE 5.2 – Transformation d'un nœud

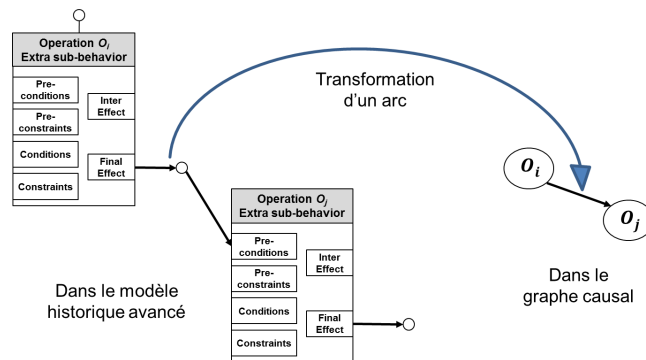


FIGURE 5.3 – Transformation d'un Arc

Afin d'automatiser le passage du modèle logique à un premier graphe causal simplifié, nous avons développé l'algorithme présenté dans la figure 5.4.

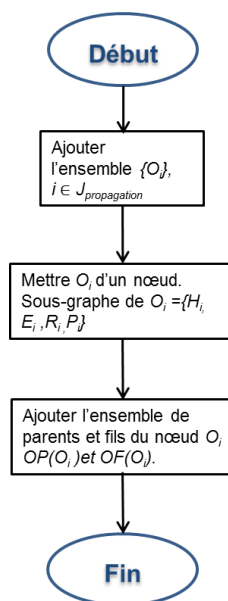


FIGURE 5.4 – **Algorithme 8** : Transformation du modèle logique en graphe causal

Dans cet algorithme, nous utilisons :

- $Parent(O_i)$ est l'ensemble des opérations parentes de l'opération O_i , pour $i \in J_{propagation}$. $Parent(O_i) = OP(O_i)$ correspond donc à l'ensemble des opérations antérieures à l'opération O_i ,
- $Fils(O_i)$ est l'ensemble des fils de l'opération O_i . $Fils(O_i) = OF(O_i)$ correspondant donc à l'ensemble des opérations ultérieures à l'opération O_i .

3.3 Dégrouper les nœuds

Le graphe causal étant généré justement sur la base de macro-nœuds opérations (modélisant chacun un sous-graphe), reste désormais à dégrouper ces nœuds afin de pouvoir aboutir au réseau Bayésien souhaité. Pour ce faire, nous allons nous intéresser ici à spécifier le sous-graphe opération constitué des variables $\{H, E, R, P\}$.

3.3.1 Relations internes entre $\{H, E, R, P\}$

Dans les systèmes automatisés de production, la performance du système est évidemment liée aux capacités offertes par sa partie opérative et donc ses équipements. Cependant, l'outil de production n'est rien sans ses recettes, sans ses opérateurs de maintenance, sans le produit qu'il est sensé transformé. Tout ceci est clairement identifié dans (Bettayeb, 2014). Dans ces travaux, M. Belgacem Bettayeb démontre que la performance de l'équipement réside non seulement dans la technologie, mais montre également qu'une bonne ou mauvaise maintenance de la part des opérateurs humains peut venir l'altérer. C'est également le cas si la matière première injectée dans le SAP n'est pas conforme aux attentes ou si la recette n'est pas correctement qualifiée.

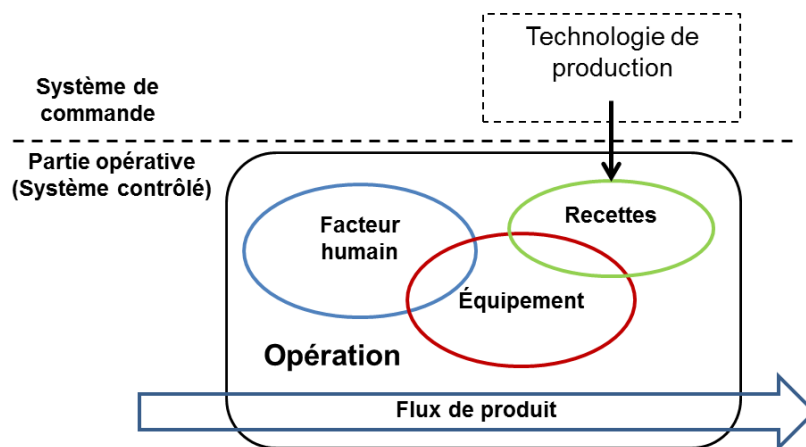


FIGURE 5.5 – Relation entre les éléments de l'ensemble $\{H, E, R, P\}$

La figure 5.5 extraite de (Bettayeb, 2014) en est une parfaite illustration. Elle expose clairement le fait que la qualité même du produit fabriqué dépend d'un ensemble de facteurs qui interagissent entre eux. Ces interactions ou relations de cause à effet sont généralement reconnues comme étant :

- Impact direct du facteur humain sur la qualité des produits : mauvaise manipulation et mauvais positionnement d'un produit par exemple ;

- Impact direct de l'équipement sur la qualité des produits : dérive d'un composant de l'équipement par exemple ;
- Impact de la Recette sur la qualité des produits : les recettes étant généralement qualifiées dans des environnement non stressés, elles peuvent se révéler inadaptées au contexte réel de production. L'équipement peut ici largement supporter la recette, mais les conséquences directes sont sur la qualité du produit ;
- Impact direct de l'humain sur l'équipement : c'est le cas notamment au travers de l'activité de maintenance où de nombreuses études ont démontré la grande variabilité d'application des procédures, provoquant parfois des dérives produits importantes ;
- Impact de la Recette sur l'équipement : les recettes étant généralement qualifiées dans des environnement non stressés, elles peuvent se révéler inadaptées au contexte réel de production. Ici en revanche, la recette peut ne pas être supportée par l'équipement. Il est à ce titre souvent observé qu'un changement de recette initialement RA vers RB est supportée alors que le contraire ne l'est pas, sans que l'on sache à ce jour en expliquer la raison.

De surcroît, et comme nous l'avons déjà noté dans les chapitres précédents, une défaillance potentielle peut se propager d'une opération à l'autre véhiculée en fait par le flux produits (i.e. propagation de défaillances). Cela signifie qu'après que l'opération a été exécutée, les interactions entre l'humain, l'équipement et les recettes ont toutes des répercussions sur la qualité du produit, entraînant ainsi un phénomène de cascade d'échecs.

Notons également que en théorie, le produit peut avoir un impact significatif tant sur l'équipement que sur l'être humain. Ces impacts sont généralement relatifs aux risques d'inflammabilité, d'explosion, ou d'intoxication (Flaus, 2013). Dans le domaine dans lequel nous nous plaçons, nous posons une hypothèse restrictive comme quoi le produit n'a pas d'impact sur la santé humaine. Bien entendu, ceci n'est pas généralisable à tous les procédés.

3.3.2 Sous-graphe opération

La section précédente a décrit les impacts du facteur humain sur les équipements et les produits. A des fins de simplification dans un premier temps, nous avons volontairement regrouper en seul nœud graphique tous les facteurs humains qu'ils aient un impact sur le produit (manupulation direct) ou sur l'équipement (maintenance). Il en résulte le figure 5.6, et le cas 3 que nous retenit par la suite de l'étude.

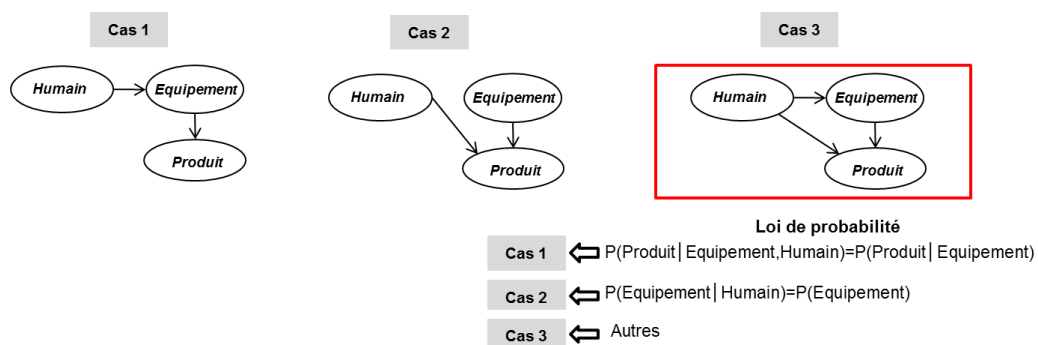


FIGURE 5.6 – Impact de facteur humain

Il en sera de même pour l'impact des recettes sur l'équipement et/ou le produit selon la figure 5.7.

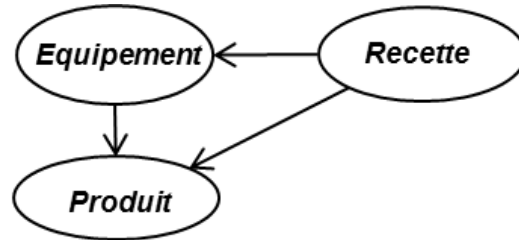


FIGURE 5.7 – Impact de recette

Définition : Nous proposons le sous-graphe opération mettant en relation les variables $\{H, E, R, P\}$ comme indiqué dans la figure 5.8. Nous appellerons ce sous-graphe **modèle de structure interne d'opérations** (MSIO) (Nguyen et al., 2015).

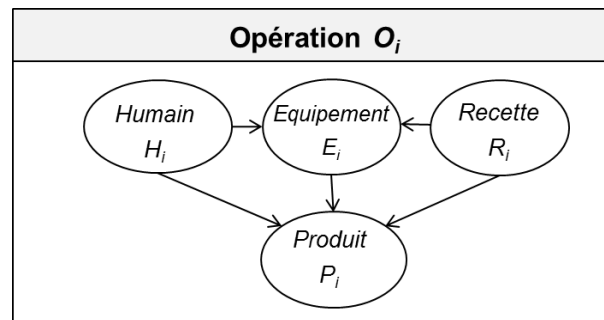


FIGURE 5.8 – Relation de causalité entre les éléments de l'ensemble $\{H, E, R, P\}$

Remarque : selon les différentes catégories d'opérations (opérations d'information, induite, requise ou d'action, cf. la section 2.2 page 44), ce sous-graphe peut être amené à être réduit, la variable équipement restant cependant quant à elle irréductible pour des raisons évidentes. La figure 5.9 présente les différentes structures possibles du sous-graphe.

Liens entre les nœuds dégroupés

Une fois le sous-graphe remonté au niveau du graphe, reste à déterminer les liens de causalités entre les différents sous-graphes. Deux cas sont à considérer ici, selon la nature des opérations et leurs effets sur le flux de produit. S'il y a effet, la relation entre deux sous-graphes sera de type Impact du produit sur l'équipement, sinon de l'équipement du sous-graphe précédant à l'équipement du sous-graphe suivant (Nguyen et al., 2015). La figure 5.10 illustre ce principe.

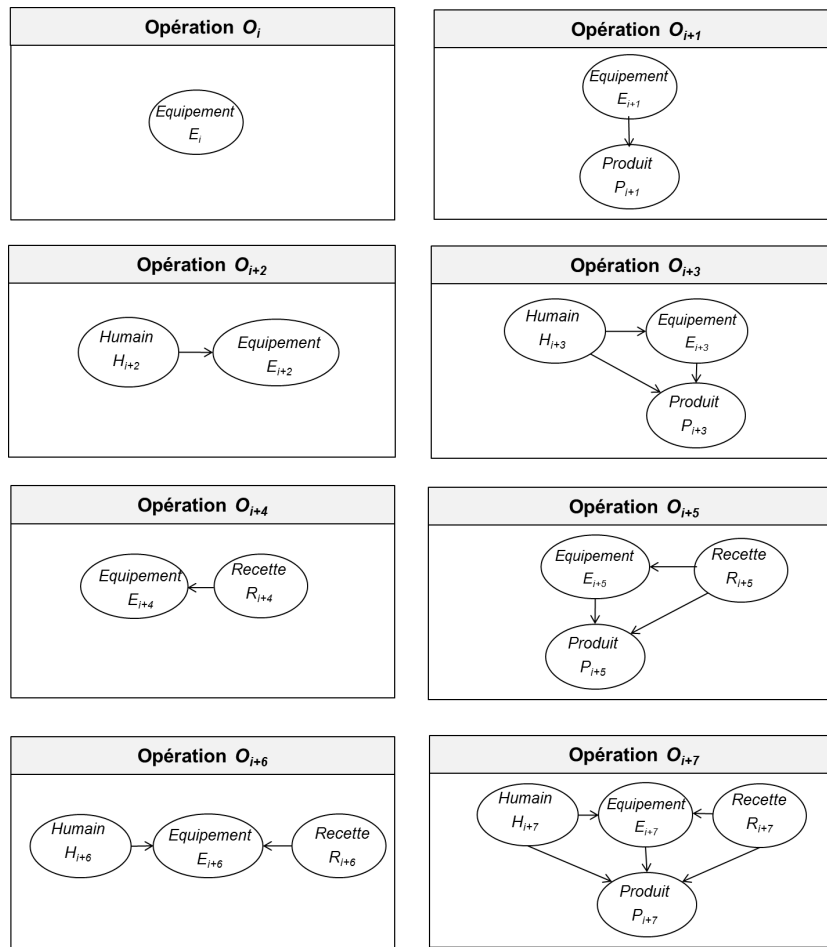


FIGURE 5.9 – Différentes catégories des relations internes de l'ensemble $\{H, E, R, P\}$

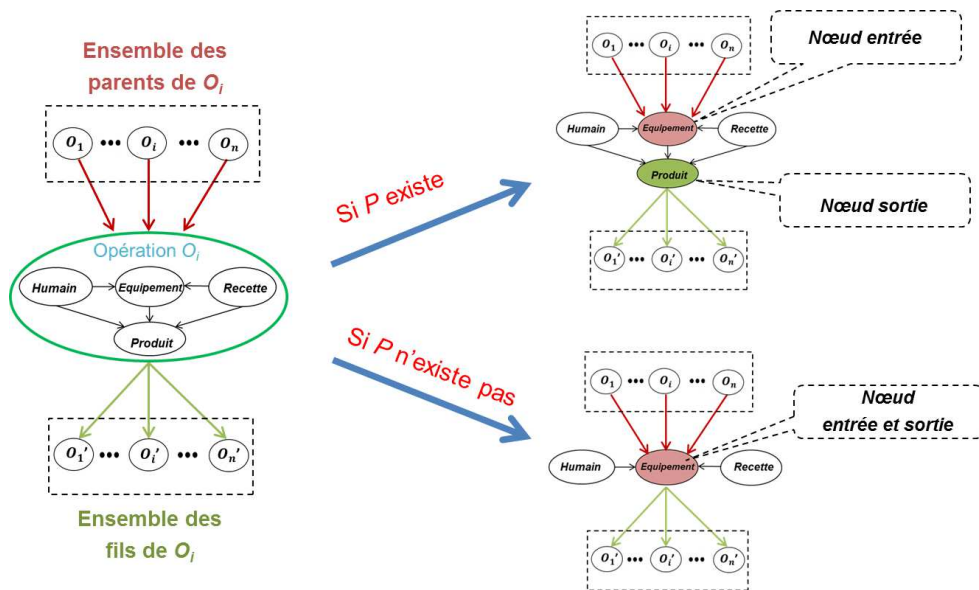


FIGURE 5.10 – Nœud d'entrée et nœud de sortie

3.3.3 Algorithme de dégroupement des nœuds

Sur la base des principes exposés précédemment, et dans un objectif d'automatisation du dégroupement des nœuds nous avons proposé les algorithmes formalisés dans les figures 5.11 et 5.12).

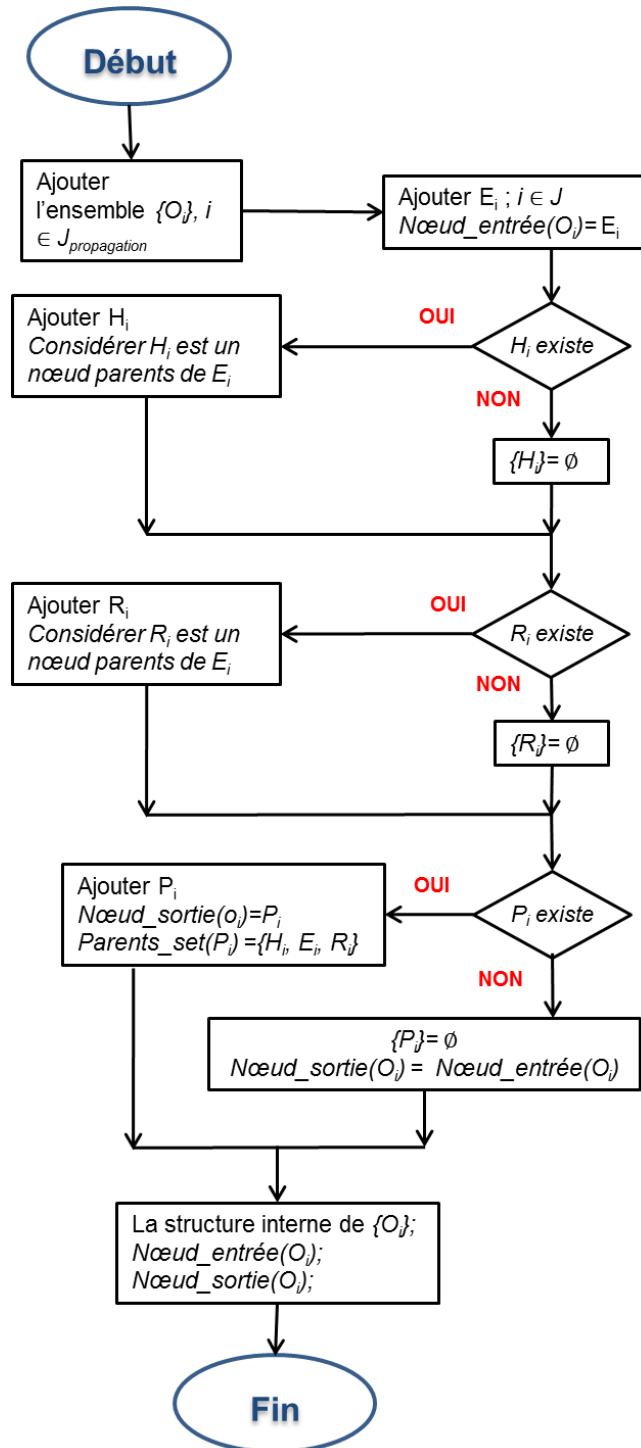
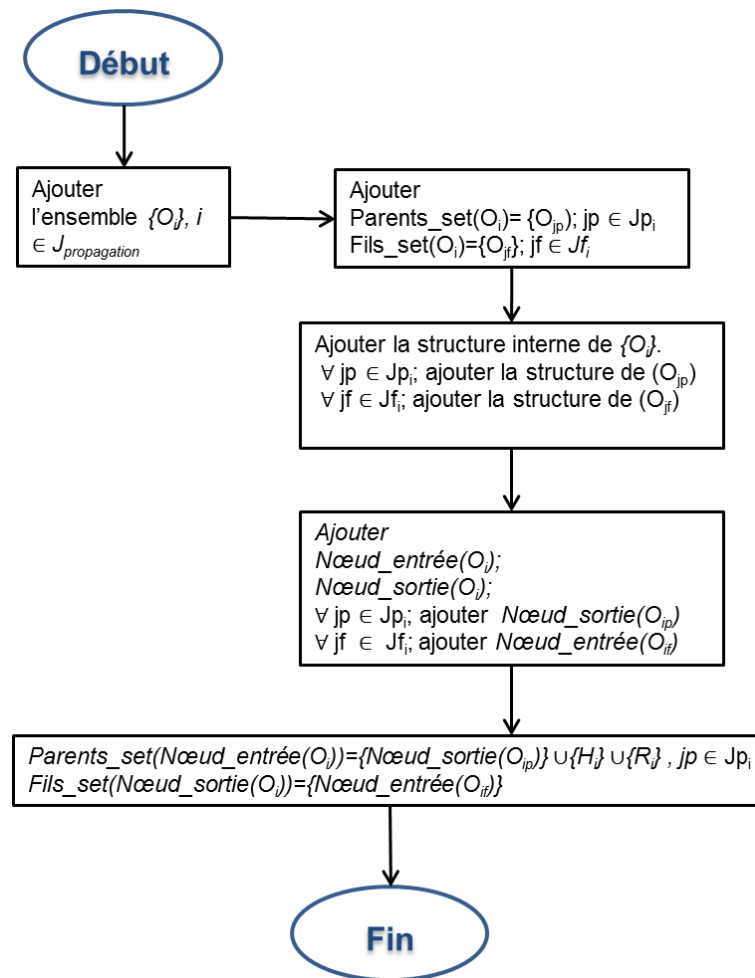


FIGURE 5.11 – **Algorithme 9** : Dégroupement des nœuds opération

FIGURE 5.12 – **Algorithme 10** : Connexion entre les nœuds dégroupés

Si nous prenons un exemple basé sur quatre opérations couvrant les différents types que sont la Préparation, la Transformation, et la Transitique O_1 , O_2 , O_3 et O_4 associées respectivement aux équipements E_1 , E_2 , E_3 et E_4 , le graphe de la figure 5.13 en donne une représentation tout à fait réaliste.

Comme nous pouvons le constater, cette phase plutôt basique de transformation du modèle logique entre graphes causaux ne présente pas de complexité algorithmique importante. Nous comprenons ici tout l'avantage de la méthode proposée qui permet de mener un apprentissage dynamique ciblé et en temps masqué de la structure Bayésienne. La méthode est d'autant plus avantageuse dans un contexte de forte variabilité produit ou process. Dans le cas d'une forte variabilité process, il s'agit pour les opérateurs humains de rajouter les fiches opératoires correspondantes.

Reste désormais à mener les calculs de probabilité pour finaliser le réseau Bayésien, c'est l'objet de la section suivante.

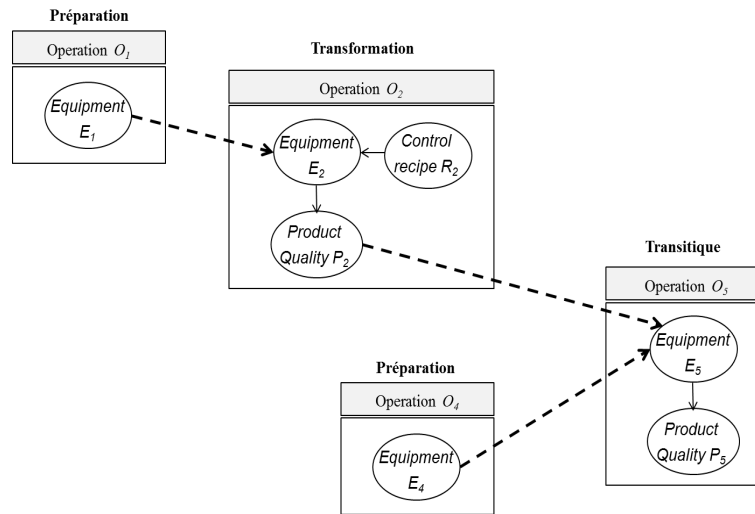


FIGURE 5.13 – Exemple des liens entre les nœuds

4 Calcul de probabilités

4.1 Classe de Réseau Bayésien

La structure du RB que nous avons obtenu après la transformation s'apparente à une structure augmentée de Réseau Bayésien. Aussi avons nous retenu la classe des Réseau Bayésien TAN (Naïm et al., 2011).

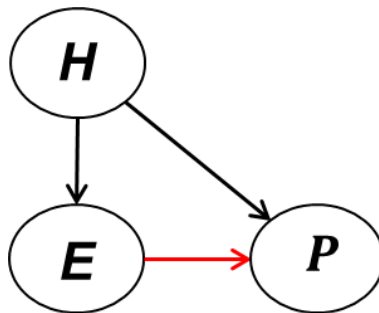


FIGURE 5.14 – Structure augmenté

4.2 Loi de probabilité du RB

La loi de probabilité RB décrit les probabilités conditionnelles associées avec les parents pour tout nœud dans ce RB. Ces probabilités conditionnelles sont estimées à partir de la base de données statique d'état des nœuds dans le graphe du RB. La loi de probabilité permet de calculer la probabilité conditionnelle entre deux nœuds quelconques ou entre un nœud et un ensemble de nœuds quelconques dans le RB.

Considérons que le nœud N_j a l'ensemble de parents $\{N_j^1, \dots, N_j^{k_j}, \dots, N_j^{n_j}\}$, avec $k_j = 1 \dots n_j$. La probabilité conditionnelle de N_j associée avec ses parents est calculée par $P(N_j | N_j^1, \dots, N_j^{k_j}, \dots, N_j^{n_j})$

4.3 Apprentissage des lois de probabilité

Comme nous avons pu le voir, la structure des sous-graphes étant à « géométrie variable » en fonction du type d'opération à considérer, nous avons été amenés à développer un algorithme adapté à cette spécificité. Il est présenté dans la figure 5.15 et calcule en fonction de la structure interne de l'opération O_i pour $i \in J$, les lois de probabilités correspondantes.

- $P(H_i)$: Probabilité de H_i .
- $P(R_i)$: Probabilité de R_i .
- $P(E_i \mid Parents_set(E_i))$: Probabilité associée de E_i sachant ses parents.
- $P(P_i \mid H_i, E_i, R_i)$: Probabilité associée de P_i sachant ses parents.

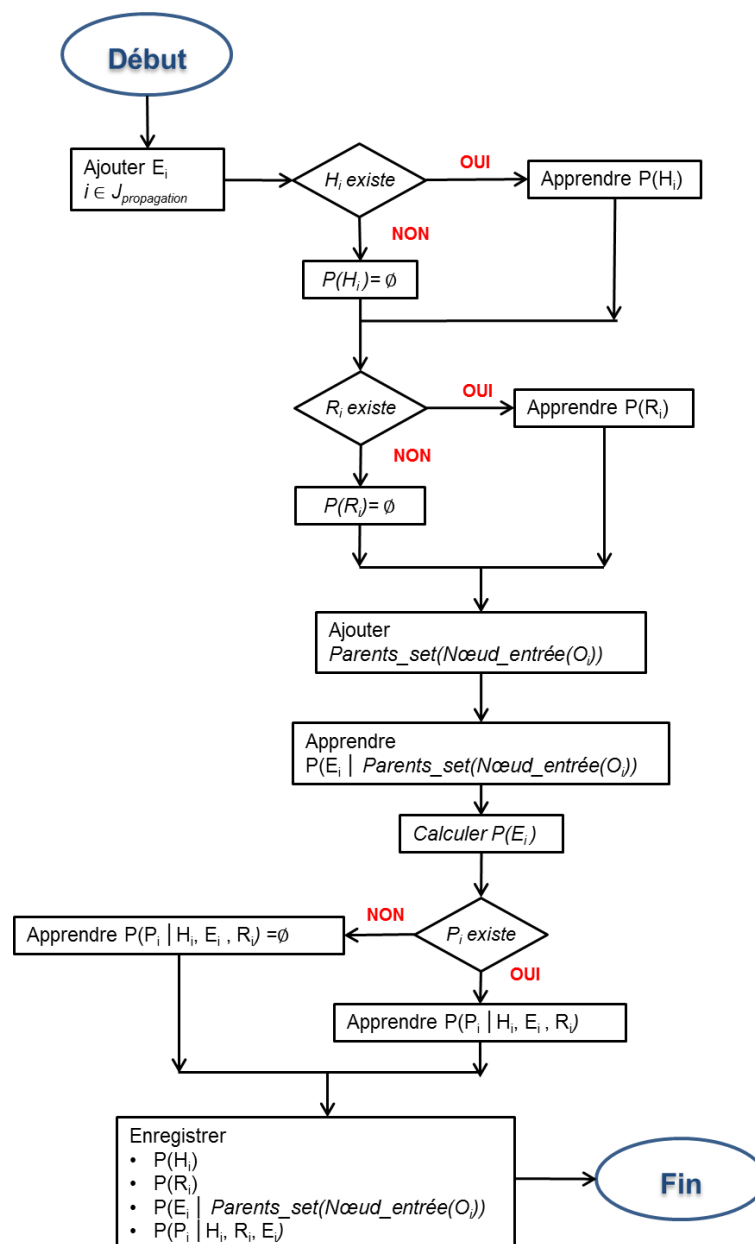


FIGURE 5.15 – **Algorithme 11** : Apprentissage des lois de probabilité entre les nœuds

4.4 Calcul des probabilités

L'idée ici est de calculer la probabilité d'un nœud dans la structure correspondante à l'occurrence de la défaillance ; c'est-à-dire la probabilité conditionnelle d'un nœud N_j sachant la défaillance pour $j \in [1, n]$. Comme chaque nœud N_j est supposé prendre plusieurs valeurs, nous appellerons $E(N_j)$ cet ensemble. Nous nous proposons donc de calculer la probabilité $P(N_j = n_j \mid Défaillance)$.

Si nous prenons les notations suivantes :

- $Fils(N_j)$ est ensemble des fils du nœud N_j .
 - $Fils^k(N_j) = Fils(Fils^{k-1}(N_j)) = N_{i,j,k}$ est ensemble des fils de $Fils^{k-1}(N_j)$. L'indice $k \in [1, m_{(i,k,j)}]$ est déterminé par $Fils_{(i,k,j)}^m(N_j) = Défaillance$.
 - $Parents(Fils^k(N_j))$ est ensemble des parents des $Fils^k(N_j)$.
- alors,

$$P(N_j = n_j \mid Défaillance) = P(N_j) \left\{ \prod_{k=1}^{m_{i,k,j}} \sum_{N_{i,k,j}} .P(N_{i,k,j} \mid Fils^k(N_j) \mid Parents(Fils^k)) \right\} \cdot P(Parents(Fils^k)) \cdot P(Défaillance \mid Parents(Défaillance)) \quad (5.1)$$

Ces probabilités représentent les probabilités de pertinence de la cause recherchée (nœud) dans la structure du RB. Elles vont permettre d'aider l'opérateur de maintenance dans sa phase de recherche des causes et de maintenance.

5 Conclusion

Ce chapitre clôt cette seconde partie en présentant le cœur de notre approche centrée sur l'hybridation entre l'approche de diagnostic logique initialement proposée dans (Deschamps, 2007) et une approche Bayésienne. Au delà des intérêts de l'hybridation que nous avons mis en exergue ici, il s'est agi de proposer une technique de transformation de modèle permettant de passer d'un modèle historique d'opération, au sens de la planification automatique, à la structure graphique d'un réseau Bayésien. Cette transformation est centrée sur les notions de nœuds, d'arcs, de graphe causal et de sous-graphes. Ainsi, à partir d'un modèle d'historique nous avons proposé un ensemble d'algorithmes de transformation permettant de générer au besoin la structure graphique Bayésienne correspondante. D'autres algorithmes sont ensuite proposés pour mener les calculs des TPC sur la base des données extraites du site de production. Les fondements de notre approche étant désormais posés, la partie suivante va se consacrer à mettre notre approche à l'épreuve d'un cas d'étude inspiré d'un procédé réel.

Troisième partie

Exemple d'application

Chapitre 6

Présentation du cas d'étude

1 Introduction

Dans cette partie, nous avons souhaité développer un exemple d'application de notre approche de diagnostic hybride Logique et Probabiliste afin de valider, sur un cas réaliste repris d'une véritable structure industrielle, notre proposition. Nous avons choisi ici de retenir un processus de production inspiré des projets Européens IMPROVE et INTEGRATE, projets menés en collaboration forte et étroite entre le laboratoire G-SCOP et ST-MicroElectronics. Le système et les données retenus sont donc repris de ces projets. Nous tenons ici à exprimer toute notre reconnaissance aux différents acteurs de ces recherches, qu'ils soient industriels ou chercheurs, et qui m'ont non seulement inspiré dans le cadre de mes recherches et qui m'ont également permis d'offrir à ma thèse un cas d'expérimentation précieux.

Ce premier chapitre va donc s'attacher à présenter le cas d'étude que nous avons retenu, alors que le chapitre suivant s'attachera à présenter l'application de notre approche sur ce cas d'étude.

La première section en donnera les caractéristiques techniques essentielles. Après quoi nous décrirons son architecture de pilotage au sein de laquelle nos algorithmes seront testés. Ensuite, la section suivante s'attachera à la modélisation des opérations. Après quoi nous présenterons la structure des bases de données que nous avons pré-traité et utilisé pour l'apprentissage. Ce chapitre se terminera par la présentation rapide de notre environnement de développement.

2 Système de production de Semi-Conducteur

2.1 Généralités

Le marché du semi conducteur est en forte croissance depuis quelques années avec l'évolution considérable des technologies mises à disposition du consommateur (i.e. téléphone, portable, ordinateur,...).

Cette évolution se traduit en particulier par une miniaturisation importante des composants électroniques atteignant des taux de réduction de taille de 13% par an (cf. figure 6.1), Moore (Schaller, 1997) annonçant pour cette année l'atteinte de la taille minimale.

En parallèle à cette miniaturisation, une augmentation de la taille des disques de silicium

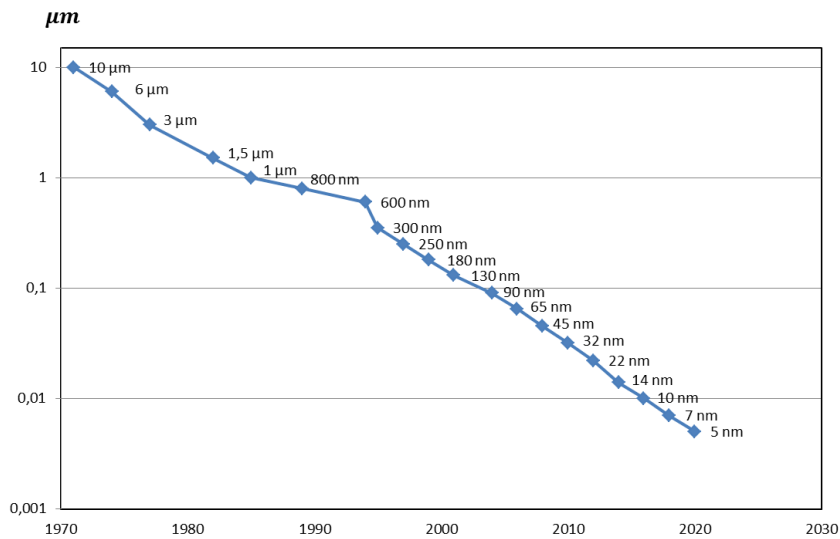


FIGURE 6.1 – Taille minimale des transistors (*Roadmap SIA 1999*)

(aussi appelés Wafer) sur lesquels les composants électroniques sont gravés a été opérée passant de diamètres de 100mm, il y a quelques années, à des diamètres de 300mm aujourd'hui.

Il en résulte une maîtrise des process de fabrication de plus en plus difficile qui a naturellement participé à l'émergence du contrôle avancé de tels procédés (Advanced Process Control - APC).

Nous pouvons également rajouter que le marché du processeur s'est transformé en moins de 5 ans de manière très significative. En effet, là où il y a encore quelques années, le marché du processeur généraliste bien maîtrisé était de cour (le marché du PC en fait), aujourd'hui, le PC tend à disparaître au profit de produits à très courtes durées de vie que sont les tablettes et les smartphones. Ce bouleversement dans les manières de consommer a un impact significatif dans la production, instaurant une variabilité produit très importante sur des cycles de vies extrêmement courts (Hubac et Zamai, 2013).

2.2 Process de fabrication

La fabrication du wafer s'appuie sur une série d'étapes de traitement importantes (Quirk et Serda, 2001), en moyenne 7000 opérations sur un parc de 300 machines.

La figure 6.2 en est une illustration, présentant les grandes étapes de cette fabrication depuis la matière première (silice) au boîtier (microprocesseur).

Ces étapes clés sont le Physical Vapor Deposition (PVD) et le Chemical Vapor Deposition (CVD), la photolithographie, la gravure par plasma (Plasma Etch), le Rapid Thermal Processing (RTP), et le polissage mécano-chimique (CMP) (voir figure 6.3).

Durant la fabrication, un cristal de silicium est étiré lentement afin de former un cylindre appelé lingot. Ce dernier est ensuite découpé en disques appelés *wafers* au moyens de fils diamantés. Chacun d'entre eux est ensuite poli pour obtenir la qualité du surface demandée.

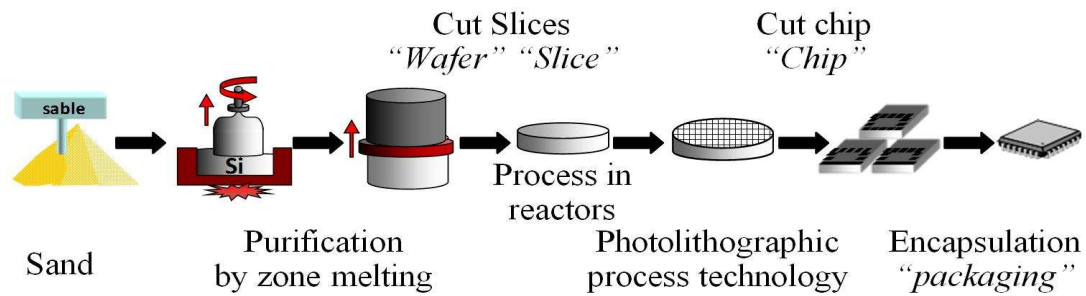


FIGURE 6.2 – Vue simplifiée des étapes de fabrication d'un microprocesseur (Duong et al., 2012b)

Un procédé de dépôt de substrat est par la suite appliqué au disque poli afin de fournir une surface propre (non contaminée) prête à accueillir les autres processus de fabrication. Ainsi, le wafer est exposé à de très hautes températures afin dans un premier temps de former une couche (SiO_2) diélectrique (oxydation). Le wafer est alors revêtu d'un matériau photo-résistant. Un processus de lithographie peut alors être lancé afin de réaliser un masque sur le matériau photo-résistant : la lumière durcit les portions exposées. Un processus de gravure permet alors d'enlever l'oxyde (SiO_2) des parties non durcies, reste alors le motif endurci sur le wafer. Un processus de diffusion et d'implantation est ensuite appliqué afin de déposer des ions (dopage) sur parties exposées. Différentes régions aux caractères de conductivité différents sont alors obtenues. Ce processus de lithographie est répété autant de fois que le nécessite les spécifications du microprocesseur considéré afin de former les transistors. Un processus de métallisation est ensuite lancé afin de déposer une couche d'aluminium ou de cuivre sur le wafer. L'excès de métal est ensuite enlevé par un autre processus de lithographie afin de conférer l'inter-connectivité désirée. Une autre couche d'oxyde diélectrique est déposée sur l'aluminium ou cuivre pour isoler la première couche d'aluminium de la couche suivante. A ce point, chaque wafer est poli utilisant un polissage mécano-chimique (CMP) pour obtenir un wafer à surface lissée. Puis la couche d'aluminium (cuivre, titane) suivante est déposée, modelée, et gravée afin de créer une autre couche. Le processus est répété en autant de couches inter-connectées que nécessaire pour la conception du chip. In fine, pour fabriquer un wafer, il se sera écoulé entre 56 et 60 jours, pendant lesquels plus de 700 opérations/produit (7000 steps unitaires) auront été effectuées sur un parc de 300 équipements.

2.3 Process de production et maîtrise de la qualité de fabrication

Comme nous venons de le voir, le processus de production des wafers se révèle être complexe et particulièrement sensible ne serait-ce que sur le plan économique :

- Coût d'un wafer nu : 100\$
- Coût d'un wafer en sortie d'usine : 1000\$-15000\$

Aussi, était-il nécessaire de mettre en place un processus de surveillance adapté au contexte. Ainsi, au delà des machines de traitement complexes, des machines de métrologies ont été implantées afin de tester les produits durant le cycle de fabrication. La figure 6.4 les représente.

Une analyse, synthétisée dans la figure 6.5, des rejets des produits ayant subi des tests permettent de révéler que les causes principales des produits évalués *Out of control* (OC).

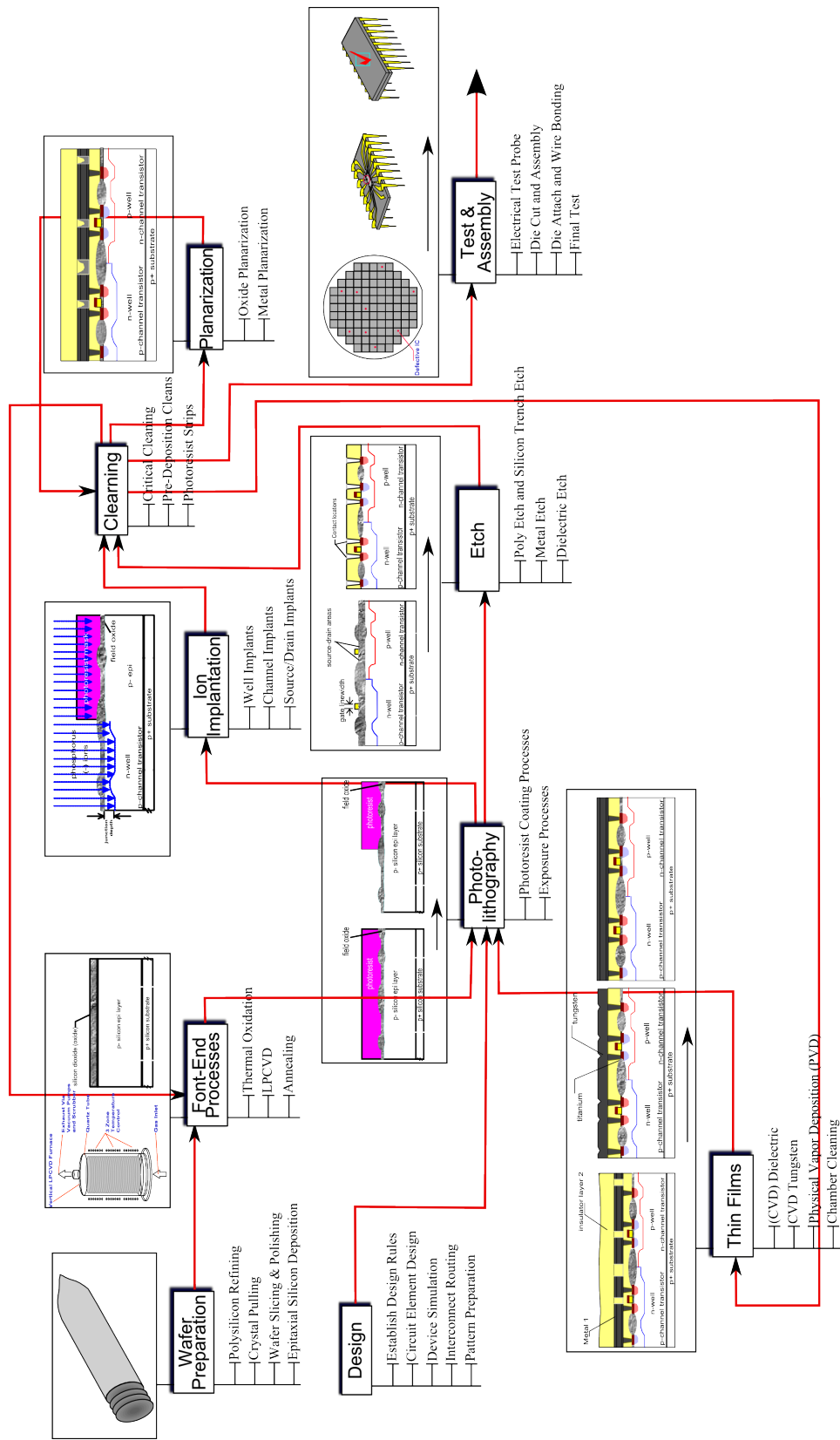


FIGURE 6.3 – Processus global de production semi-conducteur (Duong et al., 2012b)

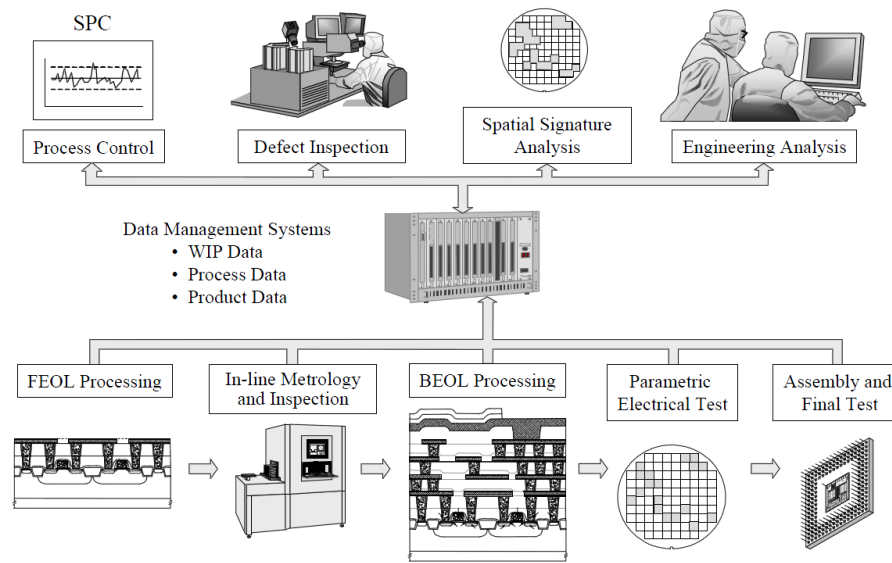


FIGURE 6.4 – Métrologie dans la système de production (Quirk et Serda, 2001)

Comme nous pouvons le voir, les équipements, recettes, produits et maintenances (other) sont pointés du doigt dans ces analyses, montrant à quel point notre approche est pertinente. Le besoin de localiser, suite à la détection d'un produit hors contrôle, l'origine du défaut est de fait crucial pour :

- lancer une maintenance corrective sur la bonne machine ou une révision des procédures de maintenance, ou une révision des recettes...,
- améliorer ainsi les rendements de production minimisant les temps d'arrêts et de dérives.

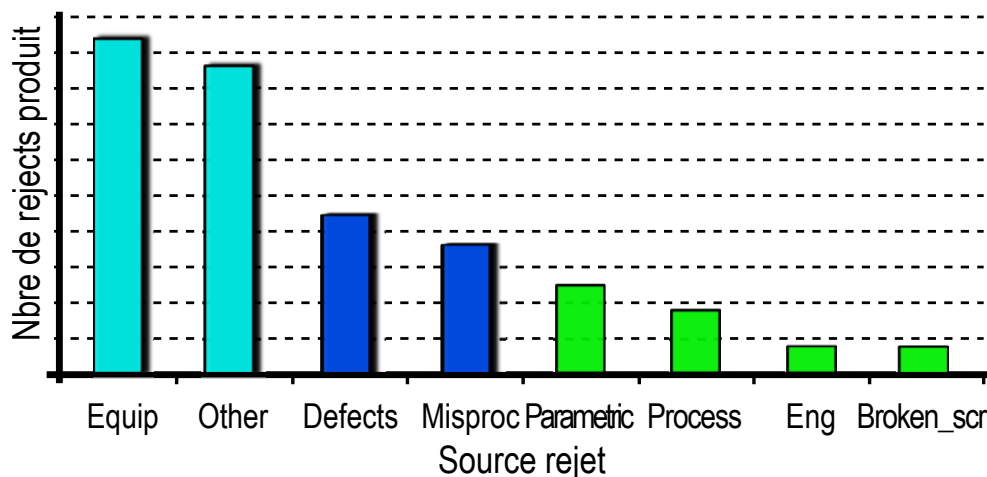


FIGURE 6.5 – Nombre de rejets classés par source (Mili, 2009)

3 Procédures de contrôle dans le domaine du semi-conducteur

D'un point de vue industriel et afin de contribuer à limiter les pertes et donc les rendements, les industriels ont intégré différentes techniques de contrôles basées sur l'exploitation

des différentes données du site de production, que ce soient celles issues de la métrologie, du monitoring, etc... Ces approches font parties de ce que l'on appelle l'Advanced Process Control (APC) (Martinez et Edgar, 2006), (Sematech, 1997) et sont généralement structurées selon trois blocs distincts tel que l'expose la figure 6.6 : le Run 2 Run, la FDC, la métrologie (SPC).

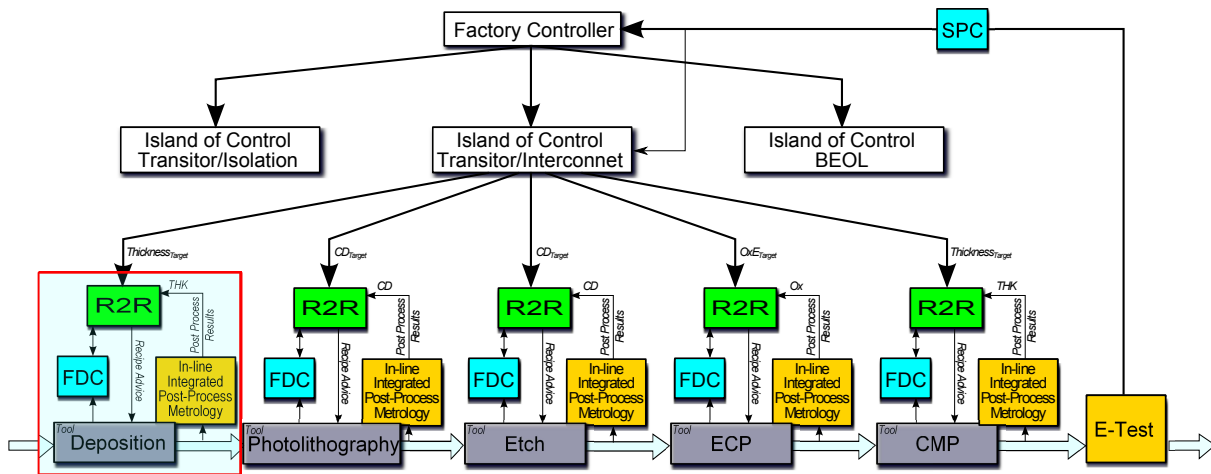


FIGURE 6.6 – Contrôle des processus de semi-conducteurs

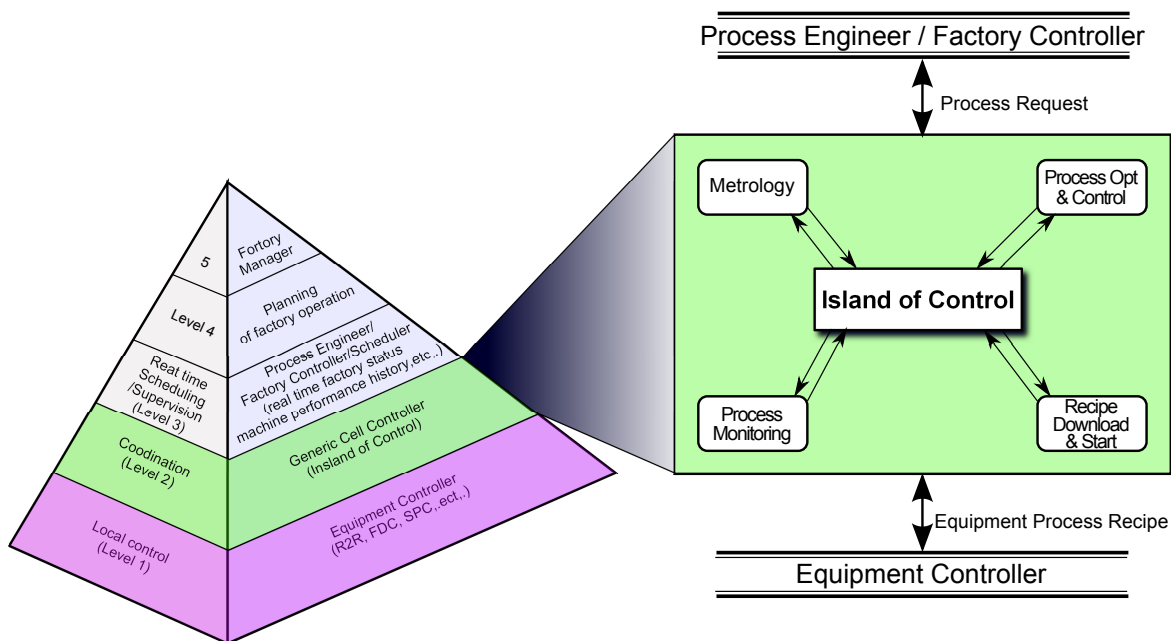


FIGURE 6.7 – Architecture CIM dans la production de semi-conducteurs

Notons que ces approches se localisent dans la structure CIM au sein des deux premiers niveaux de l'architecture tel que montré dans la figure 6.7 (Hill et Nettles, 1997; Baweja et Ouyang, 2000; Sonderman et Spanos, 2005). Détaillons alors ces trois blocs.

3.1 Contrôle Run to run

La Figure 6.8 représente des boucles de régulation qui ont pour objectif de garantir la stabilité des processus de fabrication durant toute la vie de l'équipement (Martinez et Edgar, 2006). Ces boucles assurent la liaison entre SPC et FDC.

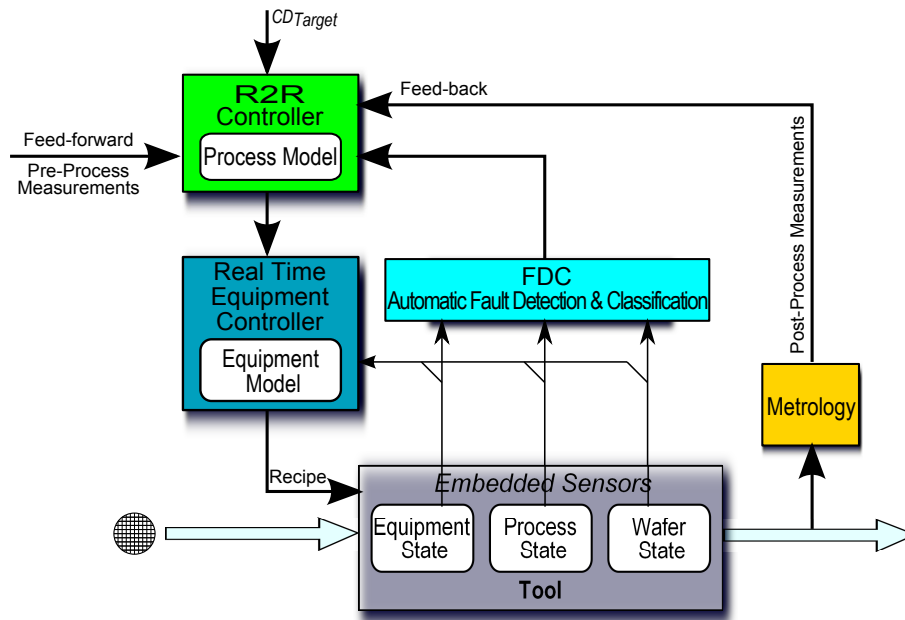


FIGURE 6.8 – Contrôle Run to Run

3.2 Contrôle FDC

Correspondant à *Fault Detection and Classification*, ce système permet de suivre en temps réel l'évolution des paramètres des équipements (température, pression, etc). Ainsi, durant les étapes de fabrication, les paramètres équipement sont collectés permettant ainsi de mieux maîtriser les anomalies de fonctionnement.

3.3 Contrôle SPC

SPC correspond au *Statistical Process Control*. Un tel système permet de contrôler les lignes de production à l'aide d'indicateurs de performance comme par exemple l'épaisseur, la pression, le centrage (Martinez et Edgar, 2006). Le SPC est généralement appliqué sur trois types de mesures :

- Les mesures physiques qui permettent de vérifier avec retard la conformité de ce qui a été fait par rapport à ce qui a été demandé.
- Les mesures paramétriques, réalisées sur tous les wafers d'un lot. Nous parlerons de PT (Parametric Tests).
- Les mesures électroniques faites sur des produits finis et donc en fin de ligne. Nous parlons de EWS (Electrical Wafer Sorting).

Ces types de mesures sont associés à deux types de limites :

- Les limites de spécification à ne pas franchir pour un bon fonctionnement des circuits.
- Les limites de contrôle fixant des frontières qualifiant le bon fonctionnement de l'opération en cours.

4 Focus sur le processus étudié

Comme précisé dans l'introduction de ce chapitre, nous nous sommes largement inspirés des cas d'études développés dans le cadre des projets menés par mes prédécesseurs au sein des projets Européens IMPROVE puis INTERGRATE (Bouaziz et al., 2011; Duong et al., 2012a; Shahzad et al., 2012; Said et al., 2014; Asma et al., 2015).

Le cas d'étude que nous avons retenu, et qui s'inspire en partie de ces travaux est illustré dans la figure 6.9. Il se compose de 5 îlots de fabrication : Fabrications des masques (M_1 ; M_2), Wafer-cleaning (M_3 ; M_4 ; M_5), Deposition (M_6 ; M_7), Lithography (M_8) et Etching (M_9 ; M_{10}). Chaque îlot est autonome, sa structure de pilotage exécutant la recette appropriée au lot à traiter indiqué par l'opérateur assurant le chargement et le déchargement du lot dans l'équipement. La fin du processus est marqué par une phase de métrologie qui permet d'évaluer la qualité des opérations réalisées précédemment.

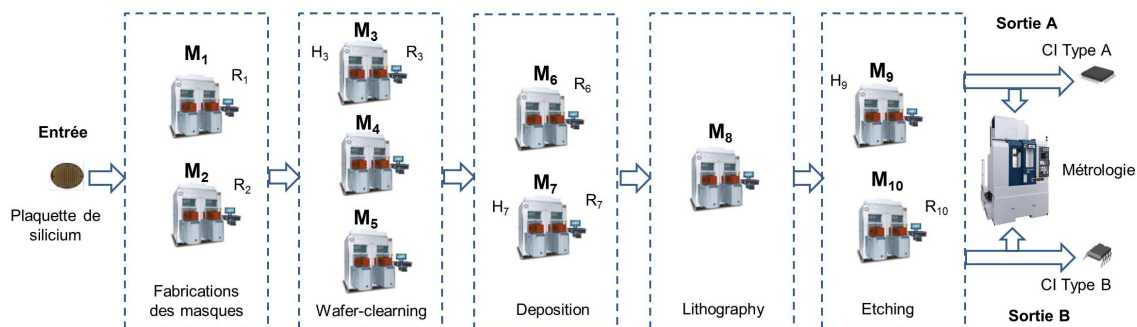


FIGURE 6.9 – Processus étudié

Ce processus de production "en partie simplifié" pour des raisons de place, permet de fabriquer deux types de produits CI Type A et CI Type B. Seul un échantillon des produits fabriqués est testé par l'îlot de métrologie.

5 Description des opérations

Sur la base de la définition du modèle d'opérations présentée section 2.1 page 43 nous en analysons :

- 10 opérations d'action qui correspondent aux 10 machines M_i , $i \in [1, 10]$.
- 4 opérations induites pour transporter les produits entre les îlots.
- 3 opérations requises pour introduire d'une part des produits dans l'atelier, et d'autre part les extraire (sortie A et B).

- 10 opérations de surveillance installées sur les équipements pour surveiller l'état des machines et les variables de comportement qui ont participé à l'exécution de l'opération d'action considérée. L'opération de surveillance peut être activée ou non selon la politique de R2R.

Afin d'alléger ce chapitre, nous avons rejeté en annexe l'ensemble des descriptions de ces opérations et avons proposé de n'en conserver qu'une seule au sein de ce chapitre, l'opération d'action de "Wafer-cleaning" exécutée par la machine M_3 .

La fiche correspondante à cette opération d'action O_3 affectée à la machine M_3 est représentée par la figure 6.10. Il y est indiqué que l'équipe humaine en charge de la machine est H_3 , R_3 correspond à la recette et P_3 au lot de wafer qui subira cette opération de transformation. Les dates de début et fin sont 09h52m14s et 10h21m42s. L'opération consiste à amener l'état du produit de non nettoyé à l'état nettoyé. Au niveau de l'évolution de la chaîne fonctionnelle, l'effet transitoire caractérise le fait que l'opération de lavage est en cours, l'effet final indique que l'opération de lavage est terminée et que l'équipement est prêt à refaire un cycle de lavage.

Opération Wafer-cleaning O_3		
Équipement M_3	Facteur humain H_3	Recette R_3
L'instant de début: 09:52:14	Produit P_3	Etat d'opération : Non suspect
L'instant de fin : 10:21:42		
Évolution chaîne fonctionnelle		
Effet Transitoire: Lavage en cours	Pré-condition: Fonction M_1 == disponible	Pré-contrainte: Intérieur M_3 == vide
Effet final: Lavage terminé Équipement prêt IC(Lavage terminé)=1	Condition:	Contrainte: Etat M_3 == prêt
Évolution de flux de produits		
Effet Transitoire: P_3 == nettoyage en cours	Pré-condition: P_3 == non nettoyé	Pré-contrainte: P_3 == prêt
Effet final: P_3 == nettoyé IC(P_3 == non nettoyé) = 1	Condition:	Contrainte: Sortie == P_3 nettoyé

FIGURE 6.10 – Opération de Wafer-cleaning O_3 sur la machine M_3

6 Pré-traitement des informations statistiques

Afin d'alimenter le processus de calculs de lois de probabilité, nous présentons dans cette section les bases de données que nous avons utilisées pour mener ces calculs. Bien entendu, pour des raisons de confidentialité, les données sont générées et nous avons également modifié les noms des variables.

Les bases de données sur lesquelles nous nous sommes appuyées représentent 6 mois de vie du process de production et sont le résultat d'une synthèse des bases de données FDC (Fault Detection and Classification), SPC (Statistical Process Control) et RMS (Base de recettes). Les données se composent des numéros d'identification des opérations, des dates de lancement, du nom de l'équipement, du numéro de lot, du nom de l'opération, de la recette et de l'identificateur de l'équipe humaine en charge de l'équipement au moment de l'opération. La Figure 6.11 présente un extrait de ces données.

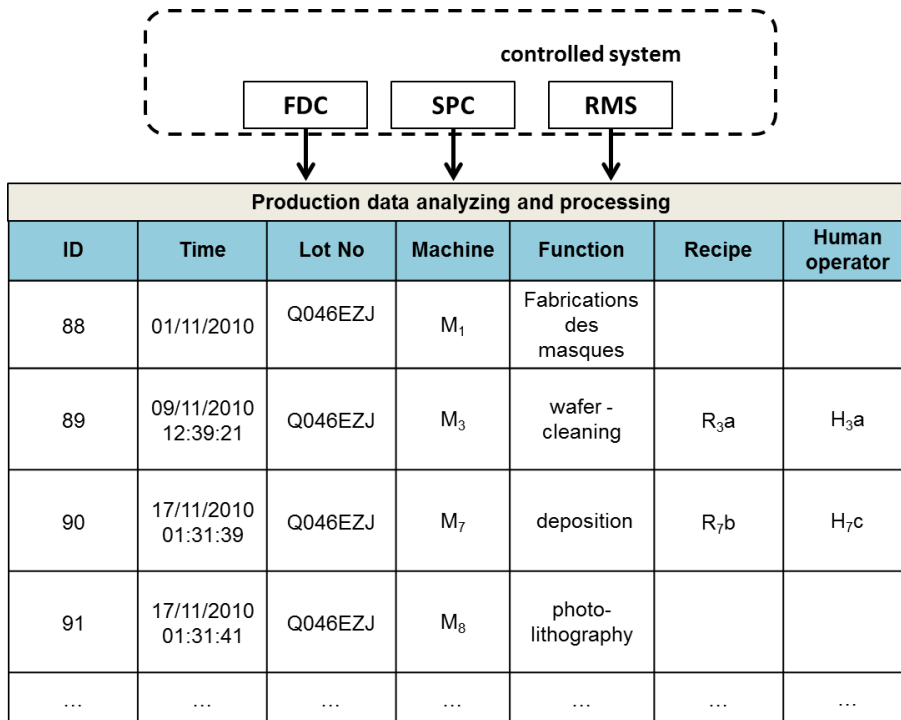


FIGURE 6.11 – Historiques de production

Une seconde base de données est également utilisée, basée sur la même période de vie du process, soit 6 mois. Elle permet de synthétiser trois autres bases en mettant l'accent sur la vie de l'équipement et donc de sa maintenance. C'est ce qu'illustre la figure 6.12 avec R2R pour Run To Run, FDC pour Fault Detection and Classification et CMMS pour Computerize Maintenance Management System.

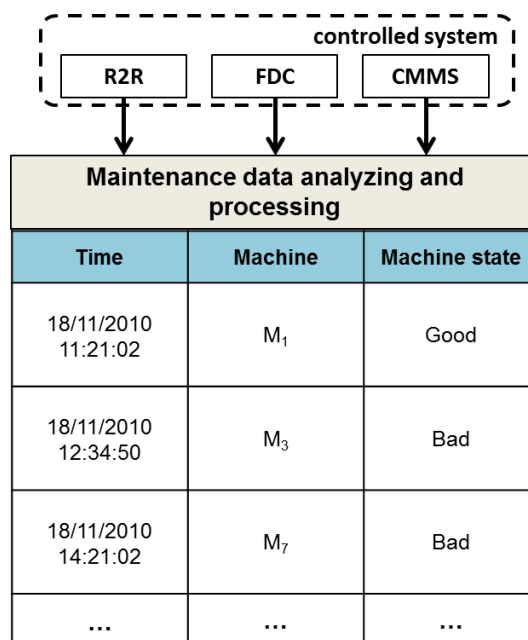


FIGURE 6.12 – Base de données maintenance

Enfin, nous utiliserons une troisième base de données relative quant à elle à la métrologie produit comme présenté dans la figure 6.13. Ces informations servent à calculer les probabilités d'occurrence de la défaillance.

Metrology
 ↓

Metrology data analyzing and processing			
Time	Product type	Lot No	Product quality
09/12/2010 12:39:17	Type A	Q046EZJ	Good
09/12/2010 14:39:17	Type C	Q044HJH	Bad
09/12/2010 18:39:17	Type A	Q046AJA	Bad
...

FIGURE 6.13 – Base de données metrologie

7 Environnement de développement

Nous terminerons ce chapitre par une courte présentation de l'environnement de test sur lequel nous avons mené nos développements informatiques. Les algorithmes que nous avons décrits tout au long de ce mémoire ont été développés sur la plate-forme logicielle MATLAB R2011b déployée sur PC Intel Core i5-2400 (fréquence processeur 3,1GHz, mémoire RAM 4 Go). Concernant l'apprentissage des lois de probabilité nous avons eu recours à l'algorithme espérance-maximisation (en anglais Expectation-maximisation algorithm, souvent abrégé EM), proposé par (Dempster et al., 1977). Il s'agit d'une classe d'algorithmes permettant de trouver le maximum de vraisemblance des paramètres de modèles probabilistes lorsque le modèle dépend de variables latentes non observables. L'ensemble des tests s'est appuyé sur trois scénarios qui sont présentés dans le chapitre suivant.

8 Conclusion

Ce chapitre s'est attaché à présenter le décor de notre cas d'étude. Repris en partie d'un cas concret industriel il met en scène 10 équipements de production, 10 opérations d'actions, 4 opérations induites, 3 opérations requises et 10 opérations de surveillance (métrologie). A partir de ces éléments réels, nous avons montré comment les opérations ont été modélisées et qu'elles bases de données générées et pré-traitées nous allons utiliser pour mener les ap-

prentissages. Les éléments de cadrage de ce cas d'étude étant présentés, nous proposons au lecteur de découvrir l'application de notre approche dans le dernier chapitre de ce mémoire.

Chapitre 7

Application du diagnostic hybride

1 Introduction

Sur la base du processus étudié, nous proposons dans ce chapitre de tester nos algorithmes que nous avons développés sur la plate-forme Matlab-simulink.

Dans ce dernier chapitre du manuscrit, nous nous proposons de réaliser des tests permettant de valider, sur des données générées pour cause de confidentialité, notre approche de diagnostic logique et probabiliste. Ces tests seront basés sur 3 scénarii permettant d'illustrer les différentes situations difficiles auquel nous pourrions être confrontés : cas d'un multi-routage de produits, cas des opérations simultanées et enfin situation de multiples défaillances. L'étude de ces trois scénarii permet de mettre en valeur les apports de notre approche.

Notons que quelque soit le scénario considéré, nous avons posé en tant qu'hypothèse avant lancement d'une simulation que :

- Les opérations offertes par les équipements sont disponibles et prêtes.
- Les équipements sont exempts de produits en cours de traitement.
- Les sorties des équipements sont vides.
- Le système de captage des équipements est fiable.
- Les zones de transits entre les équipements sont vides.

L'ensemble des algorithmes présentés dans ce manuscrit ont été développés par mes soins sur la plate-forme Matlab-Simulink.

```

1 - Lot= Q050_Type_A;
2 - Time_start= 17; Time_end_max=52; Time_end_min=47;
3 - Load(Set_operation); Load(t);Load(List_operation(t));
4 - i=0;
5 - while isempty(List_operation(t))<1
6 -     i=i+1;
7 -     Load(Operation(i)); Load(niveau_suspecte(i));
8 -     Load(E(i), H(i), R(i), P(i));
9 -     Load(Pre_condition(i); Pre_contrainte(i); Condition (i); Contrainte(i));
10 -    Pre_requis_O(i)=(Pre_condition(i); Pre_contrainte(i); Condition (i); Contrainte(i));
11 -    Load(Eft_E(i), Eft_E(i), Eft_P(i), Eft_P(i));
12 -    Effet_O(i)=(Eft_E(i) & Eft_E(i), Eft_P(i), Eft_P(i));
13 -    Load(IC_O(i));
14 -    List_operation(t)=setdiff(List_opeartion(t),Operation(i));
15 -    I=i;
16 - end
17 - clear i;
18 - I=length(Set_operation);
19 - Set_non_suspect_suple=[];
20 - for i=1:I
21 -     OP_O(i)=[];
22 -     OF_O(i)=[];
23 -     for l=1:L
24 -         O_av(l)=Set_operation(l,l);
25 -         connexion_past =intersect(Pre_requis_O(i),Effet(O_av(l)));
26 -         if isempty(conexion) =0
27 -             OP_O(i)=union(OP_O(i),O_av(l));
28 -             OF_O_av(l)=union(OF_O_av(l),O(i));
29 -         end
30 -     end
31 -     for j=1:I
32 -         if j<>i
33 -             connexion_present=intersect(Pre_requis_O(i), Effet_O(j));
34 -             if connexion_present=0
35 -                 OP_O(i)=union(OP_O(i),O(j));
36 -                 OF_O(j)=union(OF_O(j),O(i));
37 -             end
38 -         end
39 -     end
40 -     Operation_simultanee_O(i)=union(OP_O(i), OF_O(i));
41 -     if IC_O(i)=1
42 -         disp('O(i) est non suspecte);
43 -         else disp('O(i) est suspecte')
44 -         Set_non_suspect_suple=union(Set_non_suspect_suple,O(i));
45 -     end
46 - end
47 - Set_operation =union(Set_operation, Liste_operation(t));
48 - Set_non_suspect=union(Set_non_suspect,Set_non_suspect_suple);
49 - Set_suspect=setdiff(Set_operation,Set_non_suspect);

```

Génération d'opérations exécutées

Détermination d'opérations simultanées

Réduction d'opérations non suspects

FIGURE 7.1 – Programme pour la génération des opérations

2 Scénario 1 : Multi-routes et multi-produits

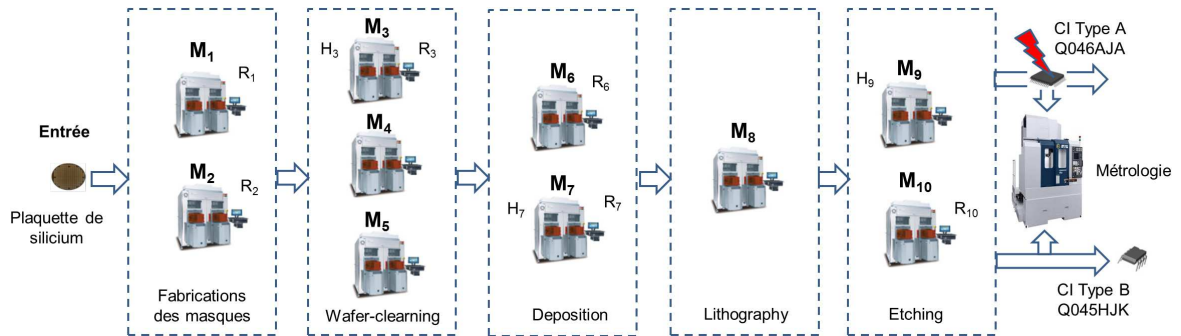


FIGURE 7.2 – Scénario 1 : Multiples routes et produits

Dans ce scénario, nous avons souhaité évaluer les capacités de notre approche à traiter une problématique de propagation de défaillances dans le cadre de la fabrication parallèle de deux types de produits (Type A et Type B) qui suivent des routes différentes comme indiqué dans la figure 7.2. A 18h 39mn et 17s, une défaillance est détectée sur l'équipement de métrologie au niveau du lot Q046 relatif au produit de Type A. Supposons alors le modèle historique réduit aux opérations à suspecter de la figure 7.3.

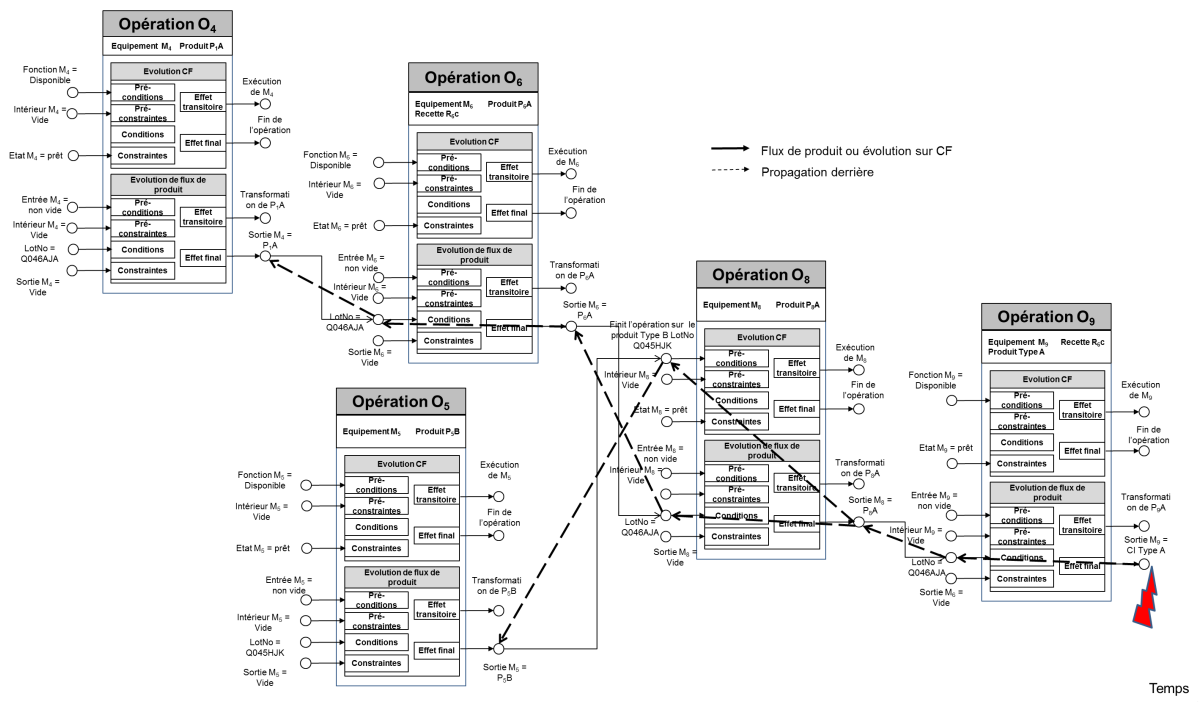


FIGURE 7.3 – Modèle historique logique au moment de la défaillance

Examinons désormais pas à pas le comportement de notre programme.

A 18h 39m 17s donc, le mécanisme propageant la suspicion arrière est lancé se basant sur l'algorithme 7 (cf. page 57) pour nous fournir l'ensemble des origines possibles de la défaillance détectée sur le lot Q046. Selon ce mécanisme, le programme 7.4 les opérations à suspecter

sont : O_4, O_5, O_6, O_8, O_9 exécutées respectivement sur les équipements M_4, M_5, M_6, M_8, M_9 . Notons que O_5 a bien été identifié alors que associée au produit de type B. Comme il n'existe pas de divergence vers l'avant dans le modèle, l'ensemble des opérations est naturellement suspecté ; l'opérateur de maintenance est certes aidé, puisque parmi les 10 opérations qu'offre l'atelier, sa zone d'investigation a été réduite de 50%. Reste cependant à affiner l'ensemble des candidats potentiels restant.

```

1 - Load(EfT_suspect(n), EfF_suspect(m));
2 - for j=1:J
3 -     if or(ismember(EfT_suspect(n), O(j))==1, ismember(EfF_suspect(m), O(j))==1)
4 -         disp('O(j) est suspecte');
5 -         Liste = union(Liste, O(j));
6 -     end
7 -
8 -     OP_propa=OP(O(i));
9 -     Liste= union(setdiff(Liste, O(i)), OP_propa);
10 -     while isempty(Liste)<1
11 -         L=length(Liste);
12 -         for i=1:L
13 -             OP_(i)=OPListe(1, i);
14 -         end
15 -         Liste=setdiff(union(Liste, O(i)), OP_propa);
16 -     end

```

FIGURE 7.4 – Extrait du programme permettant la propagation arrière au moment de la défaillance

Le tableau 7.1 présente une synthèse de la situation faisant suite au mécanisme de propagation arrière.

TABLE 7.1 – Résultat de la propagation du scénario 1

Opération suspecte	Connaissance	Situation précédente	Situation suivante
O_9	H_9, M_9	O_8	Défaillance Produit Type A
O_8	M_9	O_5, O_6	O_9
O_6	R_6, M_6	O_4	O_8
O_5	M_5	\emptyset	O_8
O_4	M_4	\emptyset	O_6

Les opérations suspectes O_4, O_6, O_5, O_8, O_9 sont transformées en nœuds du graphe causal. Ensuite les nœuds obtenus sont ensuite dégroupés par l'algorithme de dégroupement proposé section 3.3.3, page 70. La partie du programme réalisant ce service est donnée figure 7.5.

```

1 - Load(Liste);
2 - for k=1:I
3 -     T=O(k);
4 -     for j=1:J
5 -         N(t)=T;
6 -         Inter_N(t)=[H(k), E(k), R(k), P(k)];
7 -         Parent_N(t)=OP_propa_O(k);
8 -         Fils_N(t)=OF_propa_O(k);
9 -     end
10 -    Str_N(t) = Zero(2,3);
11 -    if isempty(H(k)) ==1
12 -        Str_N(t)(1,1)=0;
13 -    elseif isempty(H(k)) ==0
14 -        Str_N(t)=H(k);
15 -    end
16 -
17 -    if isempty(R(k)) ==1
18 -        Str_N(t)(1,3)=0;
19 -    elseif isempty(R(k)) ==0
20 -        Str_N(t)(1,3)=R(k);
21 -    end
22 -
23 -    if isempty(P(k)) ==1
24 -        Str_N(t)(2,2)=0;
25 -        output_node=E(k);
26 -    Parent_E(k)=union(Parent_N(t), Str_N(t)(1,1)*H(k), Str_N(t)(1,3));
27 -    Fils_E(k)=Fils(Str_N(t)(1,2));
28 -    elseif isempty(P(k)) ==1
29 -        Str_N(t)(2,2)=P(k);
30 -        out_put_node=P(k);
31 -        Fils_E(k)=P(k);
32 -        Fils_P(k)= strc(Child_N(t))(1,2);
33 -    end

```

FIGURE 7.5 – Extrait du programme permettant la transformation et le dégroupement des nœuds en sous-graphes

Présentés sous la forme d'un tableau (cf. 7.2), les résultats de ce dégroupement sont synthétisés ci-après :

TABLE 7.2 – Informations du graphe causal

Nœud	Ensemble de parents	Ensemble de fils
Défaillance sur produit Type A	M_9	\emptyset
M_9	H_9, P_8	Défaillance sur produit Type A
H_9	\emptyset	H_9
P_8	M_8	M_9
M_8	P_5, P_6	P_8
P_6	M_6, R_6	M_8
M_6	P_4, R_6	P_6
R_6	\emptyset	M_6, P_6
P_5	M_5	\emptyset
M_5	\emptyset	P_5
P_4	M_4	\emptyset
M_4	\emptyset	P_4

À l'issue de cette étape, l'apprentissage des lois de probabilité du graphe est lancé. La partie du programme relative à cette étape est donnée dans la figure 7.6

```

1  % input data apprentissage
2  %clear all
3  load scenario10h_data;
4
5  P_M4=[matran(1,1) 1-matran(1,1)];
6  P_R6=[matran(2,1) 1-matran(2,1)];
7  P_M5=[matran(3,1) 1- matran(3,1)];
8  P_H9=[matran(4,1) 1- matran(4,1)];
9
10 %compute P_P5
11
12 P_P5=P_M5(1,1)*P5_M5(1,2)+(1-P_M5(1,1))*P5_M5(2,2);
13
14
15 P_P6a=zeros(4,1);
16 %compute P_P6
17 for i=1:4
18     P_P6a(i,1)=P6_M6R6(i,3)*(P_M6+(1-P6_M6R6(i,1))*(1-2*P_M6))*(P_R6(1,1)+(1-P6_M6R6(i,2))*(1-2*P_R6(1,1)));
19 end
20
21 %compute P_P4
22
23 P_P4=P_M4(1,1)*P4_M4(1,2)+(1-P_M4(1,1))*P4_M4(2,2);
24
25 % compute P_M6
26 P_M6a=zeros(4,1);
27 %P_M6_P4R6
28 for i=1:4
29     P_M6a(i,1)=M6_P4R6(i,3)*(P_P4+(1-M6_P4R6(i,1))*(1-2*P_P4))*(P_R6(1,1)+(1-M6_P4R6(i,2))*(1-2*P_R6(1,1)));
30 end
31 P_M6=sum(P_M6a);
32 clear i;
33 P_P6=sum(P_M6a);
34 clear i;
35
36 P_M8a=zeros(4,1);
37 %computeP_M8
38 P_M8a=zeros(4,1);
39 for i=1:4
40     P_M8a(i,1)=M8_P6P5(i,3)*(P_P6+(1-M8_P6P5(i,1))*(1-2*P_P6))*(P_P5+(1-M8_P6P5(i,2))*(1-2*P_P5));
41 end
42 P_M8=sum(P_M8a);
43 clear i;
44
45 %compute P_P8
46
47 P_P8= P_M8*P8_M8(1,2)+(1-P_M8)*P8_M8(2,2);
48

```

Apprentissage des probabilités conditionnelles

FIGURE 7.6 – Extrait du programme permettant l'apprentissage à partir des bases de données historiques

Pour cela, notre programme met en place le calcul de l'ensemble des probabilités conditionnelles suivantes :

- $P(M_4=bad)$, $P(M_5=bad)$ sont les probabilités de M_4 , M_5 qualifiées à l'état "bad".
- $P(R_6)=R_6A$) et $P(H_9=H_9C)$ sont les probabilités d'occurrence de $R_6)=R_6A$ et de $H_9=H_9C$.
- $P(P_4 | M_4)$ et $P(P_5 | M_5)$ sont les probabilités conditionnelles entre P_4 , P_5 et M_4 , M_5 .
- $P(M_6 | P_4, R_6)$ est la probabilité conditionnelle de M_6 sachant P_4 et R_6 .

- $P(P_6 | M_6, R_6)$ est la probabilité conditionnelle de P_6 sachant M_6 et R_6 .
- $P(M_8 | P_6, P_5)$ est la probabilité conditionnelle de M_8 sachant P_6 et P_5 .
- $P(P_8 | M_8)$ est la probabilité conditionnelle entre P_8 et M_8 .
- $P(M_9 | P_8, H_9)$ est la probabilité conditionnelle entre M_9 et P_8 et H_9 .
- $P(Fa | M_9)$ est la probabilité conditionnelle sachant Fa et M_9 .

La figure 7.7 représente de manière graphique le réseau Bayésien obtenu jusqu'à présent.

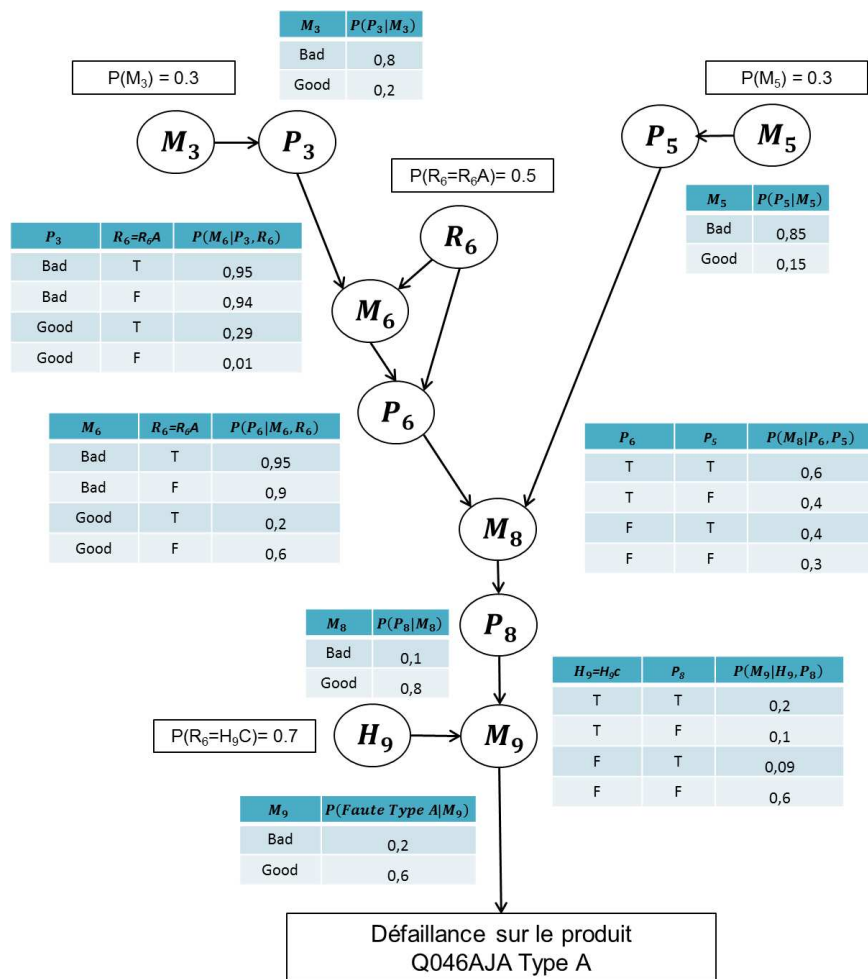


FIGURE 7.7 – Réseau Bayésien obtenu relatif au scénario 1

Reste désormais à interroger le modèle sur les différentes causes potentielles. C'est ce que présente de manière synthétique le tableau 7.3 obtenu à partir du programme de la figure 7.8.

Au terme de cette dernière étape, nous pouvons en déduire que l'origine de la défaillance est plus probablement de l'ordre de l'équipement et qu'il s'agit, fort probablement, de la machine M_4 . Nous conseillons à l'opérateur de maintenance d'orienter son diagnostic détaillé

```

Editor - C:\Users\Bigboss\Documents\MATLAB\These\scenario1_calcul.m
File Edit Text Go Cell Tools Debug Desktop Window Help
Stack: Base
- 1.0 + ÷ 1.1 x
62 %*****
63 %*****CALCUL DE PROBABILITE*****
64
65 %compute P_M6_P4R6
66 P_M6_P4R6=M6_P4R6(1,3)+M6_P4R6(2,3)+M6_P4R6(3,3)+M6_P4R6(4,3);
67 %Compute P_P6_M6R6
68 P_P6_M6R6=P6_M6R6(1,3)+P6_M6R6(2,3)+P6_M6R6(3,3)+P6_M6R6(4,3);
69 %compute P_M8_P6P5
70 P_M8_P6P5=M8_P6P5(1,3)+M8_P6P5(2,3)+M8_P6P5(3,3)+M8_P6P5(4,3);
71 %Compute P_P8_M8
72 P_P8_M8=P8_M8(1,2)+P8_M8(2,2);
73 %compute P_M9_H9P8
74 P_M9_H9P8=M9_H9P8(1,3)+M9_H9P8(2,3)+M9_H9P8(3,3)+M9_H9P8(4,3);
75 % compute P_Fa_M9
76 P_Fa_M9= Fa_M9(1,2)+Fa_M9(2,2);
77 % P_P4_M4Bad
78 P_P4_M4Bad= P4_M4(1,2);
79 % compute B
80 B = P_P5*M8_P6P5*P_P8_M8*P_H9(1,1)*P_M9_H9P8*P_Fa_M9;
81 C= P_P6*P_M8_P6P5*P_P8_M8*P_H9(1,1)*P_M9_H9P8*P_Fa_M9;
82
83 %***** P_M4Bad_Fa*****
84 P_M4Bad_Fa = P_M4(1,1)*P_P4_M4Bad*P_R6(1,1)*P_M6_P4R6*P_P6_M6R6*B;
85 %*****
86
87 %*****P_R6=R6A_Fa*****
88 %compute P_M6_R6AP4
89 P_M6_R6AP4a=zeros(4,1);
90 for i=1:4
91 P_M6_R6AP4a(i,1)=M6_P4R6(i,2)*M6_P4R6(i,3);
92 end
93 P_M6_R6AP4=sum(P_M6_R6AP4a);
94 clear i;
    
```

Les probabilités
calculées

FIGURE 7.8 – Extrait du programme permettant le calcul des probabilités conditionnelles à partir des bases de données historiques

TABLE 7.3 – Probabilités de pertinence relatives du scénario 1

Priorité	Causes potentielles	Probabilité
Priorité 1	Machine M ₄	$P(M_4 = bad Fa) = 0.2127$
Priorité 2	Machine M ₆	$P(M_6 = bad Fa) = 0.1277$
Priorité 3	Machine M ₅	$P(M_5 = bad Fa) = 0.0978$
Priorité 4	Humain H ₉ C	$P(H_9C Fa) = 0.0877$
Priorité 5	Recette R ₆ A	$P(R_6A Fa) = 0.0827$
Priorité 6	Machine M ₉	$P(M_9 = bad Fa) = 0.0799$
Priorité 7	Machine M ₈	$P(M_8 = bad Fa) = 0.022$

sur cette machine.

Sur un plan plus quantitatif, le programme que nous avons développé est parti sur un nombre global de 10 opérations et 10 équipements de 21 nœuds. Ce nombre a été réduit à 12 candidats potentiels, soit une réduction de 42,9%. Concernant les temps de calculs, ils ont été de $Time_{localisation} = 28ms$ pour propager la suspicion dans le modèle et de $Time_{diagnostic} = 6.7 s$ pour les phases de transformation et de calculs ; la base de données sur laquelle nous nous appuyons comptant 10000 lignes et 29 colonnes.

3 Scénario 2 : cas des opérations simultanées

Le deuxième scénario présente un processus de production de produits de Type B comme le montre la figure 7.9. Le 15/12/2010 à 10h00m27s, une défaillance est détectée par l'équipement de métrologie sur le lot Q051. Nous noterons que les opérations O_4 et O_5 sont exécutées en parallèle et en interactions, l'une étant le pré-requis de l'autre et vice-versa.

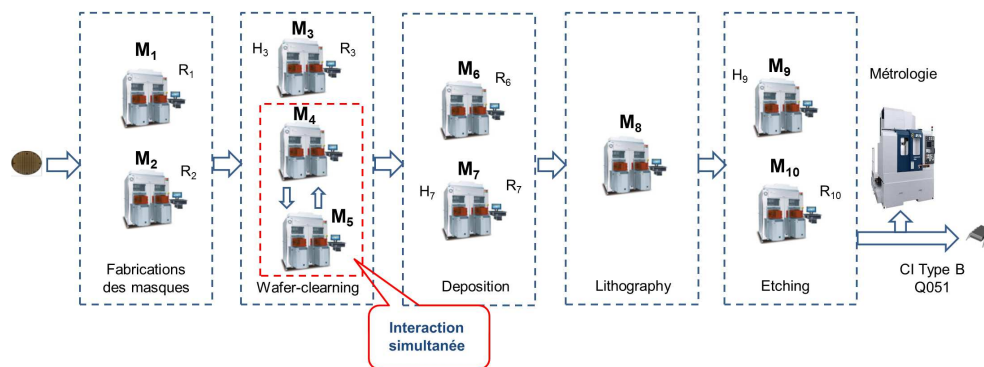


FIGURE 7.9 – Le processus de production du produit Type B lot Q051

Le même mécanisme décrit dans la section précédente est reproduit. Nous ne le reproduisons donc pas dans le détail ici mais l'illustrerons cependant dans l'extrait du programme donné dans la figure 7.9.

Le mécanisme de diagnostic est lancé par les algorithmes développés dans les chapitres précédents. Nous obtenons ici un graphe causal, voir figure 7.11 qui compose de nœuds étant origines possibles suite à l'occurrence de la défaillance de lot Q051 du produit Type B.

A l'issue de l'exécution du programme permettant de générer le modèle Bayésien, nous obtenons le graphe de la figure 7.11 dans lequel :

- $P(M_4=bad)$, $P(M_5)=bad$ sont les probabilités de M_4 , M_5 qualifiées à l'état de "bad".
- $P(R_7=R_7d)$ est la probabilité d'occurrence de $R_7=R_7d$.
- $P(M_5 | M_4)$ et $P(M_4 | M_5)$ sont les probabilités conditionnelles de M_5 sachant M_4 .
- $P(P_{45} | M_4, M_5)$ est la probabilité conditionnelle de P_{45} sachant M_4 et M_5 .
- $P(M_7 | P_{45}, R_7)$ est la probabilité conditionnelle de M_7 sachant P_{45} et R_7 .
- $P(P_7 | M_7, R_7)$ est la probabilité conditionnelle de P_7 sachant M_7 et R_7 .
- $P(M_{10} | P_7)$ est la probabilité conditionnelle de M_{10} sachant P_7 .
- $P(Fb | M_{10})$ est la probabilité conditionnelle de Fb sachant M_{10} .

```

L=length(Set_operation);
Set_non_suspect_suple=[];
for i=1:I
    OP_O(i)=[];
    OF_O(i)=[];
    for l=1:L
        O_av(l)=Set_operation(l,l);
        connexion_past =intersect(Pre_requis_O(i),Effet(O_av(l)));
        if isempty(connexion) ==0
            OP_O(i)=union(OP_O(i),O_av(l));
            OF_O_av(l)=union(OF_O_av(l),O(i));
        end
    end
end
for j=1:I
    if j>i
        connexion_present=intersect(Pre_requis_O(i), Effet(O(j)));
        if connexion_present==0
            OP_O(i)=union(OP_O(i),O(j));
            OF_O(j)=union(OF_O(j),O(i));
        end
    end
end
end
Operation_simultanee_O(i)=union(OP_O(i), OF_(O(i)));
if IC_O(i)=1
    disp('O(i) est non suspecte');
else disp('O(i) est suspecte')
    Set_non_suspect_suple=union(Set_non_suspect_suple,O(i));
end
end
end
    
```

FIGURE 7.10 – Extrait du programme relatif à la gestion des opérations simultanées

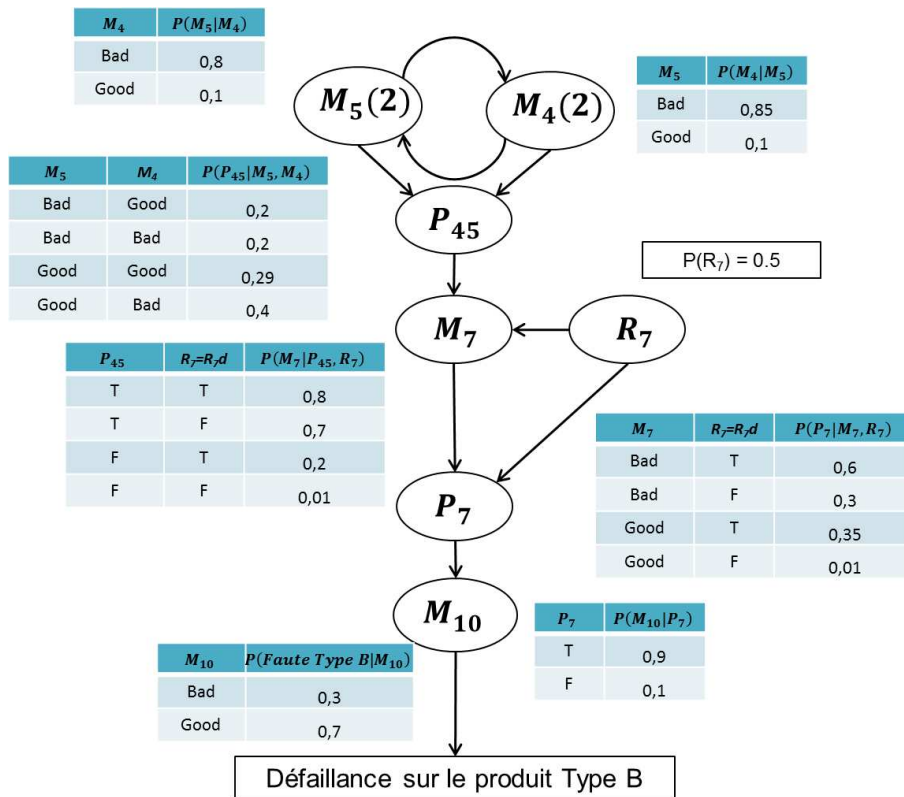


FIGURE 7.11 – Apprentissage de paramètres du scénario 2

L'interrogation du modèle obtenu donne alors les résultats suivants synthétisés dans le tableau 7.4.

TABLE 7.4 – Probabilités de pertinence relatives du scénario 2

Priorité	Causes potentielles	Probabilité
Priorité 1	Machine M_7	$P(M_7 = bad Fb) = 0.2882$
Priorité 2	Machine M_{10}	$P(M_{10} = bad Fb) = 0.1635$
Priorité 3	Recette R_{7d}	$P(R_{7d} Fb) = 0.1585$
Priorité 4	Machine M_4	$P(M_4 = bad Fb) = 0.1425$
Priorité 5	Machine M_5	$P(M_5 = bad Fb) = 0.0862$

Au terme de cette dernière étape, nous pouvons en déduire que l'origine de la défaillance est plus probablement de l'ordre de l'équipement et qu'il s'agit, fort probablement, de la machine M_7 avec un score de 0.2882. Nous conseillons à l'opérateur de maintenance d'orienter son diagnostic détaillé sur cette machine.

Sur un plan plus quantitatif, le programme que nous avons développé est parti sur un nombre global de 10 opérations et 10 équipements de 21 nœuds. Ce nombre a été réduit à 7 candidats potentiels, soit une réduction de 67.33%. Concernant les temps de calculs, ils ont été de $Time_{localisation} = 19ms$ pour propager la suspicion dans le modèle et de $Time_{diagnostic} = 4.2 s$ pour les phases de transformation et de calculs.

4 Scénario 3 : Multi-produits et multi-défaillances

Le troisième et dernier scénario présenté dans ce mémoire doit permettre l'évaluation de notre approche face à un cas multi-produits et multi-défaillances tel que représenté au sein de la figure 7.12.

Dans ce scénario, deux défaillances sont détectées à la fois sur le lot Q054 (produit de Type A) et sur le lot Q053 (produit de Type B) à 10h 24m 02s.

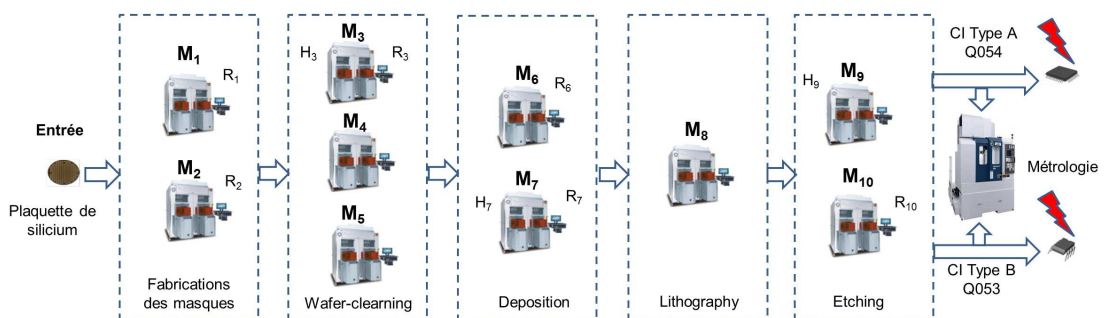


FIGURE 7.12 – Scénario 3 : multi-produits et multi-défaillances

L'exécution du mécanisme de diagnostic hybride que nous avons proposé permet alors de générer le modèle Bayésien montré dans la figure 7.13 au sein duquel :

- $P(H_1=H1b)$, $P(R_2=R2e)$ sont les probabilités de l'occurrence de H_1a et R_2b .
- $P(M_1 | H_1)$ et $P(M_2 | R_2)$ sont les probabilités conditionnelles de M_1 sachant H_1 et de M_2 sachant R_2 .
- $P(P_1 | M_1, H_1)$ est la probabilité conditionnelle de P_1 sachant M_1 et H_1 .
- $P(P_2 | M_2, R_2)$ est la probabilité conditionnelle de P_2 sachant M_2 et R_2 .
- $P(M_8 | P_1, P_2)$ est la probabilité conditionnelle entre M_8 sachant P_1 et P_2 .
- $P(P_8 | M_8)$ est la probabilité conditionnelle de P_8 sachant M_8 .
- $P(M_8 | M_{10})$ et $P(M_9 | M_{10})$ sont les probabilités conditionnelles de M_8 et M_9 sachant M_{10} .
- $P(Fa | M_9)$, $P(Fb | M_{10})$ sont les probabilités conditionnelles de Fa sachant M_9 et de Fb sachant M_{10} .

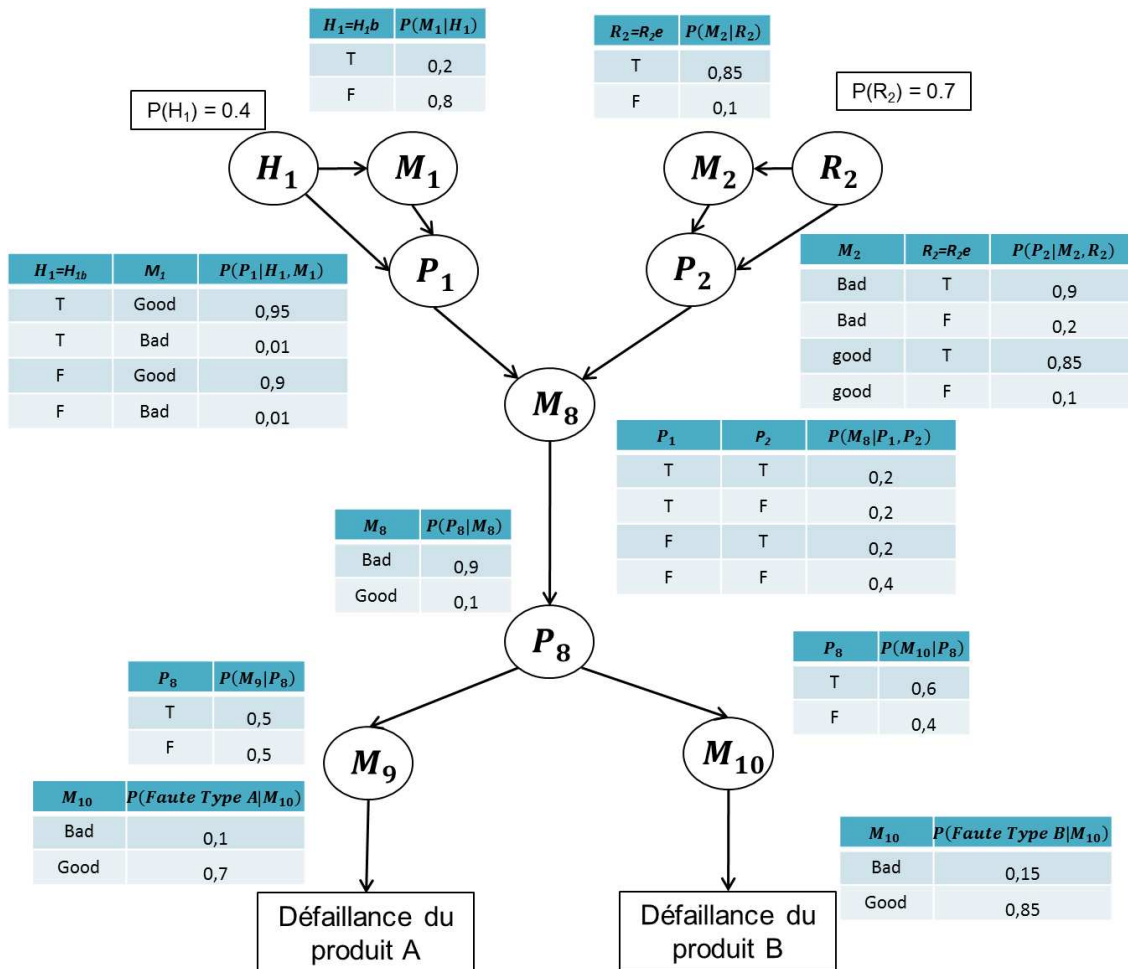


FIGURE 7.13 – Modèle Bayésien correspondant au scénario 3

Le modèle Bayésien résultant est sur le plan de sa structure cohérent aux attentes. D'une part le nœud M_8 exprime bien le phénomène de machine goulot pour les produits P_1 et P_2 , et M_9 et M_{10} ouvrent respectivement sur les deux défaillances détectées.

L'interrogation du modèle permet de calculer les probabilités de pertinence de chacune des causes. Nous pouvons en déduire que l'origine de la défaillance est plus probablement de l'ordre d'une mauvaise qualification recette et qu'il s'agit, fort probablement, de la recette R_{2e} avec un score de 0.4328. Nous conseillons à l'opérateur de maintenance de faire remonter au bureau des méthodes ce problème afin de l'approfondir en tout premier lieu.

TABLE 7.5 – Probabilités calculées du scénario 3

Priorité	Causes potentielles	Probabilité
Priorité 1	Recette R_{2e}	$P(R_{2e} Fa, Fb) = 0.4328$
Priorité 2	Machine M_1	$P(M_1 = bad Fa, Fb) = 0.2207$
Priorité 3	Recette M_{2d}	$P(M_{2d} Fa, Fb) = 0.200$
Priorité 4	Machine M_8	$P(M_8 = bad Fa, Fb) = 0.1671$
Priorité 5	Machine M_{10}	$P(M_{10} = bad Fa, Fb) = 0.1294$
Priorité 6	Machine M_9	$P(M_9 = bad Fa, Fb) = 0.1250$
Priorité 7	Humain H_{1b}	$P(H_{1b} = bad Fa, Fb) = 0.0409$

Sur un plan plus quantitatif, le programme que nous avons développé est parti sur un nombre global de 10 opérations et 10 équipements de 21 nœuds. Ce nombre a été réduit à 7 candidats potentiels, soit une réduction de 67.33%. Concernant les temps de calculs, ils ont été de $Time_{localisation} = 36ms$ pour propager la suspicion dans le modèle et de $Time_{diagnostic} = 9.2 s$ pour les phases de transformation et de calculs.

5 Discussion

L'approche que nous avons proposée permet d'allier les avantages de la recherche logique de causes et de l'approche Bayésienne. L'optimisation résultante est importante. En effet, si nous avons retenu une approche pure Bayésienne, nous nous serions heurté à une incapacité de calcul lié à la complexité du système auquel nous nous intéressons. En effet, si le système possède N nœuds, le nombre d'arcs possible est de $N!$. En outre, si chaque nœuds comprend deux états, le nombre d'états de tous les nœuds est de 2^N .

Par conséquent, le nombre total de variables est calculé par l'équation 7.1.

$$\text{nombre de variables} = \text{nombre de nœuds} + \text{nombre d'arcs} + \text{nombre d'état} = N + N! + 2^N \quad (7.1)$$

Le processus étudié dans ce chapitre (cf. la section 4) se compose de 10 opérations d'action. Si toutes opérations et leurs sous-comportements sont à analyser par un RB, la

structure complète du RB comptera 21 nœuds (toutes les opérations n'offrant pas un sous-graphe complet, i.e. opérations induites, requises, etc...). Dans ce cas, le nombre d'arcs maximum sera de $21!$, le nombre d'états est de 2^{19} , et enfin de nombre de variables serait de $5.109094217171154e+19$; un dépassement mémoire serait rapidement atteint.

L'approche proposée réduisant la taille de la structure Bayésienne à analyser, il en découle une optimisation intéressante que nous synthétisons dans le tableau 7.6 suivant sur la base d'un comparatif entre une approche Bayésienne pure et la notre appliquée aux trois scénarii.

TABLE 7.6 – La réduction de nombre de variables

	Système complet	Scénario 1	Scénario 2	Scénario 3
Nombre de nœuds	21	12	10	12
Nombre d'arcs	$21!$	12	10	13
Nombre de variables	$5.1090942171711e+19$	56	46	52

L'approche proposée permet ainsi de réduire de manière significative l'ensemble des origines possibles à évaluer. Enfin, afin nous avons mené pour les 3 scénarios présentés une étude comparative permettant de donner une idée des temps de calculs et taux d'occupation mémoire relatifs à notre cas d'étude. Ils sont synthétisés dans le tableau 7.7 :

TABLE 7.7 – Temps de calcul et mémoire RAM utilisée

	Scénario 1	Scénario 2	Scénario 3
CPU Time (s)	6.7	4.2	9.3
Mémoire utilisée (byte)	$8.69351e+008$	$9.04462e+008$	$8.67258e+008$

6 Conclusion

Dans ce chapitre, nous avons mené notre approche jusqu'à sa validation sur la base d'un cas d'étude largement inspiré d'un cas réel issu de l'industrie du semi conducteur. Il s'est agi notamment de prouver que les modèles ainsi que les algorithmes proposés répondent à nos attentes. Ces derniers ont été développés sous Matlab. Les toolbox du domaine ont été volontairement écartées de l'activité de codage afin de privilégier la maîtrise des développements actuels et évolutions et adaptations futures. L'ensemble a été évalué sur différents scénarios.

Les résultats obtenus à partir des scénarios ont permis de vérifier que :

- le mécanisme de propagation fonctionne, même en présence d'opérations simultanées,
- la structure du graphe causal est transformée à partir du modèle historique, s'affranchissant ainsi d'un apprentissage traditionnellement long,
- d'autres causes que celles traditionnellement centrées sur les équipements sont désormais prises en compte,

-
- le raisonnement Bayésien permet de classer les causes par priorité d'intervention, facilitant et optimisant ainsi l'aide au diagnostic pour les opérateurs,
 - de par l'hybridation, les temps de calcul permettent désormais une application à un contexte fortement stressé par le produit (forte variabilité et cycle de vie court),
 - l'approche est adaptée au contexte multi-défaillances.

Ces résultats nous encouragent ainsi à pousser cette évaluation désormais plus loin, à savoir en situation réelle sur un atelier semi-conducteur.

Conclusion générale

Les travaux que nous avons présentés dans ce document traitent du diagnostic des procédés industriels complexes placés dans un environnement stressé par une forte variabilité produit, à court cycle de vie maximisant les dérives. Ils font suite à ceux déjà réalisés par (Deschamps, 2007) et (Duong et al., 2012a) dans ce domaine au sein de l'équipe Gestion et Conduite des Systèmes de Production (GCSP) du Laboratoire des Sciences pour la Conception, l'Optimisation et la Production (G-SCOP) de Grenoble.

La contribution principale de nos travaux réside dans la proposition d'une approche basée sur l'hybridation d'une approche de diagnostic logique et d'une approche probabiliste Bayésienne de défaillances propagées au travers du produit. Au delà des extensions que nous avons proposées en terme de modélisation des opérations pour prendre en compte d'autres causes que celles liées aux dérives équipement, comme le facteur humain, les recettes ou encore la matière première, nous avons développé une démarche qui permet progressivement de transformer le modèle initial en une structure Bayésienne support à l'aide à la maintenance. De ce point de vue, l'approche proposée, novatrice dans le domaine, a donné lieu à des résultats significatifs. Ils se caractérisent comme suit :

- L'approche est centrée sur une acquisition de la connaissance hors ligne basée sur l'étude du fonctionnement normal des capacités opératoires offertes par le système considéré. Le bureau d'étude est dirigé par une méthodologie orientée par des fiches opératoires définies en termes de pré-requis, de dates, d'effets finaux escomptés. La non prise en compte des comportements anormaux en fait une approche réaliste pour les systèmes complexes.
- Un mécanisme de génération en ligne d'un modèle d'historique permet de conserver, et ce à chaque « run », mémoire des opérations sollicitées.
- Un mécanisme de réduction du modèle historique permet « d'oublier » les opérations dont les effets ont pu être vérifiés par métrologie. Ce mécanisme garantit une certaine maîtrise de la taille du modèle et résout le problème posé par les opérations dites simultanées interdépendantes. Ainsi, à chaque étape du processus de production, le modèle résultant représente l'ensemble des causes potentielles en cas de défaillances. Si une défaillance venait à être détectée, et sous réserve d'avoir pris en compte suffisamment de facteurs, le coupable s'y trouve.
- En présence d'une défaillance propagée par le produit et détectée sur un équipement de métrologie (non-conformité produit), un mécanisme de propagation de contraintes arrière permet d'isoler encore les causes en rejetant les opérations qui n'ont pas été visitées par le produit incriminé. Nous avons pu montrer que ce mécanisme est fort peu gourmand en temps, s'agissant d'un simple processus de propagation de contraintes.

- Dès lors, nous avons imaginé un processus de transformation du modèle réduit en une structure graphique Bayésienne, s'affranchissant ainsi d'un apprentissage structurel illusoire tant sur le plan temporel dans un procédé complexe que sur le plan de la connaissance lorsqu'il s'agit d'un contexte soumis à forte variabilité produit.
- Un mécanisme classique de calcul des tables de probabilités conditionnelles est ensuite mené pour finaliser le modèle Bayésien. Les bases de données considérées ici sont pré-traitées et cohérentes.
- Un mécanisme classique d'exploitation du réseau a été développé afin de pouvoir questionner le modèle sur les probabilités de pertinence des causes. L'approche se veut donc facilement intégrable en milieu industriel en proposant aux opérateurs de maintenance des niveaux de priorités (scores) affectés à chacune des causes potentielles évaluées.
- Enfin, et sur un plan technique, nous avons développé l'ensemble de l'approche dans l'environnement Matlab-Simulink afin de conserver maîtrise et développements futurs de la plate-forme de diagnostic.

Au terme de ces travaux, plusieurs axes de recherche se dégagent pour envisager, du point de vue des perspectives, de prolonger l'étude menée pendant ces trois ans.

A court terme, quatre axes d'investigations peuvent être envisagés :

- Premièrement, et sur la base de l'atelier logiciel développé, une intégration sur site réel devra être envisagée de manière à valider notre approche. Cette validation devra être menée de manière progressive et en collaboration étroite avec les équipes maintenance et IT de l'entreprise. Les retours d'expériences devront contribuer à l'affinement des modèles proposés. La mise en exergue d'autres causes à prendre en compte pourra être envisagée, cela ne remettant pas en cause la généralité de l'approche proposée. Cette démarche pourrait être portée dans un premier temps dans le cadre d'un master de recherche poursuivi éventuellement par un diplôme de recherche et d'innovation.
- Deuxièmement, nous devons envisager de lever l'hypothèse de bon fonctionnement des équipements de métrologie. En effet, ces derniers ne sont pas exempts de défaillances, en particulier au niveau de leur système de captage. Il faudra dans ce cas mener une analyse fine de ce type d'équipements qui ne sont pas soumis aux mêmes contextes de fonctionnement que ceux de production et ainsi élaborer les modèles appropriés.
- Troisièmement, le développement informatique que nous avons mené dans le cadre de cette thèse mériterait d'être repris à des fins d'optimisation des temps de cycle.
- Quatrièmement, l'étude des performances que nous avons menée sur la base de trois scénarii reste évidemment à conforter tant par la pratique sur des systèmes plus complexes encore que par une étude théorique poussée de la complexité algorithmique.
- Enfin, et sur un plan technique, une intégration en tant que fonction avancée de supervision devrait être étudiée avec des fournisseurs de SCADA et/ou de solutions MES (Manufacturing Execution Systems).

A moyen terme, nous pouvons mettre en exergue au moins quatre orientations de recherches.

- Premièrement, l'approche que nous avons développée s'est centrée uniquement sur le diagnostic de causes. Dans (Deschamps, 2007), des solutions étaient mises en perspectives pour développer, sur la base d'une propagation de contraintes avant un mécanisme de pronostic des conséquences des défaillances. Il serait pertinent d'étendre l'approche que nous proposons ici à cette activité d'analyse de conséquences. Il en résulterait d'autres indicateurs permettant d'assister encore davantage la prise de décision.
- Une deuxième perspective consisterait à exploiter les approches de génération de modèles et de réduction afin de déterminer un placement des capteurs optimal vis à vis du critère de diagnosticabilité des services. Le système de captage a une influence directe sur la taille du modèle résultant.
- Troisièmement, nous avons pris pour hypothèse dans ce manuscrit que le produit n'impactait ni l'opérateur humain ni l'environnement pendant l'exécution d'une opération. Cette hypothèse doit être levée dans le cadre de sites industriels à risques manipulant des produits dangereux. L'extension de notre approche à ce type de problème en collaboration étroite avec des spécialistes du risque serait une voie intéressante à explorer et un terrain d'applicabilité de nos travaux de recherche.
- Enfin, bien que l'étude de l'impact de la structure hiérarchique sortait du cadre de nos travaux, nous avons souligné à plusieurs reprises que cette hypothèse atteignait ces limites dans le cadre d'un fonctionnement anormal. Il s'avère nécessaire d'étendre l'approche développée dans le cadre d'une structure mixte (hiérarchique et distribuée). Ainsi, le module à l'origine de la prise en compte du dysfonctionnement pourrait prendre la main sur le processus de diagnostic non seulement en interne mais également sur l'ensemble des autres modules concernés. Ceci nous semble tout à fait réaliste compte tenu de la prise en compte dans chacun des modules de coordination, des opérations requises. En ce sens nous rejoignons les travaux développés dans le cadre du diagnostic distribué par M. Marcos Dasilveira (Da Silveira et al., 2003).

Par ailleurs, à plus long terme, il faudrait envisager l'étude de la pertinence de l'approche proposée dans des contextes de natures différentes, comme par exemple les réseaux de distribution électriques et les systèmes embarqués.

Bibliographie

- Abdelwahed S., Karsai G., Mahadevan N., Ofsthun S.,C. (2009). Practical implementation of diagnosis systems using timed failure propagation graph models. *Instrumentation and Measurement, IEEE Transactions on*, 58(2) : pp. 240–7.
- AFNOR (12/1991). Norme nf x50-151 - "analyse de la valeur, analyse fonctionnelle - expression fonctionnelle du besoin et cahier des charges fonctionnel".
- Abu-Samah, A., Shahzad, M. K., Zamai, E., et Hubac, S. (2015). Effective Maintenance by Reducing Failure–Cause Misdiagnosis in Semiconductor Industry (SI), *PHM Journal*.
- Atoui, M. A., Verron, S., et Kobi, A. (2014). Conditional Gaussian Network as PCA for fault detection. *The International Federation of Automatic Control, In World Congress*, 19(1), pp. 1935–1940.
- Baweja, G. et Ouyang, B. (2000) Cim strategy for semiconductor fab-building blocks approach. In *The Ninth International Symposium on Semiconductor Manufacturing. Proceedings of ISSM*, pp. 241 –244.
- Berruet, P. (1998). *Contribution au recouvrement des systèmes flexibles de production manufacturière : analyse de la tolérance et reconfiguration*. PhD thesis, L'université des sciences et techniques de Lille.
- Berruet, P., Lallican, J. L., Rossi, A., et Philippe, J. L. (2005). A component based approach for the design of FMS control and supervision. In *Systems, Man and Cybernetics, IEEE International Conference on*, Vol. 4, pp. 3005–3011. IEEE.
- Bettayeb, B. et Bassetto, S.J. (2014). Effects of process learning and product lifecycle on risk-based quality control plans. *Systems Conference (SysCon), 8th Annual IEEE*, pp. 505–510.
- Bichon, B., J., McFarland, J., M., et Mahadevan, S. (2011). Efficient surrogate models for reliability analysis of systems with multiple failure modes. *Reliability Engineering & System Safety*, 96(10), 1386–1395.
- Bouaziz, M. F., Zamai, E., Duvivier, F., et Hubac, S. (2011). Dependability of complex semiconductor systems : Learning bayesian networks for decision support. In *The 3rd International Workshop on Dependable Control of Discrete Systems (DCDS)*, pp. 7–12.
- Boulanger, J. L. (2010) «Principles.» *Safety of Computer Architectures, ISTE–WILEY*, pp. 1–46.
- Bourdon, J. (2012) *Sources Probabilistes : des séquences aux systèmes*. Thèse de doctorat, Université de Nantes, 2012.

- Chabir, K., Sid, M. A., et Sauter, D. (2014). Fault diagnosis in a networked control system under communication constraints : A quadrotor application. *International Journal of Applied Mathematics and Computer Science*, 24(4), pp. 809–820.
- CIM (1989). A reference model for computer integrated manufacturing from the viewpoint of industrial automation. In *International Journal of Computer Intergrated Manufacturing*, vol2, pp. 114–127.
- Combacau, M., Burruet, P., Zamai, E., Charbonnaud, P., et Khatab, A. (2000). Supervision and monitoring of production systems. In *The IFAC 2nd Conference on Management and Control of Production and Logistics (MCPL'00)*, Grenoble, France.
- Conrady, S. et Jouffe, L. (2013). *Tutorial on Driver Analysis and Product Optimization with BayesiaLab*.
- Crama, Y., Alwin G. O., et Frits S. (2012). Production planning in automated manufacturing. *Springer Science & Business Media*.
- Da Silveira, M., Combacau, M., et Boufaied, A. (2003). Prognosis and recovery evaluation in flexible manufacturing supervision. *Journal of Decision Systems*, 12(1), pp. 93–109.
- Dejaeger, K., Verbraken, T., et Baesens, B. (2013). Toward comprehensible software fault prediction models using bayesian network classifiers. *Software Engineering, IEEE Transactions on*, 39(2), pp. 237–257.
- Dempster, A. P., Laird, N. M., et Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), pp. 1–38.
- Deschamps, E. (2007). *Diagnostic de services pour la reconfiguration dynamique de systèmes à événements discrets complexes*. PhD thesis, Laboratoire des Sciences pour la Conception, l'Optimisation et la Production de Grenoble (G-SCOP).
- Dhillon, B.S.(2002). Design reliability : Fundamentals and Applications. *CRC press*.
- Dhillon, B. (2005). Dynamic reliability evaluation models. In Dhillon, B., editor, *Reliability, Quality, and Safety for Engineers*. Taylor & Francis.
- Doggett, A.M. (2005). A statistical comparison of three root cause analysis tools. *Journal of Industrial Technology*, 20(2), pp. 2–9.
- Ditlevsen, O. (1982). Model unvertainty in structural reliability. *Structural Safety*, vol 1, pp. 73–86.
- Dubuisson, D. (2001). Diagnostic, intelligence artificielle et reconnaissance des formes. *Hermès science publications*.
- Duong, Q.-B., Zamai, E., et Tran-Dinh, K.-Q. (2012). Confidence estimation of feedback information for logic diagnosis. *The International Journal of Intelligent Real-Time Automation, Engineering Applications of Artificial Intelligence (EAAI)*, 26(3), pp. 1149 – 1161.

-
- Duong, Q. B. (2012). Approche probabiliste pour l'estimation dynamique de la confiance accordée à un équipement de production : vers une contribution au diagnostic de services des SED (Doctoral dissertation, Université de Grenoble ; 243 University of Danang).
- Duret, D et Pillet, M. (2011) Qualité en production : de l'ISO 9000 à Six Sigma. *Editions Eyrolles*.
- Duval, C., Fallet–Fidry, G., lung, B., Weber, P., et Levrat E. (2012). A Bayesian network-based integrated risk analysis approach for industrial systems : application to heat sink system and prospects development. *Proceedings of the Institution of Mechanical Engineers, Part O : Journal of Risk & Reliability*.
- Erickson, K.T. et Hedrick, J.L. (1999). Plant-Wide Process Control. *John Wiley and Sons*.
- Flaus, J. M. (2013). Risk Analysis : Socio–technical and Industrial Systems. *John Wiley & Sons*.
- Fuster–Parra, P., Garcia-Mas, A., Ponseti, F. J., Palou, P., et Cruz, J. (2014). A Bayesian network to discover relationships between negative features in sport : a case study of teen players. *Quality and Quantity*, 48(3), pp. 1473–1491.
- García, A. et Gilabert, E. (2011). Mapping FMEA into Bayesian Networks. *International Journal of Performability Engineering*.
- Genrich, H.J., Hanisch, H.M., et Wollhaf, K. (1994). Verification of recipe-based control procedures by means of predicate/transition nets. *Application and Theory of Petri Nets*. Springer, pp. 278–297.
- Groover, Mikell P. (2007). *Automation, production systems, and computer–integrated manufacturing*. Prentice Hall Press.
- Hamani, N., Dangoumau, N., et Craye, E. (2009). Verification and validation of a SSM model dedicated to mode handling of flexible manufacturing systems. *Computers in Industry*, 60(2), pp. 77–85.
- Henry, S. (2005). *Synthèse de Lois de Commande pour la Configuration et la Reconfiguration des Systèmes Industriels Complexes*. PhD thesis, Institut National Polytechnique de Grenoble.
- Henry, S., Zamai, E., et Jacomino, M. (2012). Logic control law design for automated manufacturing systems. *Engineering Applications of Artificial Intelligence*, 25(4), pp. 824–836.
- Heras-Saizarbitoria I. et Boiral O. (2013) ISO 9001 and ISO 14001 : Towards a research agenda on management system standards. *International Journal of Management Reviews*, 15(1), pp. 47–65.
- Hill, T. et Nettles, S. (1997). Advanced process control framework initiative. In SPIE, editor, *Plug and Play Software for Agile Manufacturing*, vol 304 of *Proc. SPIE 2913*.

- Hubac, S., Duvivier, F., Zamai, E., et Mili, A. (2010). Predictive maintenance supported by advanced process control (apc) opens new equipment engineering and manufacturing opportunities. In *The 21st Advanced Semiconductor Manufacturing Conference, ASMC 2010, San Francisco*.
- Hubac, S. et Zamai, E. (2013). Politiques de maintenance équipement en flux de production stressant. *Techniques de l'ingénieur Méthodes de production base documentaire : TIB521DUO*.
- IMPROVE.(2012). Official website in www.eniac-improve.eu.
- Isermann, R. (2006). Fault–diagnosis systems : an introduction from fault detection to fault tolerance. *Springer Science and Business Media*.
- Ishikawa, K. (1990). Introduction to quality control. *Productivity Press*.
- Joshi, S. et Smith, J. (2012) Computer control of flexible manufacturing systems : research and development. *Springer Science & Business Media*.
- Kempowsky, T., Subias, A. et Aguilar, J. (2004). Supervision of complex processes : strategy for fault detection and diagnosis. *IFAC MCPL (Conference on Management of Control of Production and Logistics)*, Santiago, Chili.
- Khomfoi, S. et Tolbert, L.M. (2007) Fault diagnostic system for a multilevel inverter using a neural network. *Power Electronics, IEEE Transactions on*, 22(3), pp. 1062–9.
- Kobbacy, K.A.H. et Murthy, D.P. (2008). Complex system maintenance handbook. *Springer Science and Business Media*.
- Lefebvre, D. (2012). Diagnosis with Petri nets according to partial events and states observation. In *Fault Detection, Supervision and Safety of Technical Processes*, 8(1), pp. 1244–1249.
- Li, B., Han, T. et Kang, F. (2013). Fault diagnosis expert system of semiconductor manufacturing equipment using a Bayesian network. *International Journal of Computer Integrated Manufacturing*, 26(12), pp. 1161–1171.
- Liu, Z., Li, Y.F., He, L.P., Yang, Y.J., et Huang, H.Z. (2014). A new fault tree analysis approach based on imprecise reliability model. *Proceedings of the Institution of Mechanical Engineers, Part O : Journal of Risk and Reliability*.
- Manfredotti, C.E. (2009). *Modeling and inference with relational dynamic bayesian network*. PhD thesis, Università Di Milano–Bicocca.
- Marquez, D., Neil, M., et Fenton, N. (2008) Solving dynamic fault trees using a new hybrid bayesian network inference algorithm. *Control and Automation, 16th Mediterranean Conference on IEEE*, pp. 609–14.
- Martinez, V. et Edgar, T. (2006). Control of lithography in semiconductor manufacturing. *Control Systems, IEEE*, 26(6), pp. 46–55.

-
- Mahulea C., Seatzu C., Cabasino M. P., et Silva M. (2012). Fault diagnosis of discrete-event systems using continuous Petri nets. *Systems, Man and Cybernetics, Part A : Systems and Humans, IEEE Transactions on*, 42(4), pp. 970–84.
- Maimon O., Z. et Rokach L. (2005). *Data mining and knowledge discovery handbook*.
- Merle, Guillaume, Jean-Marc Roussel et Jean-Jacques Lesage (2010). Improving the Efficiency of Dynamic Fault Tree Analysis by Considering Gate FDEP as Static. *European Safety and Reliability Conference*. Taylor & Francis.
- Mili, A. (2009). *Vers Des Méthodes Fiables De Contrôle Des Procédés Par La Maîtrise Du Risque - Contribution À La Fiabilisation Des Méthodes De Process Control D'une Unité De Recherche Et De Production De Circuits Semi-Conducteurs*. Génie industriel, L'Institut polytechnique de Grenoble.
- Millan, E., Descalço, L., Castillo, G., Oliveira, P., et Diogo, S. (2013). Using Bayesian networks to improve knowledge assessment. *Computers and Education*, 60(1), pp. 436–447.
- Moussa Ali, A. et Zhang, Q. (2014). An innovations approach to fault diagnosis in linear time-varying descriptor systems. *In Control Conference (ECC), European*, pp. 1287–1292.
- Munoz, E., Capón-García, E., Moreno-Benito, M., Espuna, A., et Puigjaner, L. (2011). Scheduling and control decision-making under an integrated information environment. *Computers and Chemical Engineering*, 35(5), pp. 774–786.
- Naïm, P., Wullemmin, P.-H., Leray, P., Pourret, O., et Becker, A. (2011). *Réseaux bayésiens*. Editions Eyrolles.
- Neapolitan R., E. (2012) Probabilistic reasoning in expert systems : theory and algorithms. *CreateSpace Independent Publishing Platform*.
- Nguyen, D. T., Duong, Q. B., Zamai, E. et Shahzad, M. K (2015). Fault diagnosis for the complex manufacturing system , in the International Journal of Risk and Reliability. (soumis).
- Nguyen, D. T., Duong, Q. B., Zamai, E. et Shahzad, M. K (2014a). Dynamic structure identification of Bayesian network model for fault diagnosis of FMS. *In The 40th Annual Conference of the IEEE Industrial Electronics Society*.
- Nguyen, D. T., Duong, Q. B., Zamai, E. et Shahzad, M. K (2014b). Bayesian network model with dynamic structure identification for real time diagnosis. *In Emerging Technology and Factory Automation (ETFa)*.
- Nguyen, D. T., Duong, Q. B., Zamai, E. et Shahzad, M. K (2014c). Diagnosis in complex system with multiple failure sources. *In International Conference on Modeling, Optimization and Simulation-MOSIM14*.
- Niel, E. et Craye, E. (2002). *Maîtrise des risques et sûreté de fonctionnement des systèmes de production*. Hermès-Lavoisier.
- Papadopoulos, C. T., O'Kelly, M. E., Vidalis, M. J., et Spinellis, D. (2009). Analysis and design of discrete part production lines. *New York : Springer*, pp. 279

- Pendola, M. (2000). *Fiabilité des structures en contexte d'incertitudes statistiques et d'écarts de modélisation*. PhD thesis, Université de Blaise Pascal, Clermont-Ferrand II.
- Perrin, J., Binet, F., Dumery, J., Merlaud, C., et Trichard, J. (2004). *Automatique et informatique industrielle : Bases théoriques, méthodologiques et techniques*. Nathan Technique.
- Philippot, A., Marangé, P., Gellot, F., Pétrin, J. F., et Riera, B. (2014) Fault Tolerant Control for Manufacturing Discrete Systems by Filter and Diagnoser Interactions. *Annuaire conference of The Prognostics and Health Management Society*.
- Phillips E., J. (1997). Manufacturing Plant Layout : Fundamentals and fine points of optimum facility design. *Society of Manufacturing Engineers*.
- Proctor, R.W. et Van Zandt, T. (2011). Human factors in simple and complex systems. *CRC press*.
- Quirk, M. et Serda, J. (2001). *Semiconductor Manufacturing Technology*. ISBN : 0130815209. Prentice Hall.
- Rao, K. D., Gopika, V., Rao, V. S., Kushwaha, H. S., Verma, A. K., et Srividya, A. (2009) Dynamic fault tree analysis using Monte Carlo simulation in probabilistic safety assessment. *Reliability Engineering & System Safety*, 94(4) pp. 872–883.
- Rasmuson D., M. et Kelly D., L. (2008) Common cause failure analysis in event assessment. *Proceedings of the Institution of Mechanical Engineers, Part O : Journal of Risk and Reliability*, 222(4), pp. 521–532.
- Renooij, S. (2001). Probability elicitation for belief networks : issues to consider. *The Knowledge Engineering Review*, 16(03), pp. 255–269.
- Ribot, P., Pencolé, Y., et Combacau, M. (2009). Diagnosis and prognosis for the maintenance of complex systems. In *IEEE International Conference on Systems, Man and Cybernetics, SMC'09*, pp. 4146–4151.
- Robinson, R. W. (1997). Counting unlabeled acyclic digraphs. In *Combinatorial mathematics. Springer Berlin Heidelberg*, pp. 28–43.
- Sahin, S., Tolun, M. R., et Hassanpour, R. (2012) Hybrid expert systems : A survey of current approaches and applications. *Expert Systems with Applications*, 39(4), pp. 4609–4617.
- Said, A. B., Shahzad, M. K., Zamai, E., Hubac, S., et Tollenaere, M. (2014). A Bayesian network based approach to improve the effectiveness of maintenance actions in Semiconductor Industry. In *Second European Conference of the Prognostics and Health Management Society*.
- Schaller R. (1997) Moore's law : Past, present and future. *IEEE Spectrum*, 34, pp. 52–59.
- Scheer A-W. (2012). CIM Computer Integrated Manufacturing : Towards the Factory of the Future. *Springer Science & Business Media*.
- Sematech (1997). Advanced process control framework initiative (apcfi) 1.0. specifications. Technology Transfer 97063300A-ENG, Sematech.

-
- Shahrabi, M. et Shojaei, A.A. (2014). Application of FMEA and AHP in Lean Maintenance. *Int. J. Modern Eng. Sci*, 3(1), pp. 61–73.
- Shahzad, M.K., Hubac, S., Siadat, A., et Tollenaere, M. (2012). An interdisciplinary fmea methodology to find true dfm challenges. In *12th European Advanced Process Control and Manufacturing Conference*, Grenoble, France.
- Soldani, S., Combacau, M., Thomas, J., et Subias, A. (2006). Intermittent fault detection through message exchanges : a coherence based approach. In *Fault Detection, Supervision and Safety of Technical Processes*, 6(1), pp. 1479–1484.
- Sonderman, T. et Spanos, C. (2005). Advanced process control in semiconductor manufacturing. A webcast lecture - the presentation. *Computing and Systems Technology*, CAST.
- Strasser, S. et Sheppard, J. (2011). Diagnostic alarm sequence maturation in timed failure propagation graphs. *AUTOTESTCON, IEEE* , pp. 158–65.
- Tayarani–Bathaie, S. S., Vanini, Z. S., et Khorasani, K. (2014). Dynamic neural network-based fault diagnosis of gas turbine engines. *Neurocomputing*, 125, pp. 153–165.
- Tay, M. K., Mekhnacha, K., Chen, C., Yguel, M., et Laugier, C. (2008) An efficient formulation of the bayesian occupation filter for target tracking in dynamic environments. *International Journal of Vehicle Autonomous Systems*, 6(1–2), pp. 155–171.
- Toguyei, A. K. A., Berruet, P., et Craye, E. Models and algorithms for failure diagnosis and recovery in FMSs. *International Journal of Flexible Manufacturing Systems*, 2003 ; 5(1), 57–85.
- Tran, V.T., Yang, B.S., Oh, M.S., et Tan, A.C.C. (2009). Fault diagnosis of induction motor based on decision trees and adaptive neuro–fuzzy inference. *Expert Systems with Applications*, 36, pp. 1840–1849.
- Trentesaux, D, Berger, T., et Sallez, Y. Product and resource driven control of allocation and routing processes in FMS. In *7e Conférence Internationale de MOdélisation et SIMulation*, Paris, France, 2008.
- Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., and Yin, K. (2003). A review of process fault detection and diagnosis : Part III : Process history based methods. *Computers and chemical engineering*, 27(3), pp. 327–346.
- Villemeur, A. (1992). Reliability, Availability, Maintainability and Safety Assessment, *Assessment, Hardware, Software and Human Factors*, Wiley, Vol 2.
- Viveros, P., Zio, E., Nikulin, C., Stegmaier, R., et Bravo, G. (2014). Resolving equipment failure causes by root cause analysis and theory of inventive problem solving. *Proceedings of the Institution of Mechanical Engineers, Part O : Journal of Risk & Reliability*, 228(1) : 93–111.
- Weber, P., Theilliol, D., et Aubrun, C. (2008). Component reliability in fault-diagnosis decision making based on dynamic Bayesian networks. *Proceedings of the Institution of Mechanical Engineers, Part O : Journal of Risk & Reliability*, 222(2), pp. 161–172.

- Weber, P., Medina–Oliva, G., Simon, C., et lung, B. (2012). Overview on Bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, 25(4) : pp. 671–82.
- White, J., Benavides, D., Schmidt, D.C., Trinidad, P., Dougherty, B., et Ruiz-Cortes, A. (2010) Automated diagnosis of feature model configurations. *Journal of Systems & Software*, 83(7), pp. 1094–1107.
- Whiteley, M., Fly, A., Leigh, J., Jackson, L., et Dunnett, S. (2015). Reliability modelling using Petri-Net simulation of polymer electrolyte membrane fuel cell degradation in automotive applications. In *IEEE Reliability and Maintainability Symposium (RAMS)*, Annual pp. 1–6.
- Yang, B., S., Oh, M.S., et Tan, A.C.C. (2009). Fault diagnosis of induction motor based on decision trees and adaptive neuro-fuzzy inference. *Expert Systems with Applications*, 36(2) : pp. 1840–9.
- Zwingelstein, G. (1995). Diagnostic des défaillances : Théorie et pratique pour les systèmes industriels. In *Hermes Science Publications*.

Annexe A

Notions de graphe

La structure d'un réseau Bayésien où la structure graphique du modèle historique que nous manipulons dans ce mémoire s'apparente à un graphe orienté modélisant des relations de causes à effets. Aussi, dans cette annexe, nous proposons quelques notions de base concernant ces graphes.

Arbres et Polyarbres : Une arborescence est un graphe orienté sans circuits dans lequel chaque nœud possède au plus un parent ; le graphe non-orienté sous-jacent à une arborescence est appelé arbre. Il est caractérisé par le fait qu'il ne contient pas de cycle. Plus généralement, un graphe orienté tel que, s'il existe un chemin entre deux nœuds, celui-ci est unique, est appelé polyarbre.

La figure A.1 illustre ce qu'est un polyarbre. Dans ce cas, la propagation de l'information est plus complexe, car l'information peut circuler d'un parent à un autre.

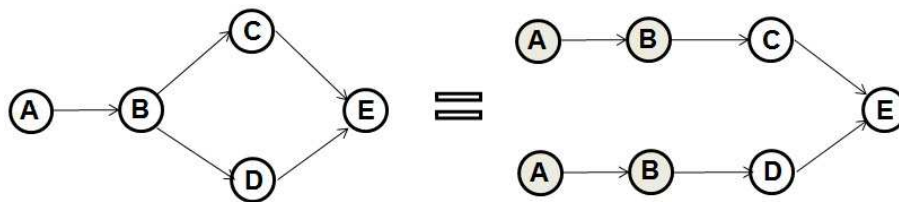


FIGURE A.1 – Exemple d'un polyarbre

Selon la figure A.1, nous pouvons écrire que :

$$P(E|A) = \sum_B P(E|B, A).P(B \text{ vert } A) \quad (\text{A.1})$$

Définition 10. Graphe de regroupement

Soit V un ensemble fini. Nous notons $\mathcal{V} \subset P(V)$ un ensemble de parties de V . Nous dirons que $G = (V, A)$ est un graphe de regroupement sur V si et seulement si :

- . G est un graphe sur V .

$$\cdot \bigcup_{u \in V} = V.$$

Définition 11. Arbre de regroupement

Si $G = (V, A)$ est un graphe de regroupement sur V , nous dirons que H est un arbre de regroupement sur V si et seulement si :

- H est un arbre sur V .
- Pour tout couple $(u, v) \in V \times V$ connecté par l'unique chemin $[u, v_0, v_1, \dots, v_n, v]$, nous avons la propriété : $\forall 0 \leq i \leq n; u \cap \subset v_i$

Selon la définition 11, nous pouvons alors écrire les équations suivantes :

$$H = \text{Regroupe}(G) \tag{A.2}$$

Regroupe est la fonction de regroupement du graphe G au graphe H .
 G est un graphe de dégroupement de H et il est illustré par l'équation suivante :

$$G = \text{Regroupe}^{-1}(H) \tag{A.3}$$

Regroupe^{-1} est la fonction de dégroupement du graphe H au graphe G .

Les techniques de graphe sont utilisées dans ce mémoire pour transformer les modèles historiques et Bayésiens (cf. Chapitre 5). Les fonctions de regroupement et dégroupement permettent de conserver les informations internes des nœuds après la transformation et l'identification de la structure du RB.

Annexe B

Notions de Probabilités

Probabilité : soit Ω un ensemble fini non vide, $(\varepsilon, \cap, \cup)$ une algèbre sur Ω ($\varepsilon \in 2^\Omega$, l'ensemble des parties de Ω). Soit $P : \varepsilon \rightarrow [0, 1]$ une fonction à valeurs réelles. P est une probabilité sur (Ω, ε) si et seulement si elle vérifie que :

- $\forall A \in \varepsilon, 0 \leq P(A) \leq 1$;
- $\forall A, B \in \varepsilon [A \cap B = \emptyset] \Rightarrow P(A \cup B) = P(A) + P(B)$. A et B sont alors dits mutuellement exclusifs ;
- $P(\Omega) = 1$ et donc $P(\emptyset) = 0$.

Variable aléatoire : une variable aléatoire est une fonction X définie sur Ω :

$$\begin{aligned} X : \Omega &\longrightarrow D_X \\ \omega &\mapsto X(\omega) \end{aligned}$$

Pour $x \in D_X$, nous notons alors $X = x$ l'événement $\omega \in \Omega \mid X(\omega) = x$. D_x est le domaine de définition de X .

Probabilités jointes : la probabilité jointe entre deux variable aléatoire X et Y est représentée par $P(x, y) = p(X = x, Y = y)$, pour $x \in D_X$, et $y \in D_Y$. Si X et Y sont indépendants, nous obtenons : $P(x, y) = p(x).p(y)$. Si la valeur de y est connue, la probabilité de $X = x$ dans ce cas est calculée par $P(x \mid y) = P(X = x \mid Y = y)$. Appelée également probabilité conditionnelle, elle est représenté par :

$$p(x \mid y) = \frac{P(x, y)}{p(y)} \tag{B.1}$$

Si X et Y sont indépendantes, nous pouvons obtenir :

$$p(x \mid y) = \frac{P(x, y)}{p(y)} = p(x) \tag{B.2}$$

Probabilités marginales : la donnée d'une probabilité jointe d'ensemble de variables permet de retrouver la probabilité jointe de chacun de ses sous-ensembles. C'est ce que nous appelons une probabilité marginale. Par conséquent, si deux ensembles de variables X et Y sont disponibles, la règle de marginalisation :

$$P(X) = \sum_y P(X, y) = \sum_y P(X | y) \cdot P(y) \quad (\text{B.3})$$

Théorème de Bayes : le théorème de Bayes est plus qu'un théorème opératoire. Il est à la base de tout un pan de la statistique ; la statistique bayésienne. Il est défini comme suit :

$$P(a | b) = \frac{P(b | a) \cdot P(a)}{P(b)} \quad (\text{B.4})$$

Plus généralement,

$$P(a | b, c) = \frac{P(b | a, c) \cdot P(a | c)}{P(b | c)} \quad (\text{B.5})$$

Annexe C

Description des opérations du cas d'application

Cette annexe complète la description des opérations de la section 5 du chapitre 6.

Fabrications des marques O_1		
Équipement M_1		Recette R_1
L'instant de début: 06:25:07	Produit P_1	Etat d'opération : Non suspect
L'instant de fin : 07:11:30		
Évolution chaîne fonctionnelle		
Effet Transitoire: Fabrique en cours	Pré-condition: Fonction $M_1 ==$ disponible	Pré-contrainte: Intérieur $M_1 ==$ vide
Effet final: Transformation terminée $IC(\text{Transformation})=1$	Condition:	Contrainte: Etat $M_1 ==$ prêt
Évolution de flux de produits		
Effet Transitoire: $P_1 ==$ Fabrication en cours	Pré-condition: $P_1 ==$ prêt	Pré-contrainte:
Effet final: $P_1 ==$ fabriqué $IC(P_1=\text{fabriqué})= 1$	Condition:	Contrainte: Sortie $== P_1$ fabriqué

FIGURE C.1 – Opération d'action fabrication O_1

Fabrications des marques O_2		
Équipement M_2		Recette R_2
L'instant de début: 06:00:07	Produit P_2	Etat d'opération : Non suspect
L'instant de fin : 08:07:30		
Évolution chaîne fonctionnelle		
Effet Transitoire: Fabrique en cours	Pré-condition: Fonction $M_2 ==$ disponible	Pré-contrainte: Intérieur $M_2 ==$ vide
Effet final: Transformation terminée $IC(\text{Transformation})=1$	Condition:	Contrainte: Etat $M_2 ==$ prêt
Évolution de flux de produits		
Effet Transitoire: $P_2 =$ Fabrication en cours	Pré-condition: $P_2 ==$ prêt	Pré-contrainte:
Effet final: $P_2 ==$ fabriqué $IC(P_2 = \text{fabriqué})= 1$	Condition:	Contrainte: Sortie == P_2 fabriqué

FIGURE C.2 – Opération d'action fabrication O_2

Wafer-cleaning O_4		
Équipement M_4		
L'instant de début: 09:27:11	Produit P_4	Etat d'opération : Non suspect
L'instant de fin : 11:18:51		
Évolution chaîne fonctionnelle		
Effet Transitoire: Nettoyage en cours	Pré-condition: Fonction $M_4 ==$ disponible	Pré-contrainte: Intérieur $M_4 ==$ vide
Effet final: Nettoyage terminé Équipement prêt $IC(\text{Nettoyage terminé})=1$	Condition:	Contrainte: Etat $M_4 ==$ prêt
Évolution de flux de produits		
Effet Transitoire: $P_4 =$ nettoyage en cours	Pré-condition: $P_4 ==$ non nettoyé	Pré-contrainte: $P_4 ==$ prêt
Effet final: $P_4 ==$ nettoyé $IC(P_4 == \text{non nettoyé}) = 1$	Condition:	Contrainte: Sortie == P_4 nettoyé

FIGURE C.3 – Opération d'action Wafer-cleaning O_4

Wafer-cleaning O_5		
Équipement M_5		
L'instant de début: 08:42:54	Produit P_5	Etat d'opération : Non suspect
L'instant de fin : 10:19:42		
Évolution chaîne fonctionnelle		
Effet Transitoire: Nettoyage en cours	Pré-condition: Fonction M_5 == disponible	Pré-contrainte: Intérieur M_5 == vide
Effet final: Nettoyage terminé Équipement prêt IC(Nettoyage terminé)=1	Condition:	Contrainte: Etat M_5 == prêt
Évolution de flux de produits		
Effet Transitoire: P_5 = nettoyage en cours	Pré-condition: P_5 ==non nettoyé	Pré-contrainte: P_5 == prêt
Effet final: P_5 == nettoyé IC(P_5 ==non nettoyé) =1	Condition:	Contrainte: Sortie = P_5 nettoyé

FIGURE C.4 – Opération d'action Wafer-cleaning O_5

Deposition O_6		
Équipement M_6		Recette R_6
L'instant de début: 11:00:07	Produit P_6	Etat d'opération : Non suspect
L'instant de fin : 15:27:30		
Évolution chaîne fonctionnelle		
Effet Transitoire: Déposition en cours	Pré-condition: Fonction M_6 == disponible	Pré-contrainte: Intérieur M_6 == vide
Effet final: Déposition terminée IC(Déposition)=1	Condition:	Contrainte: Etat M_6 == prêt
Évolution de flux de produits		
Effet Transitoire: P_6 = Déposition en cours	Pré-condition: P_6 ==prêt	Pré-contrainte:
Effet final: P_6 == Déposé IC(P_6 =Déposé)= 1	Condition:	Contrainte: Sortie == P_6 déposé

FIGURE C.5 – Opération d'action Deposition O_6

Déposition O_7		
Équipement M_7	Humain H_7	Recette R_7
L'instant de début: 12:02:00	Produit P_7	Etat d'opération : Non suspect
L'instant de fin : 15:17:30		
Évolution chaîne fonctionnelle		
Effet Transitoire: Déposition en cours	Pré-condition: Fonction M_7 == disponible	Pré-contrainte: Intérieur M_7 == vide
Effet final: Déposition terminée $IC(Déposition)=1$	Condition:	Contrainte: Etat M_7 == prêt
Évolution de flux de produits		
Effet Transitoire: P_7 ==Déposition en cours	Pré-condition: P_7 ==prêt	Pré-contrainte:
Effet final: P_7 == Déposé $IC(P_7=Déposé)= 1$	Condition:	Contrainte: Sortie == P_7 déposé

FIGURE C.6 – Opération d'action Déposition O_7

Lithography O_8		
Équipement M_8		
L'instant de début: 15:45:08	Produit P_8	Etat d'opération Suspect
L'instant de fin : 17:05:11		
Évolution chaîne fonctionnelle		
Effet Transitoire: Lithographie en cours	Pré-condition: Fonction M_4 == disponible	Pré-contrainte: Intérieur M_8 == vide
Effet final: Lithographie terminée Equipement prêt $IC(Lithographie)=1$	Condition:	Contrainte: Etat M_8 == prêt
Évolution de flux de produits		
Effet Transitoire: P_8 ==Lithographie en cours	Pré-condition: P_8 ==non lithographié	Pré-contrainte:
Effet final: P_8 == Lithographié $IC(P_8 ==Lithographié) =0$	Condition:	Contrainte: Sortie == P_8 lithographié

FIGURE C.7 – Opération d'action Lithography O_8

Etching O ₉		
Équipement M ₉	Humain H ₉	
L'instant de début: 16:02:00 L'instant de fin : 18:37:14	Produit P ₉	Etat d'opération : Non suspect
Évolution chaîne fonctionnelle		
Effet Transitoire: Gravure	Pré-condition: Fonction M ₉ == disponible	Pré-contrainte: Intérieur M ₉ == vide
Effet final: Gravure terminée IC(Gravure)=1	Condition:	Contrainte: Etat M ₉ == prêt
Évolution de flux de produits		
Effet Transitoire: P ₉ ==Gravure en cours	Pré-condition: P ₉ == non gravuré	Pré-contrainte:
Effet final: P ₉ == Déposé IC(P ₉ ==Déposé)= 1	Condition:	Contrainte: Sortie == P ₉ gravuré

FIGURE C.8 – Opération d'action Etching O₉

Etching O ₁₀		
Équipement M ₁₀	Humain H ₁₀	
L'instant de début: 17:32:00 L'instant de fin : 19:28:00	Produit P ₁₀	Etat d'opération :Suspect
Évolution chaîne fonctionnelle		
Effet Transitoire: Gravure	Pré-condition: Fonction M ₁₀ == disponible	Pré-contrainte: Intérieur M ₁₀ == vide
Effet final: Gravure terminée IC(Gravure)=0	Condition:	Contrainte: Etat M ₁₀ == prêt
Évolution de flux de produits		
Effet Transitoire: P ₁₀ == Gravure en cours	Pré-condition: P ₁₀ ==prêt	Pré-contrainte:
Effet final: P ₁₀ == Gravuré IC(P ₁₀ ==Gravure)= 0	Condition:	Contrainte: Sortie == P ₁₀ gravuré

FIGURE C.9 – Opération d'action Etching O₁₀

Transférer de la salle Fabrication à la salle Wafer-cleaning		
Durée Fabrication_Wafer-cleaning		
1 ^{ère} Évolution de flux de produits		
Effet Transitoire: Fabrications des marques ↓ Fabrications des marques_Wafer-cleaning	Pré-condition: Wafer-cleaning == non vide	Pré-contrainte: Fabrications des marques_Wafer-cleaning == vide
Effet final: Fabrications des marques_Wafer-cleaning ↓ Wafer-cleaning	Condition: Convoyeur == Disponible	Contrainte: Wafer-cleaning == vide
2 ^{ème} Évolution de flux de produits		
Effet Transitoire:	Pré-condition: Fabrications des marques_Wafer-cleaning == non vide	Pré-contrainte: Wafer-cleaning == vide
Effet final: Fabrications des marques_Wafer-cleaning ↓ Wafer-cleaning	Condition: Convoyeur == Disponible	Contrainte: Wafer-cleaning == non vide

FIGURE C.10 – Opération induite de transfert de l'îlot de Fabrication à l'îlot Wafer-cleaning

Transférer de la salle Wafer-cleaning à la salle Deposition		
Durée Wafer-cleaning_Deposition		
1 ^{ère} Évolution de flux de produits		
Effet Transitoire: Wafer-cleanings ↓ Wafer-cleanings_Deposition	Pré-condition: Wafer-cleaning == non vide	Pré-contrainte: Wafer-cleanings_Deposition == vide
Effet final: Wafer-cleanings_Deposition ↓ Deposition	Condition: Convoyeur == Disponible	Contrainte: ↓ Deposition == vide
2 ^{ème} Évolution de flux de produits		
Effet Transitoire:	Pré-condition: Wafer-cleanings == non vide	Pré-contrainte: Wafer-cleanings_Deposition == vide
Effet final: Fabrications des marques_Wafer-cleaning ↓ Wafer-cleaning	Condition: Convoyeur == Disponible	Contrainte: Wafer-cleaning == non vide

FIGURE C.11 – Opération induite de transfert de l'îlot Wafer-cleaning à l'îlot Deposition

Transférer de la salle Deposition à la salle Lithography		
Durée Deposition-Lithography		
1 ^{ère} Évolution de flux de produits		
Effet Transitoire: Deposition ↓ Deposition-Lithography	Pré-condition: Deposition == non vide	Pré-contrainte: Deposition-Lithography == vide
Effet final: Deposition-Lithography ↓ Lithography	Condition: Convoyeur == Disponible	Contrainte: Lithography == vide
2 ^{ème} Évolution de flux de produits		
Effet Transitoire:	Pré-condition: Deposition-Lithography == non vide	Pré-contrainte: Lithography == vide
Effet final: Deposition-Lithography ↓ Lithography	Condition: Convoyeur == Disponible	Contrainte: Lithography == vide

FIGURE C.12 – Opération induite de transfert de l'îlot de Deposition à l'îlot Lithography

Transférer de la salle Lithography à la salle Etching		
Durée Lithography- Etching		
1 ^{ère} Évolution de flux de produits		
Effet Transitoire: Lithography ↓ Lithography-Etching	Pré-condition: Lithography == non vide	Pré-contrainte: Lithography-Etching == vide
Effet final: Lithography-Etching ↓ Etching	Condition: Convoyeur == Disponible	Contrainte: Lithography == vide
2 ^{ème} Évolution de flux de produits		
Effet Transitoire:	Pré-condition: Deposition-Lithography == non vide	Pré-contrainte: Lithography == vide
Effet final: Deposition-Lithography ↓ Lithography	Condition: Convoyeur == Disponible	Contrainte: Lithography == vide

FIGURE C.13 – Opération induite de transfert de l'îlot de Lithography à l'îlot de Etching

Installer le produit à l'entrée du processus		
Durée		
1^{ère} Évolution de flux de produits		
Effet Transitoire:	Pré-condition: Entrée == vide	Pré-contrainte: Intérieur M_1 == vide
Effet final: Entrée == non vide	Condition:	Contrainte: Entrée == vide Etat M_1 == Prêt

FIGURE C.14 – Opération requise à l'entrée du processus

Sortir le produit du processus		
Durée		
1^{ère} Évolution de flux de produits		
Effet Transitoire:	Pré-condition: Entrée == vide	Pré-contrainte: Intérieur M_9 == vide
Effet final: Entrée == non vide	Condition:	Contrainte: Entrée == vide Etat M_9 == Prêt

FIGURE C.15 – Opération requise à l'entrée du processus

Opération de surveillance sur la machine M_i		
Équipement : Capteur CM_i		
Durée Du_s		Disponible : non suspecte
Évolution chaîne fonctionnelle		
Effet Transitoire: CM_i == occupé	Pré-condition:	Pré-contrainte: CM_i == libre
Effet final: CM_i == libre	Condition:	Contrainte:
Les variables surveillées		
Etat de M_i = non suspecte	Etat de produit P_i == Non suspecte	

FIGURE C.16 – Opération de surveillance sur la machine M_i pour $i = 1..10$