Diagnostic de Services pour la Reconfiguration Dynamique de Systèmes à Evénements Discrets Complexes

Eric Deschamps

Institut Polytechnique de Grenoble

No. attribué par la bibliothèque

THESE

pour obtenir le grade de

DOCTEUR DE L'INPG

Spécialité : Automatique-Productique

préparée au Laboratoire d'Automatique de Grenoble et au
Laboratoire des Sciences pour la Conception, l'Optimisation et la Production

dans le cadre de l'École Doctorale :
Électronique, Électrotechnique, Automatique, Traitement du Signal

présentée et soutenue publiquement

par

Éric Deschamps

le 05 novembre 2007

Titre :

Diagnostic de Services pour la Reconfiguration Dynamique de Systèmes à Événements Discrets Complexes

Directeur de thèse : Éric ZAMAÏ

JURY

Mme Mireille JACOMINO,	Professeur à l'Institut Polytechnique de Grenoble	Présidente
M. Michel COMBACAU,	Professeur à l'Université Paul Sabatier	Rapporteur
M. Janan ZAYTOON,	Professeur à l'Université de Reims Champagne-Ardenne	Rapporteur
M. Pascal BERRUET,	Maître de Conférences à l'Université de Bretagne Sud	Examinateur
M. Sébastien HENRY,	Maître de Conférences à l'Université Claude Bernard - Lyon 1	Examinateur
M. Jean-Jacques LESAGE,	Professeur à l'École Normale Supérieure de Cachan	Examinateur
M. Éric ZAMAÏ,	Maître de Conférences HDR à l'Institut Polytechnique de Grenoble	Directeur de thèse

Avant-Propos

Je souhaiterais à travers ces quelques lignes, rendre hommage à toutes les personnes qui m'ont aidé et soutenu pendant mes trois années de doctorat. Une thèse n'est pas uniquement un travail de recherche, c'est une expérience, une partie de sa vie où l'on découvre un nouveau paysage. J'ai ainsi commencé ma thèse au Laboratoire d'Automatique de Grenoble (LAG), et j'aimerais dans ce sens remercier Monsieur Alain Barraud, qui a dirigé ce laboratoire pendant mes deux premières années de thèse. J'ai ensuite eu la chance de pouvoir vivre une restructuration des laboratoires de recherche, et ainsi d'être accueilli par Monsieur Yannick Frein, que je remercie, en tant que directeur du nouveau laboratoire des Sciences pour la conception, l'optimisation et la production de Grenoble (G-SCOP).

Ma thèse a été une série d'étapes, plus ou moins longues, plus ou moins remplies de doutes, de remises en cause, de questionnements. J'ai eu l'immense chance, d'avoir à chacun de ces moments, une personne qui m'a soutenue, qui a toujours su trouver les mots pour m'aider à surpasser ces difficultés. Pour cela j'adresse ma plus sincère gratitude à mon directeur de thèse, Monsieur Eric Zamaï. Au delà de son encadrement sans faille, Eric a été un de mes plus chers collègues, mais également un ami... Je ne saurais trouver les mots pour lui exprimer ma reconnaissance de m'avoir aidé à trouver ma voie... pour la confiance qu'il m'a toujours accordée... pour les bons moments que nous avons passés tout au long de ces trois années... merci Eric.

Je voudrais exprimer toute ma gratitude aux membres de mon Jury. Tout d'abord j'adresse mes remerciements à Messieurs Michel Combacau et Janan Zaytoon pour m'avoir fait le privilège d'être rapporteurs de ma thèse, pour le temps qu'ils y ont consacré, et surtout pour leurs critiques qui sont pour moi autant de force à poursuivre dans cette voie. Je voudrais remercier Messieurs Jean-Jeacques Lesage, Pascal Berruet et Sébastien Henry pour m'avoir fait le plaisir d'être examinateurs de mes travaux de recherche et pour leurs remarques qui témoignent de l'intérêt qu'ils ont porté à mes travaux. Je voudrais enfin remercier Madame Mireille Jacomino, qui m'a fait la joie de présider mon jury de thèse. Mes remerciements ne peuvent s'arrêter là, car Mireille m'a tout d'abord accueilli à bras ouverts dans son équipe de recherche et elle a toujours su poser son regard critique sur mes travaux. Pour tout cela, et pour tous les moments que nous avons partagés, je souhaite sincèrement et très amicalement lui transmettre toute ma gratitude.

Tout au long de cette thèse, de nombreux chercheurs et enseignants-chercheurs m'ont fait l'honneur d'examiner mes travaux à travers les journées de rencontre du laboratoire, du pôle STP du GDR MACS, ou encore à travers les conférences. Je voudrais remercier toutes ces per-

sonnes, qui ont largement contribué à l'avancement et à la construction de ce travail de recherche.

J'ai rencontré de nombreuses personnes dans mes deux laboratoires de recherche, avec qui j'ai partagé de nombreuses discussions, des pauses cafés et surtout beaucoup de moments de joie. Je n'en doute pas, elles se reconnaîtront et je leur adresse mes sincères remerciements. Je voudrais en particulier exprimer ma gratitude aux personnes qui m'ont accompagnées dans toutes mes démarches administratives aussi bien pour mon intégration à l'établissement, que pour les missions que j'ai eu la chance de faire ou encore pour l'organisation de ma soutenance de thèse. Pour tout cela, merci à Marie-Rose ,Marielle, Marie-thérèse, Patricia, Virginie, Amandine, Chaneise, Chantal, Kheira, Myriam, Souad... Enfin je voudrais remercier Stéphane, Jean-Marie et Olivier, collègues enseignants-chercheurs, pour leur écoute, leurs conseils et la disponibilité dont ils ont toujours fait preuve.

Je ne pourrais parler de ces laboratoires sans parler de mes collègues de bureau qui sont devenu des amis. Tout d'abord Seb, qui m'a dans un premier temps encadré en Master, puis a partagé mon bureau les deux premières années de ma thèse. Je le remercie pour son aide, son amitié, ses précieux conseils et pour les nombreuses discussions qui m'ont aidé à aiguiser mon sens critique et ont largement contribué aux travaux que je présente dans ce manuscrit. J'ai eu également la joie de partager mon bureau avec Alexis et Long, qui en toutes circonstances ont su se montrer présent et me soutenir. Il me serait difficile de leur dire dans ces quelques lignes à quel point leur aide m'a été précieuse, mais en tout cas je les remercie pour leur patience et leur présence à chaque fois que j'en ai eu besoin. Je souhaiterais également remercier Nicolas et Freddy, avec qui j'ai pris beaucoup de plaisir à travailler, et qui ont su apporter chacun à leur façon une pierre aux travaux présentés dans ce manuscrit. Au delà des personnes du laboratoire, je souhaiterais remercier une personne en particulier, Dominique, pour la joie de vivre qu'elle sait si bien transmettre et pour toutes les fois où elle m'a si gentiment rendu service.

Parallèlement à ces trois années de recherche, j'ai eu la chance d'avoir un poste de moniteur à l'ENSGI. Je tiens pour cela à remercier Messieurs Gérard Cognet et Didier Retour, les deux directeurs successifs qui ont oeuvré pour me proposer de nombreuses formations en adéquation avec le parcours professionnel que j'ai choisi. Je tiens également à remercier dans ce cadre mes collègues Fabien, Yannick, Gulgun, Bernard, Jean-Philippe, Pierre qui m'ont parfaitement intégré à leurs équipes pédagogiques et avec qui j'ai pris beaucoup de plaisir à participer à l'organisation et l'évolution des enseignements. Je remercie également mes collègues moniteurs et vacataires, Hanane, Fethi, Zohra, Murat avec qui nous avons beaucoup échangé sur les problèmes pédagogiques.

Je voudrais remercier, tous les enseignants que j'ai connus tout au long de mon parcours scolaire. Je crois que c'est aussi grâce à ces personnes, qui m'ont appris tant de choses, que j'ai pu en arriver là. Je souhaite remercier en particulier des enseignants qui m'ont donné goût à l'enseignement et dont les méthodes pédagogiques m'ont beaucoup inspirées : Messieurs Goublet, Étienne, Madame Chanussot et Eric Escande.

Il serait injuste d'oublier tous mes amis qui m'ont supporté pendant ces trois années. Ils ont su contribuer à ce que ces trois années de thèse restent pour moi inoubliables. Merci à Alex,

Audrey, Aurélie, Dimitri, Florent, Fred, Gaëtan, Guillaume, John, Ju, Nico, Pierre, Rémi, Sam, Steph, Sylvie, Seb, Thom, Xavier... mais aussi à Rémi, Régine, Evelyne et Alain pour l'aide et le soutien que vous m'avez apporté. Parmi ces amis, je voudrais adresser un remerciement particulier à une personne qui m'a apporté tant de choses d'un point de vue personnel durant ces années de thèse... merci Ju.

Avant de rentrer dans la présentation de mes travaux, je souhaiterais les dédier aux trois personnes qui me sont les plus chères. Sans le soutien de mes parents et de ma sœur il est certain que je ne serais pas en train d'écrire ces lignes. Vous avez cru en moi sans le moindre doute depuis mon plus jeune âge. Vous vous êtes battus avec moi pour surmonter les difficultés, vous m'avez soutenu toutes ces années et vous continuez encore aujourd'hui. Il n'y a pas de mot pour vous exprimer l'intensité de ma reconnaissance, mais sachez que ces efforts n'ont pas été vains.

Table des matières

Introduction générale

Dans l'industrie manufacturière d'aujourd'hui, les marchés mondialisés sont en perpétuelle évolution. Le contexte de développement des produits, souvent perturbé par des variations internes ou externes, a un caractère particulièrement fluctuant.

Pour survivre et évoluer dans ce contexte fortement instable, les maîtres mots des industriels sont plus que jamais : réduction des coûts et des délais de production, amélioration de la qualité de fabrication pour accroître encore la productivité de l'entreprise.

Pour atteindre ces objectifs, des améliorations doivent être apportées pour faire face aux situations et problèmes inconnus qui peuvent se produire au cours d'un cycle de production. Lorsque nous nous focalisons sur les aléas issus de la partie opérative elle-même, nous entrons alors au sein de la problématique de la supervision et de la surveillance temps réel des systèmes à événements discrets complexes.

L'étude d'une telle problématique vise à spécifier et concevoir des Systèmes Automatisés de Production en vue d'une meilleure autonomie en présence de dysfonctionnements et ainsi aider les industriels à mieux maîtriser leur production.

Dans ce cadre, de nombreuses approches et solutions ont été proposées, accompagnées du développement de procédures fines de supervision, de surveillance et de commande. Parmi ces dernières il est coutume de retrouver des mécanismes de détection des symptômes de défaillances, de diagnostic permettant de retrouver les origines de ces défaillances, de décision visant non seulement à définir de nouveaux objectifs de production mais aussi de synthétiser de nouvelles lois de commande. L'ensemble de ces fonctionnalités s'appuie généralement sur un modèle de la partie opérative dont l'état doit être en permanence mis à jour non seulement par l'ensemble des évolutions provoquées par l'exécution des lois de commande, mais également vis à vis de l'ensemble des informations, plus ou moins agrégées, issues du système de captage.

Le travail que nous présentons dans ce mémoire propose d'apporter sa contribution au domaine de la surveillance et supervision, en ligne, des systèmes à événements discrets complexes. Il se place volontairement dans un contexte perturbé par l'occurrence d'aléas de fonctionnement d'une partie opérative dans laquelle deux objectifs majeurs doivent être visés par le système de pilotage réactif considéré : d'une part la réparation d'un composant défectueux et d'autre part le maintient si possible de la productivité. Pour satisfaire ces objectifs, au moins une fonction

diagnostic doit être proposée, d'une part pour localiser le composant défectueux et d'autre part, dans un temps imparti fixé essentiellement par des critères de productivité, déterminer quelles sont les capacités opératoires encore disponibles. C'est au deuxième axe que cette thèse propose d'apporter une solution.

Ce mémoire est organisé en quatre parties dont les thèmes sont donnés ci-après :

La première partie présente de manière générale la problématique de la commande, de la surveillance et de la supervision dans les systèmes automatisés de production. Ainsi après avoir exposé le contexte général des systèmes réactifs aux aléas de fonctionnement volontairement positionné aux niveaux coordination des chaînes fonctionnelles, nous proposons d'exposer les besoins d'une fonction du processus réactif, le diagnostic. Sur cette base, nous dévoilerons le cahier des charges des travaux développés dans ce manuscrit, ainsi qu'une étude critique des principales approches ayant apportées une contribution au domaine du diagnostic.

La partie II est consacrée à une présentation des éléments requis pour envisager de proposer les mécanismes de suivi et de diagnostic de services. Afin d'aborder le problème de l'acquisition des connaissances requises aux mécanismes proposés, nous présentons tout d'abord une démarche qui s'appuie sur des techniques de modélisation issue de la génération de plans en Intelligence Artificielle. Ces techniques seront étendues dans le cadre de ce mémoire aux besoins de notre problématique. Après quoi, afin de préparer la structuration des mécanismes de suivi et de diagnostic qui seront proposés, une formalisation de ces connaissances basée sur la logique propositionnelle sera dévoilée.

Dans partie III, l'ensemble des mécanismes de suivi, de diagnostic et de mise à jour sont exposés. Dans un premier temps un mécanisme de suivi est proposé, permettant in fine d'obtenir un modèle du fonctionnement passé de la partie opérative basé sur la structuration des modèles utilisés en diagnostic logique. Ce mécanisme sera conjointement utilisé avec des procédures de gestion de la taille de ce modèle afin de répondre à notre problématique. Sur cette base, nous proposons une méthode de diagnostic permettant de mettre en exergue suite à l'occurrence d'un dysfonctionnement avéré de la partie opérative, quelles sont les origines qui doivent être suspectées, ainsi que leurs impacts sur le fonctionnement de la partie opérative. Disposant ainsi d'une vision "honnête" du fonctionnement passé, un dernier mécanisme de mise à jour est proposé afin de projeter ces résultats au sein du modèle de partie opérative.

La partie IV développe un exemple d'application des mécanismes proposés sur la base d'une cellule de vernissage de cartes électroniques. Après avoir présenté d'une façon générale la partie opérative ainsi que son architecture de pilotage, nous proposons une étude, orientée sur l'analyse des performances de la fonction suivi. Nous terminons cette partie par une présentation de trois scenarii de dysfonctionnement permettant de tester et de valider les mécanismes de diagnostic et de mise à jour proposés.

Première partie

Cadre de l'étude

Chapitre 1

Contexte général

1 Introduction

Ce premier chapitre est consacré à la présentation générale du contexte de nos travaux et à la définition des concepts de base de notre étude. Volontairement localisés au niveau pilotage temps réel des Systèmes Automatisés de Production (SAP), ce chapitre a la volonté de nous sensibiliser à la problématique générale des systèmes réactifs aux aléas de fonctionnement.

Aussi, dans le premier paragraphe de ce chapitre nous présentons la structure organisationnelle et structurelle des SAP et positionnons nos travaux aux niveaux coordination des chaînes fonctionnelles. Fort de ce positionnement, nous explorons ensuite le concept d'aléas de fonctionnement ce qui nous conduira naturellement à présenter les principes de base des processus réactifs. Ceci nous amènera alors à présenter rapidement les fonctions telles que la détection, le diagnostic ou encore la décision. Après quoi nous terminerons ce chapitre sur un approfondissement du concept de chaînes fonctionnelles sur lesquelles la suite du document s'appuie.

2 Les Systèmes Automatisés de Production

2.1 Description générale

A l'origine, les SAP sont apparus afin de soulager l'homme dans ses tâches pénibles et dangereuses. Ensuite cette automatisation a permis de contribuer à l'amélioration de la productivité des entreprises. Comme tout système de production de biens, il a pour objectif de transformer des matières premières en produits finis (cf. Figure 1.1). L'automatisation permet de diminuer le temps de production de ces produits, et ainsi de diminuer les délais de livraison aux clients, tout en essayant de garantir leur qualité par rapport aux spécifications données par ces derniers (AFNOR, 1991). Les demandes des clients étant de plus en plus variées (Agard et Tollenaere, 2002) et les contraintes liées à la sécurité des biens et des personnes de plus en plus sévères (Mendez et al., 2003), il apparaît une complexité croissante de la partie opérative des SAP.

Celle-ci se traduit en termes de nombre de composants de la partie opérative du SAP et d'agencement de ces composants. Afin de répondre aux demandes clients de plus en plus exigeantes il est nécessaire que le SAP propose une certaine flexibilité physique (Berruet, 1998) quant à son exploitation. Ceci permet d'une part de pouvoir fabriquer un produit de

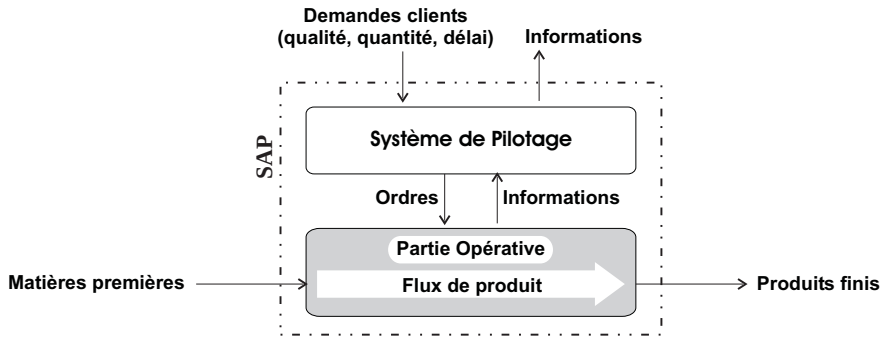


FIG. 1.1 – vue schématique d'un SAP

différentes façons afin de respecter les délais de livraison des produits même en présence d'aléas de fonctionnement de la partie opérative et d'autre part de pouvoir rapidement s'adapter à une nouvelle demande.

Toutefois la flexibilité physique de la partie opérative n'est pas en soi suffisante. Elle doit également s'accompagner d'un système de pilotage capable de décider comment les tâches à réaliser seront affectées par rapport aux flexibilités laissées par la partie opérative et de décider comment le produit sera réalisé, c'est à dire quelle recette sera utilisée. Cette capacité à s'adapter aux spécificités des produits et aux aléas de fonctionnement est plus couramment appelée la flexibilité décisionnelle (Zamai et al., 1998). Avant d'aller plus loin dans ces aspects décisionnels, les prochaines sections proposent de décrire la structure générale des SAP.

2.2 Structure d'un SAP

Comme représenté schématiquement par la Figure 1.1, un SAP peut être vu comme la composition de trois parties (Noureddine, 2005; Perrin et al., 2004) :

- **Le flux de produits** qui regroupe l'ensemble des produits se trouvant à un instant donné dans le SAP (matière première, transformée, assemblée, produits finis).

- **La partie opérative** qui peut se décomposer en chaînes d'action et en chaînes d'acquisition (cf. Figure 1.2). Une chaîne d'action est un ensemble d'éléments organisés dans le but de transformer des ordres de commande en actions physiques (déplacement, mise en rotation, ...). Ces actions ont pour objectif non seulement de modifier l'état du flux de produits, mais également l'état même des chaînes d'action afin de satisfaire des contraintes de sécurité entre autres. Les chaînes d'acquisition permettent quant à elles de mesurer des grandeurs physiques (température, position d'un vérin, vitesse de rotation, couleur du produit...) afin de remonter de l'information vers le système de pilotage.

- **Le système de pilotage** qui est l'ensemble des éléments exploitant les capacités des chaînes d'action et les informations fournies par les chaînes d'acquisition afin de tendre en permanence vers les objectifs qu'il poursuit, objectifs décrits en terme de spécification des produits à fournir, des quantités, des délais de fabrication...

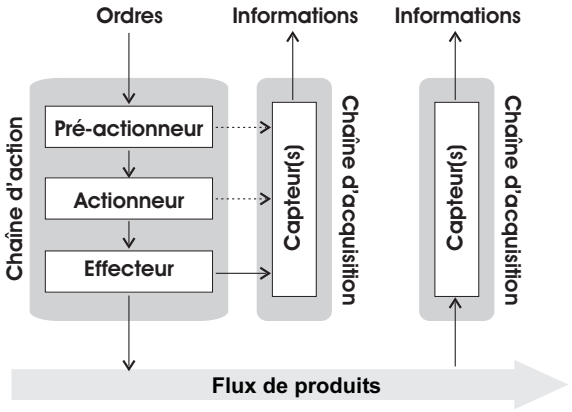


FIG. 1.2 – Chaînes d'action et chaînes d'acquisition

2.3 Système de pilotage

Afin d'appréhender la complexité des SAP, il est classiquement proposé de décomposer le système de pilotage en une structure hiérarchique et modulaire, le principe étant de décomposer un problème complexe en un sous-ensemble de problèmes de complexité moindre comme proposé dans (Jones et Saleh, 1989; O'Grady et al., 1994). Ceci amène d'une part à décomposer le système de pilotage verticalement en utilisant par exemple le modèle présenté dans (CIM, 1989) composé de 5 niveaux (cf. Figure 1.3) et d'autre part à décomposer horizontalement chacun de ces niveaux en un ensemble de modules de pilotage. Cette décomposition permet d'aboutir à une structure de pilotage hiérarchique et modulaire telle que représentée dans la Figure 1.4.

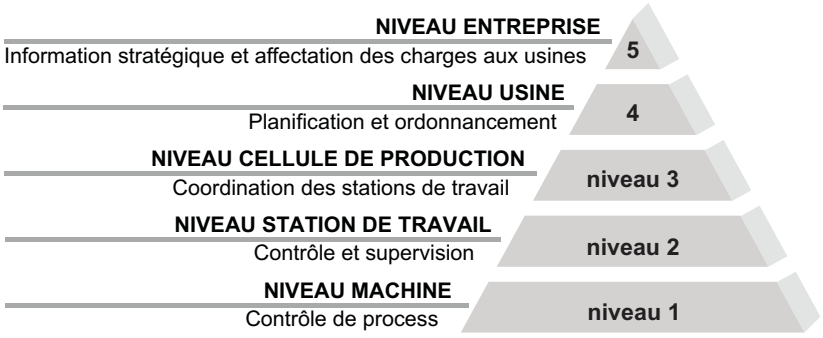


FIG. 1.3 – Architecture CIM

Une telle structure véhicule un grand nombre d'informations. En fonctionnement normal (absence d'aléas de fonctionnement de la partie opérative) un protocole de communication de type appel/réponse est généralement utilisé (Combacau, 1991). Il est basé sur le fonctionnement suivant : un module de niveau i désagrège une requête reçue du niveau i+1 en une séquence de requêtes vers les modules de niveau i-1. Chaque requête s'accompagne d'un délai alloué à la réalisation de la tâche demandée. Suite à l'envoi d'une requête, le module de niveau i est en

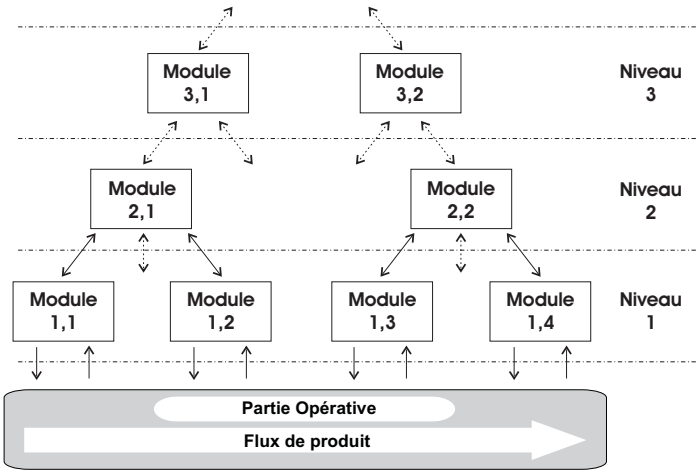


FIG. 1.4 – Structure de pilotage considéré

attente d'un compte-rendu du module de niveau i-1. Ce compte-rendu reçu traduit généralement l'achèvement correct de la requête envoyée. Cependant dans le contexte de dysfonctionnement de la partie opérative, celui-ci peut également traduire l'impossibilité du niveau inférieur d'exécuter la requête qui lui a été envoyée. Ce compte-rendu pourra être éventuellement accompagné d'informations complémentaires telles que les raisons de l'impossibilité d'exécuter la requête envoyée. La réception d'un tel compte-rendu traduira forcément un fonctionnement anormal de la partie opérative pilotée par le module en question.

Un module ne peut être capable de générer des comptes-rendus ou des informations traduisant un fonctionnement anormal que si celui-ci intègre des moyens de surveillance. Il est alors classique d'implanter dans l'ensemble de la structure hiérarchique et modulaire un ensemble de fonctionnalités de Surveillance, Supervision et Commande (Combacau et al., 2000). Ceci afin que chacun des modules soit en mesure de réagir à l'occurrence d'une défaillance et de prendre l'initiative du processus réactif afin de respecter le principe de confinement (Combacau, 1991).

Afin d'appréhender au mieux cette problématique, les sections suivantes présentent les aléas de fonctionnement et les fonctions permettant de les traiter. D'autre part une analyse des différents niveaux de pilotage temps réel (niveau 1 et 2 du CIM) sera présentée afin de mettre en exergue leurs spécificités, et ainsi les besoins en terme de surveillance et de supervision de ces niveaux.

3 Aléas de fonctionnement

3.1 Les origines des aléas

Les aléas de fonctionnement liés à l'exploitation des SAP sont aussi nombreux que variés (cf. Figure 1.5). Ceux-ci se caractérisent sous formes d'événements non prévus qui viennent perturber le fonctionnement du SAP et éventuellement remettre en cause l'objectif même de la production (Combacau et al., 2002). Ils se classent généralement en deux types : les aléas

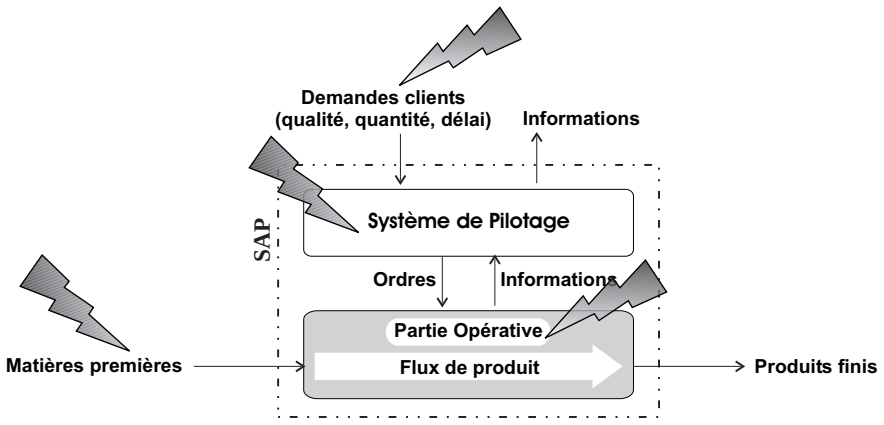


FIG. 1.5 – Origines des aléas de fonctionnement

internes et les aléas externes au SAP :

Les aléas internes :
- aléas de la partie matérielle du système de pilotage (panne de calculateur, du réseau de communication...),
- aléas de la partie logicielle du système de pilotage (erreur de spécification, erreur de codage...),
- aléas de la partie opérative (panne d'un capteur, d'un actionneur...).

Les aléas externes :
- modification de la demande client (changement de spécification du produit, diminution du délai de livraison,...),
- aléas sur les caractéristiques de la matière première (taille non conforme, dureté d'une matière non conforme,...),
- aléas sur l'environnement du SAP (panne d'alimentation générale, ...).

De nombreux travaux ont porté sur les aléas liés à la partie logicielle dans le domaine de la sûreté de fonctionnement des logiciels (Vallée, 2003) et de nombreuses normes ont été mises en place afin d'uniformiser les mises en œuvres des vérifications de spécification et de codage (Kahn, 2004). Nous considérerons tout au long de ce manuscrit le système de pilotage comme exempt d'aléas de fonctionnement afin de nous focaliser sur les aléas de la partie opérative mais également sur les aléas externes pouvant l'affecter.

Un changement de la demande client entraîne une remise en cause quasi systématique des plans de production. Cependant bien que ces changements puissent être imprévus, ces informations restent des spécifications connues. Reste donc en charge du système de pilotage de définir de nouveaux objectifs de productions. Par opposition à ce type d'aléas, ceux de la partie opérative, de l'environnement ainsi que les non-conformités de la matière première sont quant à eux beaucoup plus délicats à traiter. Bien que leurs effets soient en partie observés par le système de pilotage, il est bien souvent difficile de les identifier et de les compenser. C'est dans ce contexte qu'évoluent les travaux présentés dans ce mémoire.

3.2 Défaillances de la partie opérative

Une défaillance d'un composant peut être définie comme son incapacité à réaliser une fonction (Zwingelstein, 1999). Cette incapacité provient généralement d'un changement progressif des caractéristiques du composant telles que le vieillissement, ou le changement brutal tel qu'une casse mécanique. Une défaillance première est caractérisée par une défaillance dont l'unique origine est la panne d'un composant. Une défaillance peut également être la conséquence d'une défaillance antérieure non observée. Cette dernière a pu se propager à travers la partie opérative par le non respect de contraintes de sécurité (Chang et al., 1991). Ce cas inclut la non conformité de la matière première qui pourrait entraîner une défaillance dans le SAP.

Sans traitement, ces défaillances entraîneraient rapidement un blocage du système de pilotage, conduisant alors à une dégradation des performances globales du SAP.

4 Réactivité aux défaillances

4.1 Confinement d'une défaillance

Suite à la réception par un module considéré d'un compte-rendu traduisant un dysfonctionnement de la partie opérative pilotée (cf. 2.3) celui-ci doit être en mesure de réagir afin de lancer un processus de confinement et ainsi éviter de le propager dans les niveaux supérieurs (Combacau, 1991). Afin de pouvoir respecter ce principe fondamental, nous retrouvons au sein de chacun des modules de la structure de pilotage les fonctions suivantes : commande, suivi, détection, diagnostic, pronostic et décision. De nombreux travaux ont porté sur ces fonctionnalités et nous proposons ici des définitions (Combacau et al., 2000, 2002) qui seront utilisées dans la suite de ce manuscrit :

- **La commande** (Henry, 2005) a pour objectif d'exécuter une loi de commande donnée (séquence de requêtes à envoyer au niveau inférieur), que ce soit en mode de production normale, dégradée ou encore d'urgence.
- **Le suivi** (Zamai, 1997; Trentesaux et Sénéchal, 2002) a pour objectif d'archiver toutes les informations nécessaires aux autres fonctions de Surveillance et de Supervision.
- **La détection** (Combacau, 1991) a comme rôle de détecter l'occurrence d'un symptôme de défaillance.
- **Le diagnostic** (Kempowsky et al., 2004; Toguyeni et al., 2003) a pour objectif de localiser les origines du symptôme de défaillance détecté.
- **Le pronostic** (Boufaied, 2000) a pour rôle de déterminer les conséquences de la défaillance sur le fonctionnement futur de la partie opérative.
- **La décision** (Berruet et al., 1998; Henry et al., 2003) détermine l'état dans lequel la partie opérative et le flux de produits doivent être placés ainsi que les traitements à effectuer pour ramener le SAP en production normale.

Dans la littérature, il existe deux types de mise en œuvre de ces fonctions (Combacau, 1991) : les approches de type "intégrées" et les approches de type "séparées", dont les appellations "intégrées" et "séparées" réfèrent à l'intégration ou non de ces diverses fonctions au sein même de la commande. Le premier type d'approche consiste à prévoir les défaillances possibles puis

d'élaborer hors ligne les traitements associés et enfin de les intégrer aux lois de commande afin de pouvoir, suite à l'occurrence d'un symptôme de défaillance, les appliquer. Les approches intégrées permettent de garantir la réactivité du système de pilotage, ce qui justifie largement leur utilisation dans des domaines critiques tel que le nucléaire. Cependant il est à noter que l'applicabilité de ce type d'approche repose sur la capacité à pouvoir dénombrer l'ensemble des défaillances. Ceci entraîne tout d'abord un temps important pour concevoir le système de pilotage mais également des difficultés d'application dans le cadre des systèmes complexes. Le deuxième type d'approche consiste à séparer la surveillance de la commande. Dans ce cas, le traitement d'une défaillance sera réalisé en ligne, basé sur des raisonnements capables d'identifier la défaillance, et d'élaborer un traitement associé, le tout en associant l'opérateur humain bien entendu. Ce type d'approche est basé sur un modèle du fonctionnement normal de la partie opérative. Partant du principe que tout ce qui n'est pas normal est anormal, le système de pilotage a l'intérêt de pouvoir détecter, en théorie, l'ensemble des défaillances qui peuvent affecter la partie opérative. Toutefois, rien ne garantit que l'ensemble des fonctions arrive à converger dans les temps vers une solution permettant le confinement de la défaillance.

L'ensemble des travaux portant sur les fonctions de Surveillance, Supervision et Commande (SSC) s'accorde à reconnaître que ces fonctionnalités reposent sur l'utilisation de modèles. La section suivante propose une présentation succincte des modèles généralement utilisés.

4.2 Modèles

Un modèle est une représentation mathématique, abstraite et approchée d'un système réel. Il est le résultat d'une démarche complexe qui rend formelle et explicite un ensemble de connaissances informelles. La modélisation a pour objectif d'extraire une partie de ces informations. Elle est guidée par un but, rendre explicites et exploitables les informations nécessaires à la réalisation d'une tâche ou d'un raisonnement.

Dans le cadre des fonctions de SSC, de nombreux modèles formels sont proposés dans la littérature. Les lois de commandes sont généralement exprimées sous forme de Grafcet (Zaytoon et al., 1999) ou de réseaux de Petri (Combacau et al., 2005). L'objectif de ces lois de commande est de générer des ordres de commande. Le formalisme utilisé doit pouvoir exprimer des parallélismes, des séquentialités... Des automates à états temporisés (Bouyer et al., 2005) ou encore des réseaux de Petri (Tromp, 2000) sont proposés pour la détection et le diagnostic pour leur capacité à représenter directement ou indirectement l'ensemble des états en fonctionnement normal/anormal de la partie opérative ainsi que les aspects temporels de son comportement. Sont également utilisées dans le cadre du diagnostic, des équations en logique booléenne (Hu et al., 1999) permettant de représenter les relations entre les capteurs et actionneurs en fonctionnement normal. (Chaillet, 1995) propose également dans le cadre du diagnostic d'exploiter un modèle entité relation représentant la partie opérative d'un SAP dans l'objectif d'orienter la propagation de diagnostic au sein de la structure de pilotage.

L'utilisation d'un formalisme particulier permet de mettre en exergue dans un modèle les informations importantes pour son exploitation. Un aspect important du formalisme employé est sa capacité à offrir des outils de vérification, si ce n'est pour s'assurer que le modèle vérifie soit des propriétés intrinsèques au système modélisé, telles que la vivacité, la réinitiabilité... (Valette

et Künzle, 1994) ou que la méthode utilisant le modèle fournisse un résultat correct (Zaytoon, 1996). La phase de modélisation est une démarche cognitive complexe. Afin de faciliter cette phase, il est intéressant de séparer la phase de capitalisation des informations nécessaires de la phase de génération d'un modèle pour son contexte d'utilisation (Deschamps et al., 2007a). La Figure 1.6 illustre ces différentes phases. Cette démarche permet à l'expert de se concentrer sur les informations nécessaires sans pour autant se préoccuper des difficultés que peuvent apporter l'utilisation d'un outil. Pour une présentation plus détaillée le lecteur pourra se référer plus particulièrement à l'utilisation de l'ingénierie des modèles pour les systèmes de production présentée dans (de Lamotte, 2006).

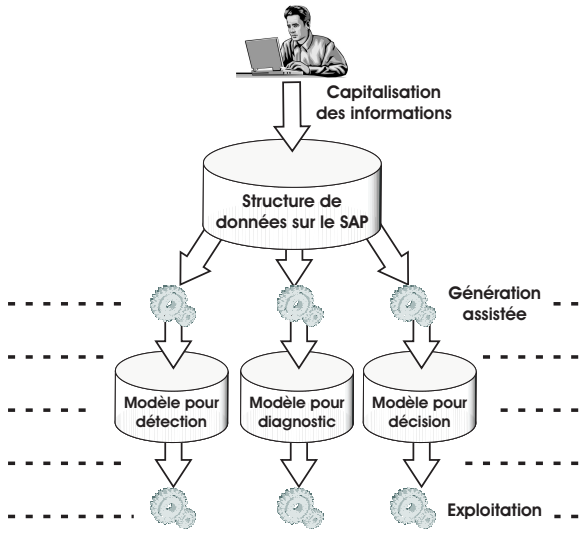


FIG. 1.6 – Phase de capitalisation des informations et phase d'exploitation

Afin de restituer l'ensemble de ces principes dans le contexte de cette thèse, la section suivante propose d'étudier les spécificités des approches de Surveillance, Supervision et Commande (SSC) en fonction du niveau auquel elles se placent.

5 Spécificité des niveaux temps réels de pilotage

Les travaux développés dans cette thèse se placent volontairement au niveau coordination des chaînes fonctionnelles. Toutefois, afin d'appréhender au mieux les problématiques de ce niveau, la section suivante propose une étude plus avancée de la structure et de la spécificité des chaînes fonctionnelles. Dans un deuxième temps, l'étude portera sur les spécificités du niveau coordination.

5.1 Chaînes fonctionnelles

5.1.1 Structure d'une chaîne fonctionnelle

Une chaîne fonctionnelle (Jacquet et al., 1995) est un sous-ensemble d'un système automatisé permettant de réaliser au moins une fonction élémentaire (transférer, positionner, transformer,...). On considérera tout au long de ce mémoire qu'une chaîne fonctionnelle est constituée de chaînes d'action et/ou de chaînes d'acquisition, et de son propre système de

contrôle/commande assimilé au niveau 1 de l'architecture CIM (cf. Figure 1.7). Avec une telle structure, une chaîne fonctionnelle est dans la mesure d'offrir un ou plusieurs services de commande (Henry et al., 2004b) à son niveau supérieur (appelé dans la suite du mémoire le niveau coordination des chaînes fonctionnelles) ; par exemple "sortir vérin", "rentrer vérin", "démarrer moteur", "arrêter moteur", pour une chaîne fonctionnelle comportant une chaîne d'action, ou encore "peser pièce", "lire code barre produit", "détecter pièce" pour une chaîne d'acquisition.

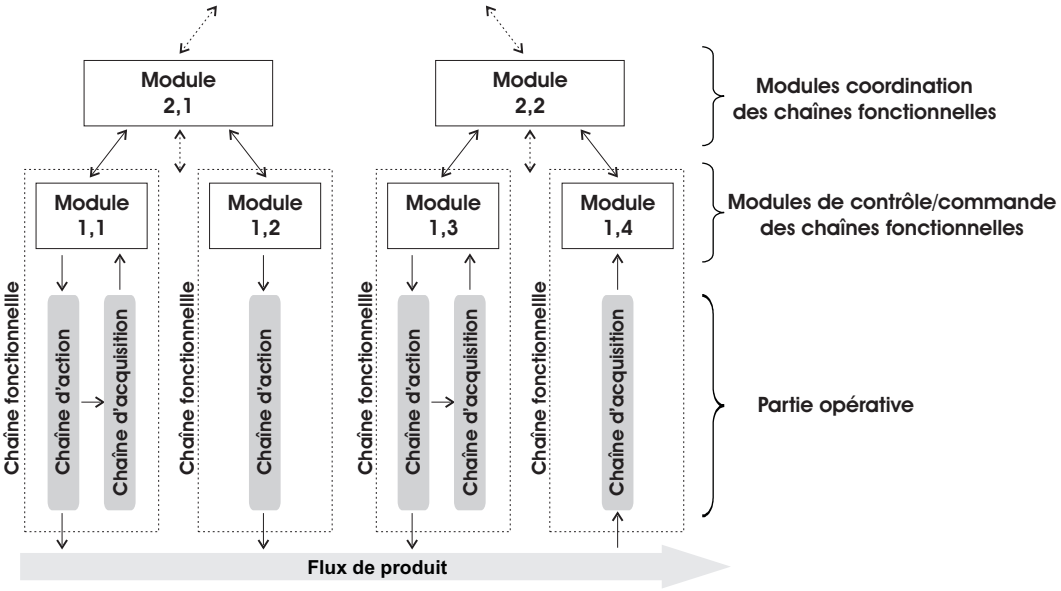


FIG. 1.7 – Coordination des chaînes fonctionnelles

Le choix de distinguer les chaînes fonctionnelles du niveau coordination est issu d'un besoin de dissocier les contraintes liées à l'utilisation particulière faite des actionneurs et les contraintes liées à la technologie des composants de la partie opérative (Henry, 2005). Ce dernier ensemble de contraintes est pris en compte par les modules de contrôle/commande des chaînes fonctionnelles. Le rôle des modules de contrôle/commande des chaînes fonctionnelles est ainsi de transformer les requêtes reçues du niveau coordination en signaux électriques compatibles avec les pré-actionneurs/actionneurs utilisés, et transformer les signaux issus des capteurs en compte-rendus compréhensibles par le niveau coordination.

5.1.2 Spécificités d'une chaîne fonctionnelle

Même si d'un point de vue fonctionnel la structure de pilotage est générique, chaque niveau de cette structure possède ses propres spécificités. En prise directe avec la partie opérative, un module de contrôle/commande a en charge un sous-système de complexité faible, composé dans la plupart des cas d'une unique chaîne d'action. La faible complexité de ce type de niveau se prête bien à l'usage d'une approche dite intégrée de surveillance, supervision et commande (cf. section 4.1) garantissant ainsi la réactivité des modules de contrôle/commande des chaînes fonctionnelles face aux défaillances de la partie opérative. Une chaîne fonctionnelle pourra ainsi être en mesure, en fonction de son niveau d'observation et de ses connaissances sur la partie opérative pilotée, de détecter une défaillance et d'en diagnostiquer les origines. Dans le cas où

elle serait dans l'incapacité de réaliser un service demandé, ces informations accompagneront le compte-rendu renvoyé au niveau coordination.

Il existe dans le cadre de cette communication entre les chaînes fonctionnelles et le niveau coordination une exception au protocole d'appel réponse (cf. section 2.3). Une chaîne fonctionnelle doit être en mesure de pouvoir envoyer des informations quant à un changement intempestif de l'état de la partie opérative ou du flux de produits (Zamai, 1997) (par exemple la présence d'un produit à une position non attendue, le changement d'état d'un actionneur suite à un événement extérieur non prévu comme l'intervention d'un opérateur...). Nous considérerons ainsi qu'une chaîne fonctionnelle peut renvoyer deux types d'informations au niveau coordination :

– Compte-rendu de bonne exécution suite à l'appel à un service.

– Compte-rendu traduisant un dysfonctionnement de la partie opérative (que nous noterons CRA dans la suite de ce manuscrit). Ce compte-rendu pourra éventuellement être accompagné d'informations issues d'un diagnostic effectué au niveau de la chaîne fonctionnelle.

Un dernier point restant à discuter concerne le fonctionnement de la chaîne fonctionnelle en l'absence de chaîne d'acquisition. En effet, dans un contexte industriel il n'est pas envisageable d'associer à chaque chaîne d'action une chaîne d'acquisition pour des raisons évidentes de coûts de l'instrumentation, de développement ou encore de maintenance. Comme nous ne maîtrisons pas le niveau d'implantation des chaînes d'acquisition, le système de pilotage de la chaîne fonctionnelle doit être capable de fonctionner en l'absence d'informations capteurs en s'appuyant par exemple sur un modèle du comportement temporel de la chaîne d'action. Un exemple illustratif est présenté dans la Figure 1.8.

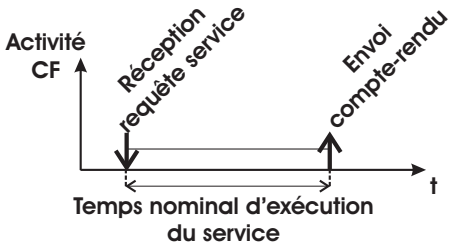


Fig. 1.8 – Utilisation d'un modèle de comportement

Toutes les fonctionnalités des modules de contrôle/commande des chaînes fonctionnelles reposent en partie sur l'implantation de capteurs sur la partie opérative et sur le flux de produits.

5.1.3 Observabilité des chaînes d'action

Une chaîne fonctionnelle peut être munie d'une chaîne d'acquisition permettant de remonter de l'information au niveau contrôle/commande de la chaîne fonctionnelle sur l'état de la chaîne d'action. Le captage de l'information peut être positionné à divers niveaux dans cette chaîne d'action, comme présenté dans la Figure 1.9. Un capteur peut être placé en observation sur :

– l'effecteur (capteur de fin de course placé sur la chambre d'un vérin chargé de surveiller la position du piston),
– l'actionneur (codeur incrémental monté sur l'axe de rotation d'un moteur pas à pas),
– le pré-actionneur (capteurs chargés de surveiller l'ouverture/fermeture des clapets d'un distributeur de pression d'air).

Il est important de noter que la position du captage sur la chaîne d'action a un impact direct sur la confiance à accorder à l'interprétation de l'information vis à vis de l'effet que doit avoir la chaîne fonctionnelle. Plus le capteur est positionné près de l'effecteur, plus nous pouvons accorder de la confiance à l'interprétation qui est faite du signal fourni caractérisant l'effet.

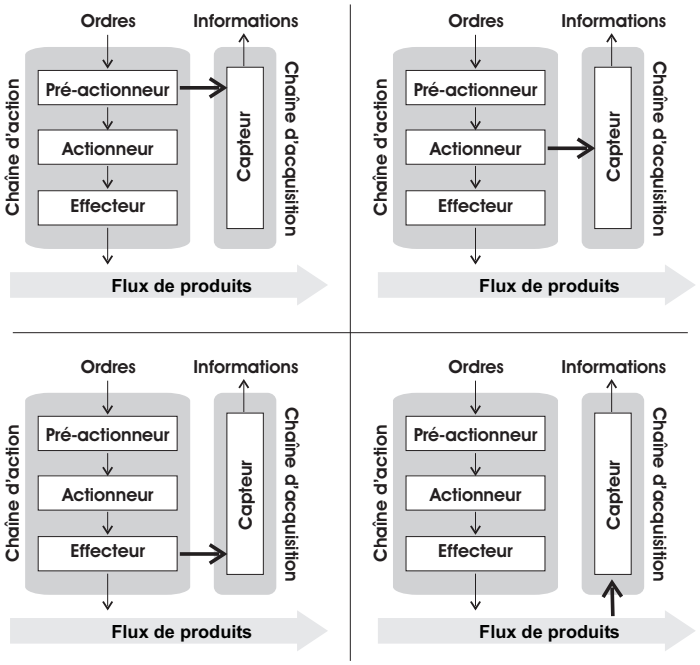


FIG. 1.9 – Placement du capteur

5.1.4 Observabilité sur le flux de produits

Un grand nombre de capteurs sont placés afin d'acquérir de l'information sur le flux de produits (cf. Figure 1.9). Généralement, ils permettent de surveiller certaines positions des produits (leurs caractéristiques après avoir subi des transformations, ...) informations requises directement par le niveau coordination dont le rôle est bien d'agir sur le flux de produits. Cependant, des capteurs "produits" peuvent être gérés au niveau le plus bas, à l'intérieur de chaînes fonctionnelles lorsqu'il s'agit par exemple d'asservir la position d'un outil sur la forme géométrique du produit.

Après avoir étudié les spécificités des chaînes fonctionnelles, la section suivante propose de voir plus en détail le rôle du niveau coordination des chaînes fonctionnelles.

5.2 Niveau coordination des chaînes fonctionnelles

Afin de réduire la complexité de la partie opérative pilotée au niveau coordination des chaînes fonctionnelles, le niveau d'abstraction qui en est fait est bien entendu plus important que celui du pilotage des chaînes fonctionnelles. Les chaînes fonctionnelles sont vues comme offrant un ensemble de services. Cependant de nombreuses informations apparaissent à ce niveau. En premier lieu, ce niveau devra prendre en compte les interactions entre les chaînes fonctionnelles. Par exemple certains services ne devront pas être lancés en parallèle afin d'éviter d'éventuelles collisions. De plus une des dimensions du niveau coordination est sa capacité à piloter le flux de produits (Henry, 2005), de décider donc de son routage dans le SAP en fonction de la charge des machines, de choisir l'ordre des transformations qu'il devra subir... La complexité résultante d'une telle diversité et indéterminismes des situations rend difficile ici le recours à des approches de surveillance et supervision intégrés à la commande pour faire face aux aléas de fonctionnement. Des approches séparées ou mixtes seront davantage préconisées (cf. section 4.1).

6 Conclusion

Nous avons présenté dans ce chapitre la problématique générale du pilotage des SAP. Nous avons précisé leur rôle et leur structure organisationnelle. Plongé dans un contexte incertain lié à l'existence d'aléas de fonctionnement nous avons montré la nécessité d'intégrer au système de pilotage des boucles réactives capables de confiner ces aléas au plus bas de la structure. Ces boucles réactives s'appuient sur un ensemble de fonctionnalités telles que le diagnostic par exemple. Chacune d'entre elles peut se voir mise en œuvre selon deux techniques. Dans les modules de contrôle/commande des chaînes fonctionnelles, une mise en œuvre de type intégrée alors que les niveaux coordination se verront appliquer des approches séparées voire mixte. C'est dans un tel contexte que le chapitre suivant se propose de décrire la problématique à laquelle ce mémoire est dédié.

Chapitre 2

Problématique

1 Introduction

Jusqu'ici, nous avons présenté la structure de pilotage et plus particulièrement les niveaux coordination dans lesquels nos travaux évoluent. Plongés dans un contexte soumis aux aléas de fonctionnement de la partie opérative, le recours à un processus réactif est requis.

C'est au cœur de ce processus que la problématique traitée dans ce manuscrit se place. Aussi, le premier paragraphe de ce chapitre se propose de rentrer encore plus au cœur de ce processus réactif, appelé par la suite le processus de reconfiguration. Après quoi, et pour les besoins d'une des fonctions de ce processus, le diagnostic, le paragraphe suivant propose une analyse des origines d'un dysfonctionnement d'une chaîne fonctionnelle. Sur cette base, la section 4 présente les contraintes temporelles qui s'exercent sur la fonction diagnostic. Enfin, la section 5 présente le cahier des charges des travaux développés dans ce manuscrit.

2 Le processus de reconfiguration

Le processus de reconfiguration (Berruet et al., 2007) se décline selon deux axes distincts : une reconfiguration de la partie opérative (matérielle), ou une reconfiguration de la partie commande (logicielle). La reconfiguration de la partie opérative consiste à intervenir sur le plan matériel de la partie opérative afin de modifier ses capacités. Il est alors possible de compenser les services perdus suite à l'occurrence d'une défaillance par l'introduction suite à la reconfiguration de nouveaux services. Ces travaux se rapportent plus particulièrement aux systèmes manufacturiers reconfigurables (Mehrabi et al., 2000). La deuxième solution consiste à posséder une commande reconfigurable capable de s'adapter et d'exploiter les services encore disponibles offerts par la partie opérative. Afin d'obtenir une commande reconfigurable deux solutions sont proposées dans la littérature :

- l'adaptation en ligne de la commande élaborée hors ligne, basée sur l'idée que l'ensemble des potentialités de la partie opérative a été pré-intégrée à la commande (Gouyon et al., 2004; Huvenoit et al., 1992),
- synthétiser en ligne une loi de commande déterministe exploitant les capacités de la partie opérative (Henry et al., 2004a; de Lamotte, 2006).

Que ce soit une reconfiguration de la partie opérative ou une reconfiguration au niveau de la commande, le processus de reconfiguration se base nécessairement sur une connaissance des services perdus suite à l'occurrence d'un symptôme de la défaillance à travers un modèle des capacités opératoires de la partie opérative. Cette connaissance est généralement issue de la fonction diagnostic qui a pour but de localiser puis d'identifier la cause de la défaillance. Il est à noter ici que le rôle d'une telle fonction diagnostic dans le cadre du processus de reconfiguration n'est pas uniquement d'identifier la cause de la défaillance dans un objectif de maintenance mais également d'identifier la défaillance ainsi que ses conséquences directes ou indirectes sur les capacités de la partie opérative et sur l'état du flux de produits, ceci dans l'objectif de mettre à jour le modèle des capacités opératoires de la partie opérative. C'est précisément dans ce contexte que s'inscrivent les travaux présentés dans ce mémoire. L'approche proposée visera à apporter **une contribution dans le domaine du diagnostic de services** dans l'objectif d'une reconfiguration du système de pilotage au niveau coordination des chaînes fonctionnelles.

3 Origine de la réception d'un CRA et conséquences sur les capacités opératoires

Le rôle de la fonction diagnostic qui sera traitée dans ce manuscrit va être, suite à un CRA, d'en déterminer dans un premier temps les origines possibles. Dans un deuxième temps, à partir des origines, elle devra analyser les conséquences qui ont pu modifier les capacités opératoires et le flux de produits. Dans ce cadre il est proposé dans cette section d'étudier, d'un point de vue coordination des chaînes fonctionnelles, les différents cas qui peuvent conduire à la réception d'un CRA, ainsi que ses conséquences.

3.1 Défaillance d'une chaîne fonctionnelle

La première origine possible de la réception au niveau coordination CRA peut être une défaillance de la chaîne fonctionnelle exécutant le service. La conséquence de cette défaillance se décline selon trois points :

- Indisponibilité de toute ou partie des services offerts par la chaîne fonctionnelle, due à une modification de ses caractéristiques physiques.

- Perte possible de la connaissance de l'état de la chaîne fonctionnelle. Il est nécessaire d'envisager que l'état de la chaîne fonctionnelle, correspondant à l'ensemble des valeurs des variables la caractérisant, soit différent de l'état attendu dû à une non conformité de l'effet du service exécuté.

- Perte de la connaissance de l'état du produit, dans le cas où le service a pour objectif de modifier les caractéristiques d'un produit.

Toutefois, la conséquence de la défaillance reste localisée à la chaîne fonctionnelle exécutant un service et éventuellement aux produits dont les caractéristiques ont été modifiées par l'exécution du service. Au delà de la prise en compte de la défaillance de la chaîne fonctionnelle exécutant le

service, le diagnostic doit s'intéresser à la propagation possible d'une défaillance comme origine d'un CRA.

3.2 Propagation d'une défaillance première

Une deuxième origine possible de la réception d'un CRA réside dans la propagation d'une défaillance première (Abdelwahed et al., 2003). Deux contextes différents doivent être distingués :

- Un vecteur de propagation à considérer est la partie opérative elle même. Ici, des services sont lancés uniquement dans l'objectif de modifier l'état de la partie opérative pour qu'il soit compatible avec le lancement d'autres services. Prenons l'exemple d'une pièce qui doit être percée, et considérons qu'il soit nécessaire pour la réalisation du service de perçage de lancer au préalable un service de chargement d'un foret sur le poste de perçage. Une défaillance durant la réalisation du service de positionnement du foret peut également induire la mauvaise réalisation du service de perçage.

- Les objectifs de production consistant à transformer successivement la matière première pour obtenir le produit fini ; le produit peut propager également une défaillance non initialement détectée. Prenons l'exemple d'une pièce qui doit être successivement percée puis taraudée. Si une défaillance, liée à la casse d'un foret, survient lors de l'exécution du service de perçage de la pièce sans être détectée, cela peut induire une mauvaise réalisation du service de taraudage. Dans ce cas, le vecteur de propagation s'avèrera être le flux de produits.

L'étude par le diagnostic de l'ensemble des propagations d'une défaillance ayant pu conduire à la réception d'un CRA permet de retrouver les défaillances premières à son origine possible (cf. section 3.2 du chapitre 1). La recherche de ces propagations de défaillance doit s'appuyer sur les lois de commandes exécutées (Hu et al., 1999) car elles détiennent l'ordre de lancement des services, et donc en partie les vecteurs possibles de propagation. Cependant, l'origine d'une propagation peut provenir également d'une modification de l'état d'une chaîne fonctionnelle ou du flux de produits non liée au lancement d'un service :

- lors d'une propagation de défaillance, un produit peut être amené à évoluer sans pour autant qu'un service soit lancé. Ceci peut alors l'amener dans un état non attendu (par exemple le déplacement d'un produit par sa gravité). Dans ce cas un peu particulier, la fonction diagnostic au niveau coordination doit être en mesure de pouvoir retrouver les évolutions qui ont pu conduire le flux de produits dans un tel état observé.

- le changement spontané de l'état d'une chaîne fonctionnelle (par exemple l'arrêt d'un moteur suite au déclenchement de sa protection) doit être également pris en compte en tant qu'origine possible d'une propagation.

3.3 Conséquence d'une propagation de défaillance

Afin de connaître les services affectés suite à l'occurrence d'un symptôme de défaillance, le diagnostic doit non seulement rechercher les défaillances premières à l'origine possible du CRA mais également les conséquences de leurs propagations. Le diagnostic devra pour chacun des services lancés à une date postérieure à l'occurrence des défaillances premières possibles vérifier que leurs exécutions n'ont pas été affectées. Les figures 2.1 et 2.2 schématisent les deux cas de figure possibles, détaillés ci-dessous :

– la conséquence est limitée à la non conformité de l'effet d'un service sur l'état de la partie opérative et du flux de produits. Prenons l'exemple d'un service qui a pour finalité de transférer une pièce d'une position A à une position B (correspondant au service Y de la Figure 2.1). Si une défaillance, durant l'exécution d'un service X antérieur, conduit à la non présence de la pièce à la position A, l'exécution du service aura pour seule conséquence la non présence de la pièce en B. Cependant les services de la chaîne fonctionnelle offrant le service de transfert de la pièce ne seront en aucun cas affectés par cette propagation.

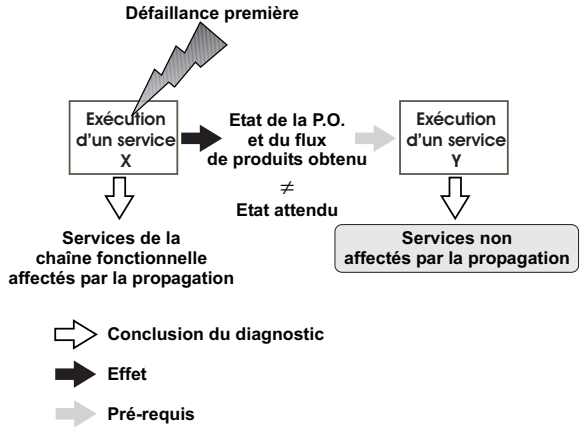


FIG. 2.1 – Conséquence d'une propagation de défaillance sur la réalisation des services

– La conséquence n'est pas limitée à l'effet d'un service mais la propagation de la défaillance affecte également la disponibilité des services offerts par les chaînes fonctionnelles. Considérons ce même service de transfert d'une pièce réalisé par la sortie d'un vérin (correspondant au service Y de la Figure 2.2), mais conditionné cette fois-ci par la rentrée d'un deuxième vérin (service Z) pour éviter une collision (contrainte de sécurité). Si une défaillance survient durant l'exécution du service de rentrée du vérin, cela induit non seulement le non transfert de la pièce dû au blocage du vérin, mais également à une dégradation potentielle des vérins. En conséquence, les services offerts par ces chaînes fonctionnelles sont potentiellement affectés par la propagation d'une défaillance.

3.4 Propagations multiples d'une défaillance

Lors de l'étude de propagations des défaillances, le diagnostic doit prendre en compte la propagation multiple d'une défaillance, correspondant à plusieurs propagations issues de la

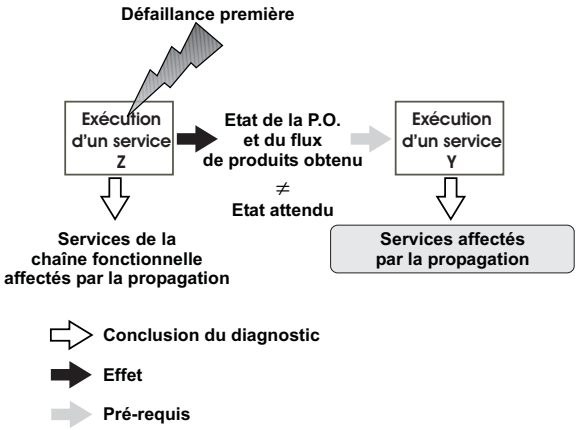


FIG. 2.2 – Autre conséquence d'une propagation de défaillance

même défaillance première. Par exemple la Figure 2.3 schématise une défaillance première (durant l'exécution du service X), à l'origine de la réception d'un CRA durant l'exécution du service Y, qui a également entraîné une deuxième propagation durant l'exécution d'un service Z.

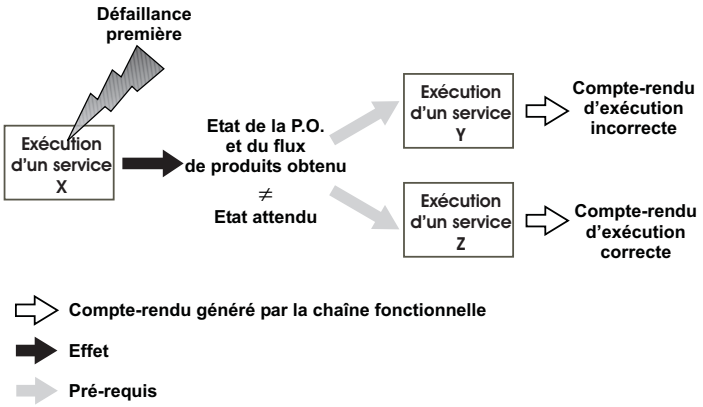


FIG. 2.3 – Propagations multiples d'une défaillance

3.5 Propagations de défaillances multiples

La composition de plusieurs défaillances premières est également une origine possible de la réception d'un CRA. Ce phénomène est couramment appelé dans la littérature défaillances multiples (Pickard et al., 2005). Au vue de l'impact de plus en plus important de ces défaillances (cf. chapitre 1 de (Hamidi, 2005)) sur le fonctionnement des systèmes, il est primordial de les considérer dans le cadre du diagnostic. Ce type de propagations est illustré par la Figure 2.4. Dans cet exemple l'exécution incorrecte du service Z est la conséquence de la réalisation incorrecte du service X et du service Y. Ce qui signifie que si uniquement le service X ou le service Y est mal exécuté, la réalisation du service Z n'est pas affectée.

L'étude de l'ensemble de ces cas, dans le cadre d'un diagnostic préalable à une reconfiguration, est soumis à de fortes contraintes temporelles comme développé dans la section suivante.

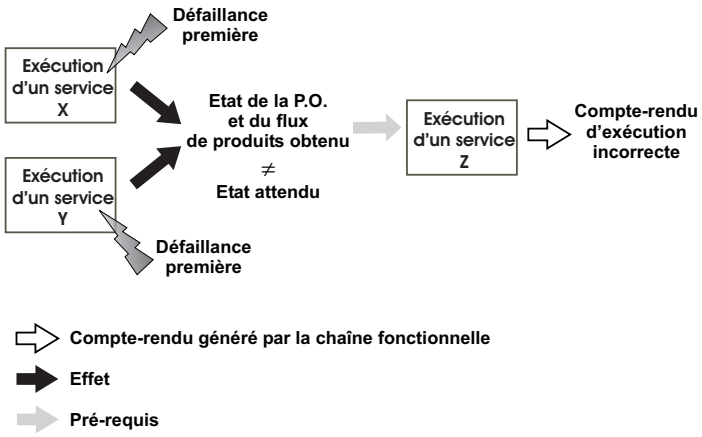


FIG. 2.4 – Propagation de défaillances multiples

4 Contraintes temporelles pour la reconfiguration

Une requête reçue par le niveau coordination est accompagnée d'un objectif à réaliser ainsi qu'une durée allouée à l'exécution du service demandé (Henry, 2005). Cette durée comprend non seulement celle nécessaire pour réaliser en temps normal l'objectif plus également une marge temporelle permettant de laisser au niveau coordination une flexibilité décisionnelle en présence d'un dysfonctionnement (cf. section 2.1 du chapitre 1). En l'absence de dysfonctionnements avérés, un compte rendu est généré vers ce niveau supérieur (cf. partie fonctionnement normal de la Figure 2.5). Dans le cas de la réception d'un CRA, le niveau de coordination doit être capable de mettre en place un processus de reconfiguration afin d'atteindre l'objectif avant la fin de la durée allouée. Sans cela, il devra propager son incapacité à répondre à l'objectif à son niveau supérieur. La durée restante (cf. partie fonctionnement anormal de la Figure 2.5) devra être exploitée par ce processus de reconfiguration (Diagnostic, décision, synthèse d'une nouvelle loi de commande) mais également par l'exécution du travail restant à faire pour atteindre l'objectif. Nous devons noter ici que cette durée peut être fortement réduite selon la date d'occurrence du CRA comme le montre la Figure 2.6.

5 Cahier des charges

Afin de résoudre une telle problématique qui part de l'occurrence d'un CRA, jusqu'à la mise à jour du modèle des capacités opératoires encore offertes par le sous-système piloté, nous nous proposons dans cette section de dévoiler la démarche générale retenue ainsi que les hypothèses principales sur lesquelles s'appuient nos travaux.

5.1 Hypothèses de travail

L'approche de diagnostic proposée dans ce mémoire s'appuie sur un ensemble d'hypothèses de travail qui seront progressivement dévoilées tout au long de ce manuscrit. Cependant nous souhaitons ici en dévoiler trois principales qui fermeront le contexte de notre étude.

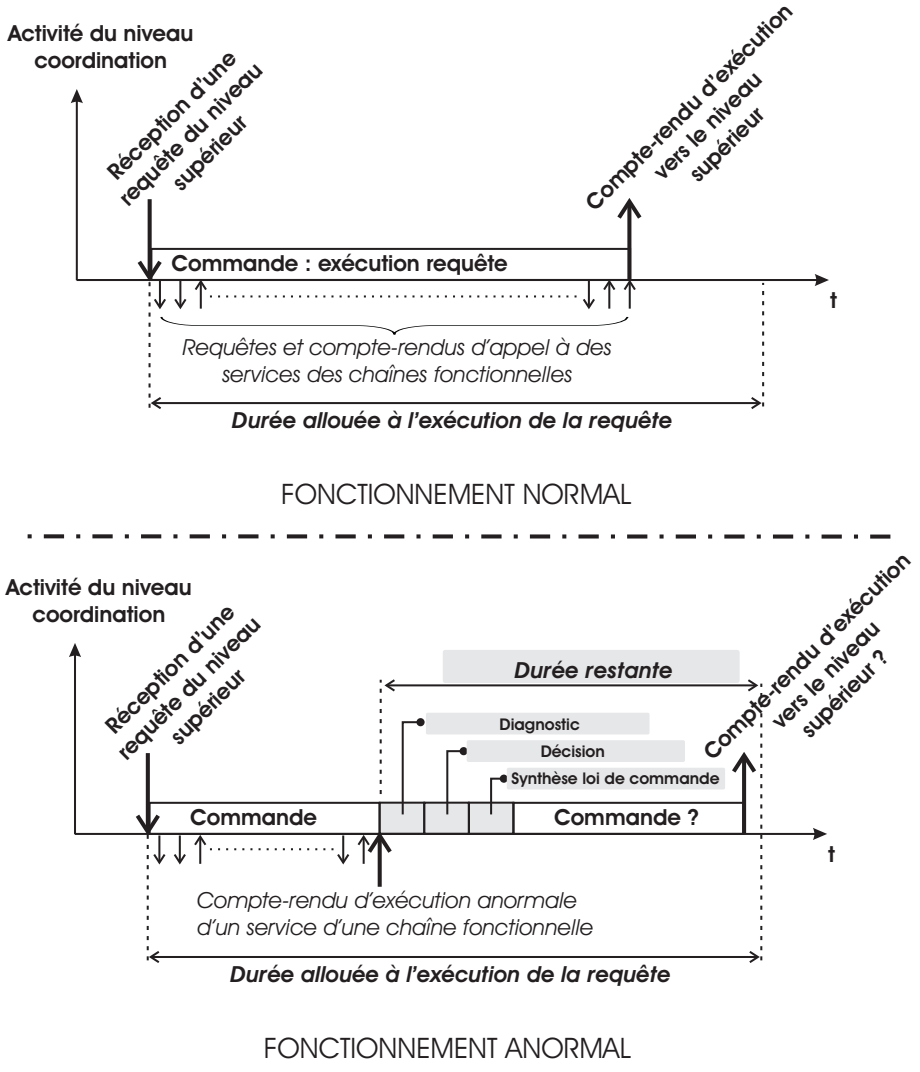


FIG. 2.5 – Contrainte temporelle pour la reconfiguration

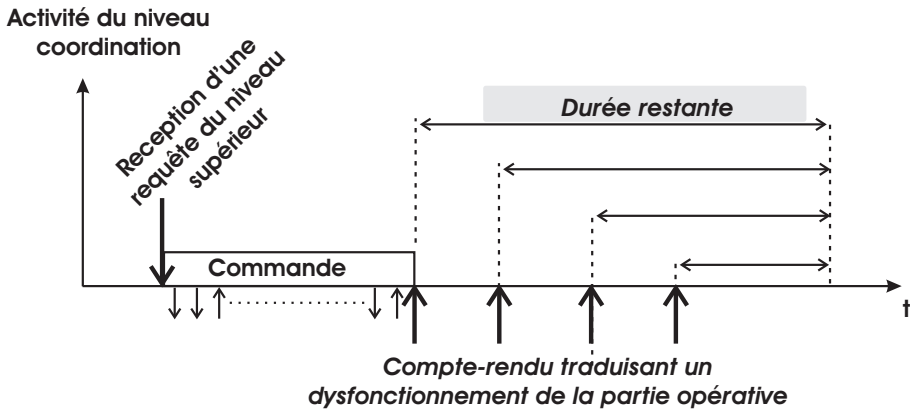


FIG. 2.6 – Variation de la durée restante pour la reconfiguration

- Un module de contrôle/commande d'une chaîne d'action et/ou d'acquisition est basé sur une approche intégrée de surveillance et de supervision à la commande. Il dispose de tous les moyens et d'une connaissance parfois partielle de l'état de la partie opérative lui permettant de rendre compte au travers de comptes-rendus de la vision qu'il a de son état de fonctionnement.

- Le comportement du flux de produits est une problématique du niveau coordination. Le niveau coordination s'appuie sur des chaînes fonctionnelles d'acquisition délivrant des services de surveillance de l'état du flux de produits. L'utilisation de tels services de surveillance est intégrée à la commande du niveau coordination.

- Un module de coordination de chaînes fonctionnelles est basé sur une approche séparée de diagnostic à la commande.

5.2 Vers une approche de diagnostic des services

Au sein de cette dernière section du chapitre, nous nous proposons de reprendre les idées principales qui vont orienter notre approche de diagnostic.

5.2.1 Diagnostic de services, de l'état de la PO et du flux de produits

Suite à la réception d'un CRA, le diagnostic doit retrouver les propagations de défaillances possibles pour remonter aux défaillances premières ayant conduit à ce dysfonctionnement. Après quoi, leurs conséquences d'une part sur la réalisation des autres services gérés par le niveau coordination et d'autre part sur l'état de la partie opérative et du flux de produits doivent être étudiés. Nous appellerons ce processus **le diagnostic de services**. Ce dernier s'articulera autour de (cf. Figure 2.7) :

- l'exploitation des caractéristiques des services offerts par les chaînes fonctionnelles. Au vu de la complexité intrinsèque du niveau coordination, il serait utopique de vouloir recenser l'ensemble des défaillances possibles ainsi que leurs conséquences sur la partie opérative et sur le flux de produits. En conséquence ces caractéristiques seront exprimées dans un contexte de fonctionnement normal.

- L'exploitation de la loi de commande, qui est en quelque sorte la mémoire du passé, afin de connaître les services qui ont été exécutés avant la réception d'un CRA.

- La recherche des origines possibles dans les services précédemment exécutés à l'occurrence du CRA sera limitée à un sous ensemble du passé délimité par la confiance attestée ou non par les chaînes fonctionnelles vis à vis des services exécutés (cf. section 5.1 du chapitre 1).

A partir de l'ensemble des informations décrites ci-dessus, le diagnostic devra être en mesure de prendre en compte les différents cas de figures suivants :

- propagation d'une défaillance à travers la partie opérative ou par l'intermédiaire du flux

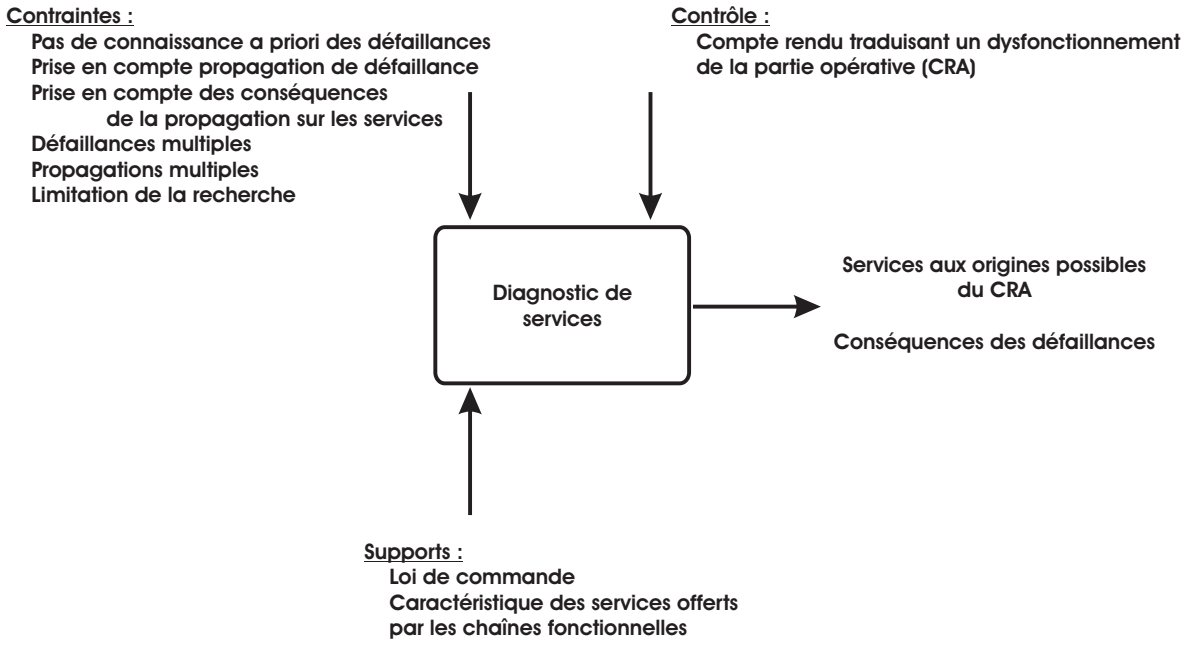


FIG. 2.7 – Problématique du diagnostic des services et de l'état de la partie opérative et du flux de produits

de produits,

- propagations multiples d'une défaillance à travers la partie opérative,
- défaillances multiples à l'origine du CRA,
- conséquences d'une propagation de défaillance sur les services et sur l'état de la partie opérative et du flux de produits.

5.2.2 Mise à jour de la description des capacités de la partie opérative

Le résultat de diagnostic (mise en évidence des défaillances possibles ainsi que leurs conséquences possibles sur les services exécutés) doit être ensuite projeté sur les caractéristiques des services correspondant aux capacités actuelles encore offertes par la partie opérative.

La Figure 2.8 reprend le processus de reconfiguration avec les différentes étapes décrites précédemment. La phase suivante à la mise à jour étant une phase décisionnelle, elle exploite le résultat du diagnostic et de la mise à jour.

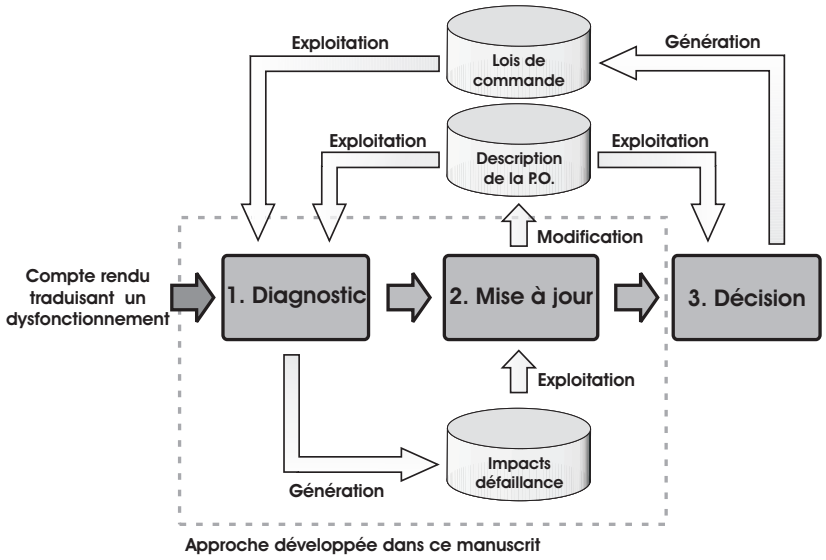


FIG. 2.8 – Principe de la reconfiguration

6 Conclusion

Dans le cadre de ce chapitre, nous avons tout d'abord présenté le processus de reconfiguration nous permettant de définir les objectifs de la fonction diagnostic proposée. Forts de ces objectifs, nous avons étudié du point de vue du niveau coordination, les origines et les conséquences des défaillances des chaînes fonctionnelles afin de mettre en exergue les difficultés à prendre en compte et ainsi structurer notre vision du système de diagnostic à développer. Ce dernier sera développé selon 2 activités :

- une activité de diagnostic capable d'une part de rechercher les origines d'un compte-rendu traduisant un dysfonctionnement de la partie opérative, et d'autre part les conséquences des dysfonctionnements obtenus sur les services précédemment lancés. Pour cela, le mécanisme proposé devra être en mesure d'explorer le fonctionnement passé de la partie opérative. L'originalité de cette fonction réside en premier lieu dans son positionnement au niveau coordination et ainsi dans la nécessité de prendre en compte les spécificités et les difficultés, mises en exergue dans ce chapitre. Une deuxième originalité découlant de la première est la nécessité de se baser sur un modèle du fonctionnement passé intégrant les caractéristiques comportementales des services exclusivement dans un contexte de fonctionnement normal.

- Un processus de mise à jour du modèle des capacités de la partie opérative afin de permettre l'exploitation du résultat de la fonction diagnostic dans la suite du processus de reconfiguration. Toute la difficulté réside dans l'interprétation du résultat de diagnostic portant sur des services exécutés afin de les projeter sur les capacités futures.

Notre problématique étant désormais clairement affichée dans le domaine du diagnostic, le chapitre suivant propose une étude bibliographique de ce domaine.

Chapitre 3

Positionnement dans la littérature existante

1 Introduction

Après avoir précisé le périmètre de nos travaux et posé la problématique à laquelle ces travaux apportent une solution, nous proposons ici au lecteur de dresser un état de l'art général du domaine de diagnostic. Nous réalisons aussi, dans ce chapitre, une étude critique des principales approches. Cinq approches sont ainsi analysées : les systèmes experts, les approches basés sur la reconnaissance des scénarios, les approches diagnostic des SED, celles du diagnostic logique et pour terminer l'extension qui en est faite dans le cadre du diagnostic de plans.

2 Systèmes experts

2.1 Principe

Une des techniques utilisée dans le cadre du diagnostic est l'utilisation de systèmes experts (Bohez et Thieravarut, 1997; Zwingelstein, 1995; Chaillet, 1995). Les systèmes experts se présentent sous formes de règles empiriques reliant l'effet à la cause. Ces règles sont fondées sur l'expérience de l'expert plutôt que sur une connaissance de la structure et du comportement de la partie opérative (ces systèmes sont dits de connaissance de surface). La fonctionnalité d'un système expert est de trouver la cause de ce qui a été observé en parcourant les règles par des techniques classiques de l'intelligence artificielle telles que le chaînage arrière.

2.2 Avantages/Inconvénients

La qualité première d'un système fonctionnant avec de telles règles est son efficacité au niveau temps de calcul. Il suffit d'attendre que survienne l'occurrence d'un symptôme de défaillance et de "sauter" directement aux conclusions. Il n'y a aucun raisonnement compliqué et coûteux en temps de calcul à tenir, aucun calcul intermédiaire à effectuer. Ceci est rendu possible car l'expert qui a produit ces règles a tenu les raisonnements nécessaires auparavant. Celui-ci n'a ensuite enregistré dans le système que les symptômes et les conclusions finales de son raisonnement. Ces règles peuvent être vues comme des raccourcis efficaces de raisonnements

généralement beaucoup plus longs.

Ce qui fait la force d'un système expert, c'est le jeu de règles efficaces résultant de l'expertise d'un humain. Mais c'est aussi le premier inconvénient d'une telle approche : elle est totalement dépendante de l'expertise faite sur le système à piloter. Ainsi, les systèmes experts sont sujets aux défauts liés à l'expertise elle-même. Ceci est d'autant plus gênant que l'acquisition de la connaissance est longue et difficile. Dénombrer l'ensemble des situations possibles peut rapidement devenir complexe pour le raisonnement humain, d'autant plus dans le contexte des SAP dans lequel nous évoluons. De plus les règles sont fixées et ne sont pas robustes face à des situations non reconnues de par l'absence de règles correspondant à ces situations. Les systèmes experts manquent de généricité. Les règles acquises sur une partie opérative ne peuvent pas être utilisées sur une autre car elles sont dépendantes de la structure de celle-ci. Un dernier inconvénient réside dans le problème de l'évolution du système. Si la partie opérative évolue par des ajouts de composants, le système de règles est à remettre en cause. Une nouvelle expertise doit être faite afin que le système expert soit toujours pertinent.

3 Reconnaissance de scénario

3.1 Principe

Les approches de reconnaissance de scénario considèrent les relations temporelles plus ou moins complexes (Toguyeni et al., 1996) entre les observations. Le formalisme des chroniques, encore appelées scénarios, est tout a fait adapté à la représentation de ces relations. Un modèle de chronique (Boufaied, 2003) est constitué d'un ensemble d'observations et d'un ensemble de contraintes temporelles entre les instants d'occurrences de celles-ci. Les modèles de chroniques représentent les évolutions de la partie opérative, traduisant le fonctionnement attendu (normal) du système ou des situations de défaillance.

Le diagnostic d'un système à l'aide de chroniques est basé sur la reconnaissance en ligne de modèles de chroniques. Le principe de la reconnaissance d'une chronique est le suivant. A chaque observation, un ensemble de chroniques candidates sont maintenues. Elles correspondent à un ensemble d'instances de modèles de chroniques pour lesquelles l'ensemble des observations reçues est compatible avec les contraintes temporelles contenues dans ce modèle de chronique. A la réception d'une nouvelle observation, les chroniques qui ne sont pas compatibles sont éliminées de l'ensemble des chroniques et celles qui peuvent débuter avec l'arrivée de cette nouvelle observation y sont ajoutées. Une chronique est reconnue lorsque tous les événements représentés dans le modèle de chronique ont eu lieu dans l'ordre et les délais notifiés dans le modèle. Une fois qu'une chronique est reconnue, l'information de diagnostic associée à cette chronique est notifiée. Des outils de reconnaissance de chroniques mettent en œuvre ce principe comme CRS (Chronicle Recognition System)(Dousson, 1994).

3.2 Avantages/Inconvénients

Les systèmes de reconnaissance de scénarios sont des outils efficaces pour la reconnaissance en ligne de situations connues. Le principal inconvénient est l'acquisition des chroniques. Elles sont en effet fondées sur une connaissance de surface du système qui demande une certaine expertise.

Afin de diminuer cette dépendance à l'expertise du système, des approches ont été développées afin d'acquérir automatiquement ces chroniques. Dans (Dousson et Du'o'ng, 1999), les auteurs présentent un outil d'acquisition FACE (Frequency Analyser for Chronicle Extraction). Cependant, subsiste encore le problème du diagnostic associé à une chronique reconnue. Il est en effet nécessaire qu'un expert soit en mesure de dire que tel scénario d'observations correspond à tel scénario de défaillance.

4 Diagnostic des Systèmes à Événements Discrets (SED)

4.1 Objectif et principe

Le diagnostic des SED (Sampath et al., 1995, 1996; Lafortune et al., 2001) consiste à détecter et à identifier des événements inobservables qui ont lieu durant le fonctionnement du système en se basant sur le modèle du système et sur les séquences d'événements observables obtenues auprès des capteurs. Les événements inobservables à détecter font partie du modèle du système et représentent par exemple des défaillances dans le fonctionnement du système.

A partir du modèle du système et éventuellement d'un modèle traduisant les spécifications du système de commande (équivalent à la loi de commande), ces approches consistent à générer un diagnostiqueur représentant un modèle évolué riche du comportement du système, qu'il soit normal ou défaillant. Un diagnostiqueur peut être vu comme un observateur étendu pour un système particulier qui fournit l'état courant du système et les défaillances passées potentielles ayant conduit le système dans cet état (cf. Figure 3.1).

Fig. 3.1 – Objectifs d'un diagnostiqueur

Les outils les plus utilisés dans le cadre du diagnostic des SED sont les automates à états et les réseaux de Petri. L'automate à états est un outil adapté pour la modélisation du comportement du système sous forme d'événements observables et non observables. Ce modèle permet de décrire les évolutions à travers des séquences d'événements qui traduisent l'état du système. C'est un outil largement utilisé dans les méthodes de diagnostic des SED surtout lorsqu'il s'agit de décrire le comportement complet du système (Sampath et al., 1995).

Les réseaux de Petri sont également utilisés pour le diagnostic des SED dans de nombreuses approches (Soldani et al., 2006; Valette et Künzle, 1994). Par exemple dans (Genc et Lafortune, 2003), un réseau de Petri labellisé modélise la partie opérative en décrivant son comportement normal et anormal. Un label, ou étiquette, est une indication sur le type d'événement présent sur les transitions du réseau de Petri qu'il soit observable ou non observable. De ce réseau de Petri, (Genc et Lafortune, 2003) expriment un diagnostiqueur décrivant toutes les situations observables possibles à partir d'une situation afin de pouvoir identifier les défaillances. Un des

grands avantages des approches basées sur les réseaux de Petri est lié à l'outil mathématique supporté par les modèles. Ainsi, les matrices d'incidences avant et arrière permettent de réaliser des estimations de séquences, ou même de vérifier certaines propriétés du modèle (Deschamps et al., 2004).

4.2 Avantages/Inconvénients

Une des forces des approches de diagnostic des SED est leur capacité à garantir qu'un ensemble de défaillances sera détecté et diagnostiqué à travers la notion de diagnosticabilité (Sampath et al., 1995). Un système est dit diagnosticable pour des sous-ensembles de défaillances et pour un ensemble d'événements observables s'il est possible de détecter l'occurrence de n'importe quelle défaillance appartenant à un des sous-ensembles de défaillances. Un deuxième point fort de ces approches est leur efficacité d'un point de vue réactivité. La synthèse des diagnostiqueurs hors ligne permet de garantir de bonnes performances temporelles pour fournir le résultat de diagnostic suite à l'occurrence d'une défaillance.

Le principal défi dans le domaine du diagnostic des systèmes à événements discrets est la gestion de la complexité des systèmes à diagnostiquer. Il est irréaliste d'aborder les systèmes complexes par des approches de type centralisée classique, qui entraînent nécessairement l'explosion de la taille des modèles. Des approches de diagnostic décentralisé sont donc proposées (Qiu et Kumar, 2006; Debouk et al., 2000; Genc et Lafortune, 2006; Boel et van Schuppen, 2002; Benveniste et al., 2005; da Silveira et al., 2002). Elles consistent à construire des diagnostiqueurs s'appuyant uniquement sur des modèles locaux des composants du système. Chaque diagnostiqueur est en mesure de fournir un diagnostic local au composant. Une seconde étape consiste à construire un diagnostic global par fusion des diagnostics locaux. Toutefois, ce type d'approche ne permet pas de tenir compte des défaillances issues des contraintes inter-composants. Les travaux présentés dans (Philippot, 2006) proposent dans ce sens, la construction d'un coordinateur basé sur un ensemble de règles gérant ces interactions, ceci revenant en quelque sorte à mettre en place un système expert pour la coordination des diagnostiqueurs locaux.

5 Diagnostic logique

5.1 Objectif et principe

Le diagnostic logique (cf. chapitre 1 (Dubuisson, 2001) rédigé par P. Dague) se base sur les principes du diagnostic à base de modèles ou encore appelé diagnostic à partir des principes premiers. Cette méthode a vu le jour aux États-Unis au milieu des années soixante-dix et a été formalisée au début des années quatre-vingt. Un nombre croissant de travaux ont été menés depuis et cette problématique est devenue un domaine de recherche à part entière de l'intelligence artificielle. Les articles les plus marquants dans ce domaine et publiés avant 1991 sont regroupés dans (Hamscher et al., 1992). Le diagnostic à base de modèles ne nécessite pas de connaître a priori les défaillances pouvant affecter un système pour pouvoir le diagnostiquer : modéliser le fonctionnement normal est suffisant. L'idée fondamentale est de comparer le fonctionnement réel du système observé par l'intermédiaire de capteurs et son fonctionnement prédit grâce aux modèles de bon comportement. Le résultat de cette comparaison permet d'établir un diagnostic de cohérence. Toute contradiction entre les observations et les prédictions

déduites des modèles est nécessairement la manifestation d'une ou plusieurs défaillances. Sur ces principes une théorie logique de diagnostic a été formalisée (Reiter, 1987). Le problème du diagnostic peut s'énoncer de la façon suivante : supposons connue une description d'un système ainsi qu'un ensemble d'observations sur son comportement. Suite à la manifestation d'une défaillance, l'observation est en conflit avec l'état attendu du système si chacun des composants était en état de fonctionnement normal. Le problème du diagnostic consiste alors à déterminer les composants dont le dysfonctionnement expliquerait les différences observées. Des extensions ont été proposées pour permettre d'intégrer et d'exprimer des modes de dysfonctionnement (Kleer et Williams, 1989). Ceci permettant de mettre en place des raisonnements abductifs dont le but est d'exprimer les causes plus précises du conflit.

Le modèle en diagnostic logique 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 inclut un prédicat noté AN (signifiant ANormal). La description structurelle quant à elle décrit les liens entre les composants.

5.2 Avantages/Inconvénients

En sus de la nécessité de ne modéliser que le fonctionnement normal du système, vient s'ajouter la généricité du raisonnement. Celui-ci n'est pas dépendant du type de système étudié. L'évolutivité du système est également un point fort des approches à base de diagnostic logique. Une simple mise à jour du modèle est requise lors de l'ajout, la modification ou la suppression de composants. En dernier lieu nous soulignerons la facilité d'acquisition du modèle compte tenu de la description modulaire proposée.

Un inconvénient majeur du diagnostic logique est le temps de calcul nécessaire pour effectuer des raisonnements. En effet afin de déterminer les diagnostics, une phase préliminaire consiste à déterminer les conflits minimaux, correspondant à un problème NP-difficile (cf. chapitre 1 (Dubuisson, 2001)). Un deuxième point faible du diagnostic logique est à retenir : le comportement du système doit pouvoir être obtenu à partir des comportements indépendants de ses composants et de la structure du système. En d'autres mots, le mauvais fonctionnement d'un composant n'a aucune conséquence sur le fonctionnement des autres composants.

6 Diagnostic de plan

Des travaux sur le diagnostic ont été également conduits dans le domaine de la génération de plans en intelligence artificielle (planification automatique) (Witteveen et al., 2005). L'objectif du diagnostic de plan est de déterminer les actions potentiellement mal exécutées d'un plan à partir d'au moins deux observations de l'état du système. La méthode développée s'appuie sur les principes du diagnostic logique. Si une action du plan est vue comme un composant et le plan comme un système, la formulation du problème de diagnostic peut être ramenée à un problème de diagnostic logique. Toutefois, les actions sont considérées comme indépendantes les unes des autres. La post-condition (sortie composant) d'une action étant considérée comme

réalisée ou inconnue, et les seules conditions à l'exécution d'une action étant la réalisation de ces pré-conditions (entrées composants) et l'absence de dysfonctionnement de l'action (dysfonctionnement composant).

Dans un deuxième temps, (de Jonge et al., 2006) propose un diagnostic des causes du dysfonctionnement de l'action (produite par la phase précédente) en se basant sur les techniques utilisées dans le diagnostic des systèmes à événements discrets. Ceci passant par une modélisation des différentes causes possibles et leurs conséquences sur la post-condition de l'action.

Nous ne reprendrons pas ici les avantages et inconvénients du diagnostic de plan, car celui-ci se base sur des méthodes présentées précédemment.

7 Positionnement des travaux

7.1 Positionnement de la modélisation et des principes du diagnostic proposé

Il est proposé dans cette section de faire un bilan des différentes approches de diagnostic dans le but de positionner l'approche proposée dans ce manuscrit. Rappelons que la problématique générale à laquelle nous tendons à apporter une solution est la reconfiguration du système de commande. Dans cet objectif et au niveau coordination auquel ces travaux se placent, il est nécessaire de diagnostiquer la disponibilité des services, l'état de la partie opérative et celui du flux de produits. Ce processus devra être en mesure de faire face à la complexité induite à ce niveau (cf. section 5.2.1 du chapitre 2). Une des premières questions à se poser est la nécessité ou non d'adopter une modélisation des défaillances possibles. Dans le cas d'un raisonnement abductif, où les origines précises qui expliquent le symptôme d'une défaillance sont recherchées, il peut être intéressant d'avoir recours à un modèle de comportement anormal du système étudié. En effet, en l'absence de celui-ci, le pouvoir explicatif est nul et uniquement fondé sur la restauration de la cohérence avec les observations émanant du système (Pencolé, 2002). Ceci correspond à notre problématique, dans le sens où les origines exactes de la réception d'un CRA ne sont pas requises. Il est uniquement nécessaire de connaître les services mal exécutés menant à un conflit entre les observations et l'état attendu du système. Il serait donc tout a fait raisonnable, d'adopter la philosophie des travaux réalisés dans le diagnostic de plan. C'est à dire considérer les services exécutés comme des composants, et l'ensemble des services exécutés comme le système. A partir d'une telle modélisation la recherche des conflits permettrait de mettre en évidence les services potentiellement mal exécutés, correspondant aux défaillances premières potentielles.

Ceci ne permet pas de répondre totalement à notre problématique. Suite à la mise en évidence des défaillances premières potentielles, il est nécessaire d'étudier la conséquence de leurs propagations sur les autres services exécutés (cf. section 3.3 du chapitre 2). Dans les approches logiques présentées dans la littérature, il n'est pas possible dans le cadre de la modélisation proposée et des algorithmes utilisés de considérer l'impact des entrées du composant sur la nature de son fonctionnement, normal ou anormal. En sus de ce problème, les composants sont considérés indépendants dans les approches de diagnostic logique. Toutefois, les services sont bien entendu dépendants des chaînes fonctionnelles, et donc le diagnostic proposé

ici ne peut plus faire l'hypothèse que le mauvais fonctionnement d'un composant n'impacte pas le fonctionnement des autres composants. L'approche logique ou même l'extension qui en est faite pour le diagnostic de plan n'est donc pas adaptée entièrement à ce problème. Il sera donc nécessaire dans nos travaux de proposer une extension de la modélisation faite dans le cadre du diagnostic logique afin de pouvoir étudier et prendre en compte les particularités de notre problématique. Cette extension devra permettre de décrire le lien entre le fonctionnement des composants, mais également l'impact des entrées d'un composant sur son fonctionnement.

Si nous considérons le principe précédemment développé, le modèle utilisé pour le diagnostic sera en quelque sorte une image du fonctionnement passé du système. Ceci induit qu'il sera nécessaire de limiter celui-ci afin de ne pas se retrouver dans la situation d'une explosion de la taille du modèle. Cette problématique n'a bien entendu pas été abordée dans le diagnostic logique dans le sens où les modèles qui représentent un système est composé d'un nombre fini de composants. En ce qui concerne le diagnostic de plan, cette limitation découle naturellement du fait qu'un plan n'est exécuté qu'une seule fois. Toutefois, la commande des SAP se différencie de la planification par la nature cyclique des lois de commande, impliquant un fonctionnement ininterrompu de la partie opérative. Il serait "malhonnête" de limiter la recherche du diagnostic à un simple critère temporel. Cette limitation sera effectuée comme développé dans la section 5.2.1 du chapitre 2 sur l'exploitation de la confiance ou non accordée aux chaînes fonctionnelles quant à l'exécution des services quelles proposent.

7.2 Positionnement algorithmique

L'utilisation des systèmes experts est difficilement envisageable au niveau coordination des chaînes fonctionnelles. Il serait utopique de demander à l'expert de recenser l'ensemble des défaillances possibles ainsi que leurs conséquences en cas de propagation, la combinatoire des cas étant difficilement appréhendable. Cependant nous porterons une attention toute particulière au mécanisme utilisé. En effet ce mécanisme s'appuie sur un parcours de règles afin de trouver les causes d'un symptôme détecté ; ce mécanisme ayant un intérêt certain d'un point de vue temps de calcul et convergence de l'algorithme de diagnostic. Notre proposition quant au diagnostic des services tentera d'exploiter le même principe dans la recherche des services potentiellement à l'origine de la réception d'un CRA, ainsi que dans la déduction des services affectés par ces défaillances premières. Ces recherches seront basées sur des règles liées à la nature même des services. Elles se limiteront à une interprétation de leur fonctionnement normal. Ces règles seront basées sur le principe suivant : si un service n'est pas exécuté dans les conditions normales de son utilisation, il en résulte une mauvaise exécution. Ce principe sera développé dans la suite du manuscrit.

8 Conclusion

Tout au long de ce chapitre nous avons réalisé un tour d'horizon des approches qui apportent leur contribution à la problématique du diagnostic. Nous retenons en particulier la réactivité des systèmes expert, des approches de reconnaissance de scénario et de diagnostic de SED. Cependant, ces trois approches se basent sur une connaissance du fonctionnement anormal du système à diagnostiquer, connaissance dont les approches soulignent la difficulté d'acquisition.

D'un autre côté, le diagnostic logique peut être basé sur un modèle exclusif du comportement normal du système à diagnostiquer. Cependant cette dernière approche souffre du temps de calcul des algorithmes utilisés.

Au delà des avantages et inconvénients présentés pour chacune de ces approches nous reprendrons à notre compte les principes de l'approche de type diagnostic logique qui se prête le mieux à nos besoins. Elle sera cependant à étendre afin de pouvoir tenir compte de trois spécificités majeures : l'effet d'un service dépend certes de son contexte d'exécution mais ce dernier impacte également le bon fonctionnement de la chaîne fonctionnelle l'exécutant. En deuxième lieu, l'exécution non conforme d'un service peut impacter l'ensemble des services de la même chaîne fonctionnelle. Enfin, la taille du modèle pour le diagnostic doit être limitée.
Forts de ces principes, la démarche qui sera proposée dans la suite de ce manuscrit suivra la trame suivante :

- soucieux de l'intégrabilité des travaux dans un contexte industriel, une première phase consistera à proposer une méthode permettant de capitaliser toutes les informations nécessaires à la fonction diagnostic,

- ensuite il sera proposé un mécanisme permettant non seulement de générer le modèle pour le diagnostic correspondant au fonctionnement passé du système mais également d'en gérer sa taille via un mécanisme de réduction.

- un mécanisme de diagnostic, exploitant le modèle précédemment généré, capable de trouver les origines d'un CRA, et leurs conséquences sur les services exécutés,

- un mécanisme de mise à jour, capable de projeter le résultat de diagnostic sur le modèle des capacités de la partie opérative.

Deuxième partie

Vers un modèle pour le diagnostic

Chapitre 4

Acquisition des connaissances sur les capacités de la partie opérative

1 Introduction

L'approche de diagnostic que nous proposons se singularise en abordant globalement la problématique de l'acquisition de la connaissance jusqu'à son exploitation.

En premier lieu nous présenterons la démarche d'acquisition de la connaissance que nous avons adoptée et étendue à nos besoins. Cette dernière s'appuie sur des techniques de modélisation issue de la génération de plans en Intelligence Artificielle. Après quoi nous présenterons dans le détail le modèle d'une opération d'action. Sur la base de ce modèle d'opération, trois autres seront ensuite présentées. Une telle classification sera considérée comme une des clefs des techniques de diagnostic que nous proposons dans la suite de ce document.

2 Démarche de description des capacités opératoires

L'objectif de la fonction diagnostic proposée est résumé dans la Figure 2.7 page 37. Afin de pouvoir remplir ces objectifs, la méthode de diagnostic est basée sur un ensemble d'informations comme les capacités de la partie opérative ou le fonctionnement passé du système. Ce chapitre se focalise sur la description des capacités de la partie opérative.

2.1 Choix de l'outil

La description des capacités de la partie opérative, indispensable au diagnostic, revient à mettre en exergue un ensemble d'informations qu'il est nécessaire d'acquérir auprès d'un expert. La section 4.2 du chapitre 1 a souligné la difficulté d'acquisition de cette connaissance. Afin de permettre une intégration à terme de nos travaux dans un contexte industriel, il est important de proposer à l'expert une démarche structurante afin de le guider dans la description des capacités de la partie opérative. En terme d'expression des capacités offertes par une partie opérative, le modèle d'opération, issu de la génération de plans en intelligence artificielle (planification automatique) (Ghallab et al., 2004) se révèle être des plus appropriés (Henry, 2005), même si comme nous le verrons plus loin, des extensions doivent être proposées pour les besoins du diagnostic. Notons cependant que le but des travaux présentés dans ce manuscrit n'est pas

de se focaliser sur ce premier outil, mais simplement d'en exploiter les possibilités en terme d'acquisition de la connaissance requise au diagnostic.

2.2 Modèles d'opérations

Les modèles d'opérations présentés dans cette section reprennent les concepts proposés dans le cadre de la synthèse de lois de commande (Henry, 2005). Un modèle d'opération se présente comme illustré dans la Figure 4.1.

- Il se compose d'un champ *évolution de la chaîne fonctionnelle* décrivant l'effet du lancement du service sur la chaîne fonctionnelle. L'obtention de cet effet est soumis à la satisfaction de pré-requis qui seront définis en détail dans la prochaine section.

- L'effet de la chaîne fonctionnelle a potentiellement pour vocation de modifier l'état d'un ou plusieurs produits. Ceci se traduit par la description d'un ou plusieurs champs *évolutions du flux de produits*. Chacune de ces évolutions est décrite d'une part par un effet sur l'état d'un produit, et d'autre part par les pré-requis à satisfaire pour obtenir cet effet.

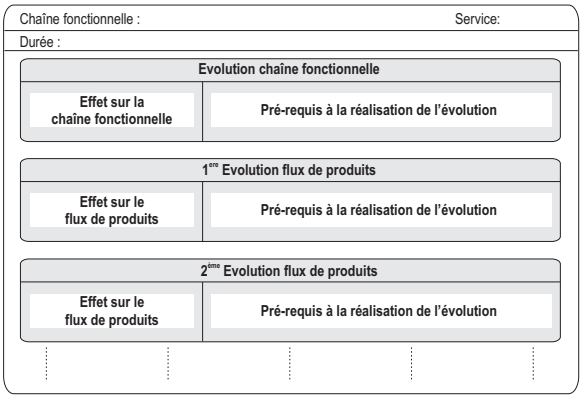


FIG. 4.1 – Structuration d'une opération

L'ensemble des opérations associées au sous-système piloté permet donc de décrire les capacités opératoires offertes par les chaînes fonctionnelles. Les différents champs proposés par la Figure 4.1 portent sur l'état des chaînes fonctionnelles et du flux de produits.

2.3 État des chaînes fonctionnelles et du flux de produits

L'état de la partie opérative est caractérisé par l'état de chacune des chaînes d'action et d'acquisition composant les chaînes fonctionnelles. Il caractérisera par exemple des vitesses de déplacement, les positions des effecteurs pour des chaînes d'action... L'état du flux de produits est caractérisé par l'état de chacun des produits se trouvant dans le sous-système piloté par le module de coordination. Il représentera les références des produits, leurs positions ainsi que l'ensemble de leurs caractéristiques physiques (couleur, forme, présence d'un alésage...)

L'état des chaînes fonctionnelles est modélisé par un ensemble de variables d'état. Les variables d'état seront notées : $Nom\_CF.Nom\_Variable\_d'état$. La description des opéra-

tions se fait donc soit sur des propositions logiques portant sur ces variables pour les prérequis, soit sur l'effet des opérations sur ces variables. Il est à noter une spécificité quant au flux de produits. Un produit est une entité qui traverse le système piloté. Il passera par un ensemble de positions pour lesquelles il est nécessaire de connaître sa présence ou non. De surcroît les effets des opérations seront exprimés sur les variables d'état d'un produit à une position donnée. La description des opérations fera donc également référence à la position des produits : $ProduitEnPositionY.Nom\_Variable\_d'état$. Le changement de position d'un produit sera modélisé quant à lui au travers de l'opérateur "→" pour l'expression $ProduitEnPosition1 \rightarrow ProduitEnPosition2$. Cet opérateur signifie que l'objet produit (ainsi que toutes ses variables d'état associées) passe de la position 1 à la position 2. La présence d'un produit pourra ainsi être testée directement sur une position en utilisant la notation suivante : $ProduitEnPositionY == vide$ (*ou non vide*). Afin de pouvoir différencier les différents produits présents dans le sous-système piloté, un identifiant leur sera associé. Cette identifiant sera exploité par le diagnostic afin de retracer les différentes évolutions qu'un produit a subi. Ainsi, lors de l'exploitation des informations du modèle d'opération, la modification d'une variable d'état d'un produit à une position ($ProduitEnPositionX.Nom\_Variable\_d'état = valeur$) sera traduite pour faire référence à l'identifiant du produit ($ProduitIdentifiant.Nom\_Variable\_d'état = valeur$).

2.4 Les différentes catégories d'opérations

Les travaux développés dans (Henry, 2005) ont mis en exergue une classification des capacités offertes par la partie opérative en plusieurs catégories d'opérations : opération d'action, induites, requises et d'acquisition. Ces distinctions, comme nous le verrons plus loin, sont essentielles pour le diagnostic. Les opérations induites, requises et d'acquisition n'étant que des cas particuliers de l'opération d'action, les sections suivantes présentent en détail l'opération d'action puis la spécificité des trois autres.

3 Description d'un service par une opération d'action

3.1 Définition

Une opération d'action est basée sur la réalisation d'un service qui a pour effet une modification de l'état de la chaîne fonctionnelle. Cette évolution entraîne en fonction de l'état du flux de produits la modification de l'état d'un ou plusieurs produits. Le déclenchement d'une opération d'action se fait par l'envoi d'une requête à un service à laquelle correspond un compte rendu d'exécution renvoyé par la chaîne fonctionnelle offrant le service.

3.2 Effet des services

3.2.1 Effet sur les chaînes fonctionnelles

La modification de l'état d'une chaîne fonctionnelle se traduit par une évolution de ses variables d'état. Cette modification peut être décomposée en deux phases. La première revient à placer la chaîne fonctionnelle dans l'état correspondant à l'exécution du service. La deuxième correspond au passage de l'état de la chaîne fonctionnelle vers son état final. Ces deux phases sont respectivement appelées *effet transitoire* et *effet final* de l'évolution de la chaîne fonctionnelle (Henry et al., 2005). L'évolution de l'état d'une chaîne fonctionnelle peut être soumise à

la satisfaction de pré-requis. Nous appellerons *pré-condition* l'ensemble des valeurs des variables d'état caractérisant l'état des chaînes fonctionnelles et du flux de produits à satisfaire pour obtenir l'effet transitoire. La *condition* correspond aux valeurs des variables d'état à satisfaire durant toute l'exécution du service pour obtenir l'effet final. Afin d'illustrer ces principes la Figure 4.2 propose une description de l'évolution de la chaîne fonctionnelle d'un service *sortir vérin*. Les pré-conditions à satisfaire correspondent d'une part aux valeurs de la variable d'état *position* du vérin et d'autre part aux variables d'état de l'environnement à satisfaire afin d'obtenir l'effet transitoire. La condition correspond uniquement aux variables d'état de l'environnement puisque la valeur de la variable d'état *position* est imposée par l'exécution du service.

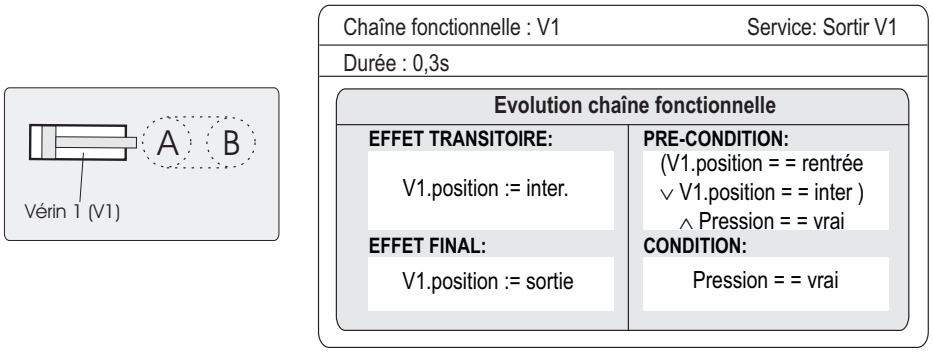


Fig. 4.2 – Évolution de la chaîne fonctionnelle

La prise en compte des *pré-conditions* et *conditions* doit permettre à la fonction diagnostic d'émettre des hypothèses sur les origines potentielles de la mauvaise exécution d'un service. Les origines s'expriment en termes de non compatibilité de l'état des chaînes fonctionnelles et du flux de produits pour le lancement du service (cf. Figure 4.3). La connaissance de l'effet des services sur les chaînes fonctionnelles permet cette fois-ci d'émettre des hypothèses quant aux mauvaises exécutions de services à l'origine possible de cette non conformité de l'état (cf. Figure 4.3), ceci correspondant à une propagation de défaillances via la partie opérative (cf. section 3.2 du chapitre 2). Une propagation de défaillances peut également se faire à travers le flux de produits. Ceci nous amène à définir le comportement d'un service sur le flux de produits.

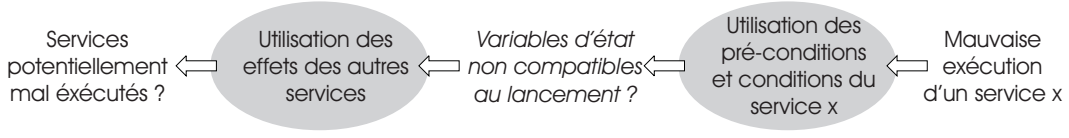


Fig. 4.3 – Utilisation de l'information sur le comportement par le diagnostic

3.2.2 Effets sur le flux de produits

Une opération décrit l'évolution d'une chaîne fonctionnelle mais également un ensemble d'évolutions possibles du flux de produits. En fonction de l'état du flux de produits au moment du

lancement d'un service, ce dernier pourra avoir plusieurs comportements différents correspondant à des évolutions différentes du flux de produits. La réalisation de l'effet transitoire d'une évolution du flux de produits dépendra de la satisfaction de la pré-condition et l'effet final de l'évolution dépendra de la satisfaction de la condition. La Figure 4.4 présente l'ensemble de ces informations sur l'exemple du service *sortir V1*.

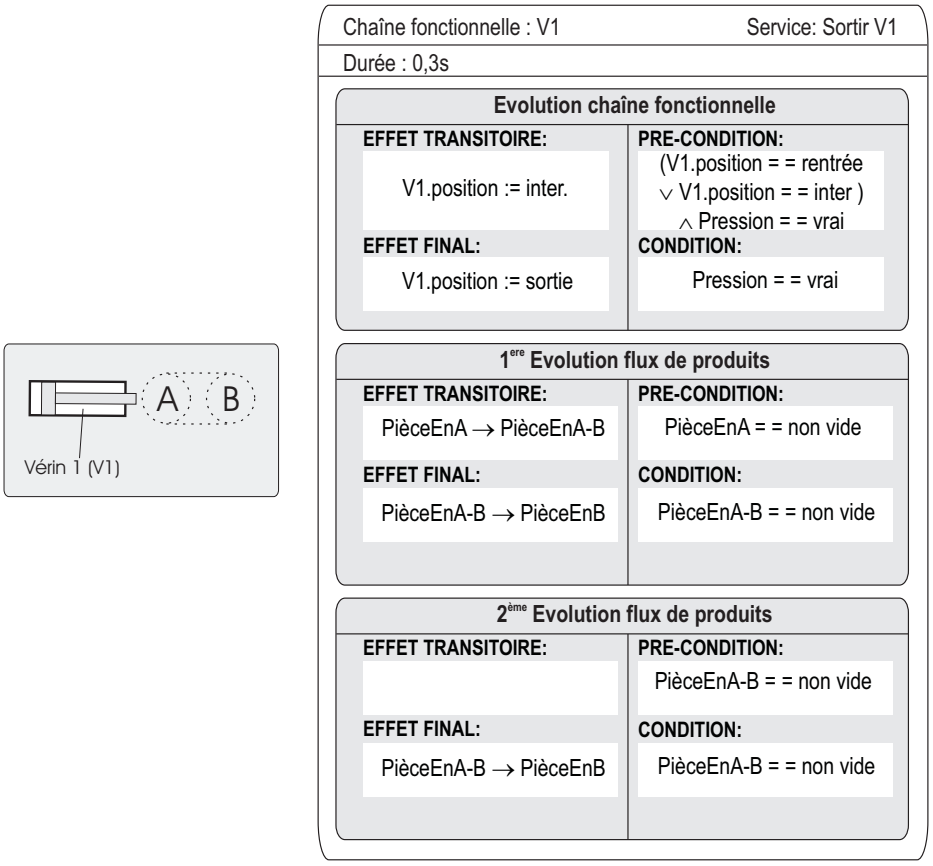
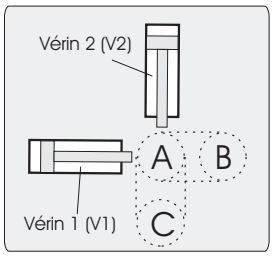


FIG. 4.4 – Évolutions du flux de produits du service *sortir V1*

3.3 Contraintes de sécurité

La section précédente a défini les différentes évolutions pouvant être produites par un service. Si les évolutions réalisées par le lancement d'un service ne sont pas conformes avec celles qui étaient attendues, alors il est possible de mettre en évidence les origines possibles en terme d'états non compatibles des chaînes fonctionnelles et du flux de produits (pré-conditions et conditions satisfaites différentes de celles attendues). Toutefois la non conformité du comportement d'un service peut également être la conséquence du non respect de contraintes de sécurité (cf. section 3.3 du chapitre 2). La connaissance des pré-contraintes à respecter avant le lancement du service et des contraintes à respecter durant sa réalisation est donc nécessaire. Ces pré-contraintes et contraintes seront associées à l'évolution de la chaîne fonctionnelle et aux différentes évolutions possibles du flux de produits. La Figure 4.5 reprend l'exemple du service *sortir V1* via l'ajout d'un deuxième vérin orthogonal. Dans cet exemple, la contrainte portant sur l'absence d'un produit entre la position A et C ainsi que sur la position du deuxième vérin sont relatives à

l'évolution de la chaîne fonctionnelle. Elles doivent être respectées quelles que soient les évolutions associées du flux de produits afin d'éviter une collision entre la tige du vérin et un éventuel produit. Cependant, la pré-contrainte "absence d'un produit à la position B" doit être respectée uniquement pour assurer l'évolution du flux de produits ; une collision entre deux produits doit en effet être interdite.



Chaîne fonctionnelle : V1 — Service: Sortir V1

Durée : 0,3s

Evolution chaîne fonctionnelle

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
V1.position := inter.	(V1.position = = rentrée ∨ V1.position = = inter) ∧ Pression = = vrai	PièceEnA-C = = vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
V1.position := sortie	Pression = = vrai	V2.position = = rentrée ∧ PièceEnA-C = = vide

1ere Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
PièceEnA → PièceEnA-B	PièceEnA = = non vide	PièceEnA-B = = vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
PièceEnA-B → PièceEnB	PièceEnA-B = = non vide	PièceEnB = = vide

2ème Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
	PièceEnA-B = = non vide	
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
PièceEnA-B → PièceEnB	PièceEnA-B = = non vide	PièceEnB = = vide

FIG. 4.5 – Description complète du service *sortir V1*

3.4 Extension du concept d'opération pour le diagnostic

La description ainsi proposée des services offerts par les chaînes fonctionnelles est une extension de celle proposée dans (Henry, 2005) pour les besoins du diagnostic. Cette extension a consisté à distinguer les pré-conditions des pré-contraintes, et les conditions des contraintes. Cette distinction n'était pas nécessaire dans (Henry, 2005) pour la synthèse de loi de commande dans le sens où lors d'un appel à un service, les (pré-)conditions, (pré-)contraintes devaient être respectées conjointement. Cependant dans le cadre du diagnostic, leur distinction est essentielle. Le non respect d'une pré-condition par exemple, entraîne la non réalisation de l'effet correspondant. En revanche, les conséquences du non respect d'une pré-contrainte sont plus difficilement prévisibles comme nous le verrons plus loin dans ce mémoire.

4 Les autres catégories d'opérations

4.1 Opération induite

4.1.1 Spécificités

Une opération induite correspond à une modification de l'état d'un ou plusieurs produits depuis un état, dit instable, des produits. Cet état instable est atteint suite à l'exécution d'opérations d'action. N'étant pas basée sur la modification de l'état d'une chaîne d'action, une opération induite n'est pas déclenchée par une requête d'appel à un service mais par la satisfaction d'au moins une pré-conditions des évolutions du flux de produits. Ainsi, une opération induite, ne comportera pas de champ évolution de la chaîne fonctionnelle.

4.1.2 Intérêt pour le diagnostic

Dans le cadre du diagnostic, la spécificité des opérations induites permet de prendre en compte une difficulté liée à la nature même des systèmes étudiés, des évolutions du produit qui ne sont pas prévisibles. Suite à une propagation de défaillance, l'état réel des chaînes fonctionnelles et du flux de produits peut être différent de l'état attendu. La satisfaction non prévue de pré-conditions peut entraîner ainsi l'évolution du flux de produits. Aussi, la recherche par le diagnostic des opérations induites intempestivement exécutées lors d'une propagation de défaillance doit permettre de retrouver les évolutions du flux de produits.

4.1.3 Prise en compte dans l'approche proposée

Une loi de commande au niveau coordination permet par définition le lancement d'un ensemble de services (envoi de requêtes et attente de compte-rendus). Ainsi, une opération induite n'étant pas déclenchée par l'appel à un service, les lois de commande ne contiennent pas les informations nécessaires pour connaître l'exécution de telles opérations prévues en fonctionnement normal.

Il a ainsi été proposé dans (Mauser, 2006) d'incorporer les opérations induites (i.e. évolution du flux de produits non contrôlé par le lancement d'un service) dans la loi de commande, afin de les mettre aux services de la fonction diagnostic. Pour cela, même si une requête et un compte rendu ne sont pas réellement échangés avec une chaîne fonctionnelle pour l'exécution d'une opération induite, les travaux présentés dans (Mauser, 2006) ont proposé que la loi de commande contienne tout de même des requêtes et des comptes rendus "fictifs" pour ces opérations induites. Ceux-ci pouvant être ainsi exploités par le diagnostic.

4.2 Opération d'acquisition

Une opération d'acquisition représente un service offert par une chaîne fonctionnelle qui est composée uniquement d'une chaîne d'acquisition. La réalisation du service provoque une modification des variables d'état du flux de produits pour le niveau coordination. D'un point de vue diagnostic, le traitement d'une telle opération sera strictement identique à celui d'une opération d'action ayant un effet sur le flux de produits.

4.3 Opérations requises

4.3.1 Spécificités

Une opération requise correspond à une évolution de l'état d'une chaîne fonctionnelle de l'environnement et/ou une évolution de l'état du flux de produits. Les opérations requises représentent les informations indispensables qu'un module coordination doit connaître de son environnement (da Silveira, 2003). En effet, compte tenu des interactions entre le système piloté par un module de coordination et son environnement, la réalisation de services offerts par les chaînes fonctionnelles pilotées est contrainte par l'état de l'environnement et des produits sur lesquels elles agissent. Une opération requise peut correspondre, à la description de l'introduction d'un produit à l'entrée du système piloté par le module de coordination considéré.

4.3.2 Intérêt pour le diagnostic

Il est important pour la fonction diagnostic de distinguer ce type d'opération. Une opération requise correspond à un service offert à un autre module de coordination que celui considéré ou à une action de l'opérateur. Lors de la recherche des origines de la réception d'un CRA, si le diagnostic remet en cause la réalisation d'une opération requise, il est dans l'incapacité d'en étudier toutes les origines possibles, car il n'en a qu'une description partielle. Seul le module de coordination qui a lancé cette opération a la connaissance de l'ensemble des pré-requis nécessaires à son exécution. Comme nous le verrons dans les perspectives de ces travaux, nous atteignons ici les limites de l'utilisation d'une structure de pilotage hiérarchisée dans un contexte de fonctionnement anormal.

4.3.3 Prise en compte dans l'approche proposée

Pour un module de coordination considéré, la connaissance des évolutions associées à l'exécution d'une opération requise gérée par un autre module de coordination est nécessaire. Ainsi, il nous parait raisonnable de prendre comme hypothèse que tout module de coordination propriétaire d'opérations requises pour d'autre, propage le début et la fin de cette opération aux modules concernés. Aussi, la remise en cause de l'exécution d'une opération requise devient une information capitale pour le module de coordination ayant lancé cette opération mais également pour tous les autres modules de coordination pour lesquels cette opération est requise. Ainsi, suite à une telle remise en cause par un module de coordination, ce module transmettra cette remise en cause accompagnée de la date à laquelle cette opération a été exécutée. Ces informations permettront à chacun des modules en question de poursuivre l'analyse diagnostique.

5 Conclusion

Ce chapitre s'est attaché à proposer une démarche structurante pour l'expert afin de l'aider dans la phase d'acquisition de la connaissance requise pour les besoins du diagnostic. Dans ce sens, nous nous sommes attachés à présenter les caractéristiques fondamentales des modèles d'opérations utilisés et offerts à l'expert. Ces dernières ont été mises en avant selon deux axes majeurs ; d'une part une classification en quatre catégories d'opérations (les opérations d'action, les opérations induites, les opérations d'acquisition et enfin les opérations requises)

permettant d'orienter le diagnostic dans ses phases de recherche origines/conséquences d'un dysfonctionnement, d'autre part une structuration sur l'expression des pré-contraintes, contraintes, pré-conditions et conditions qui ferme le contexte de l'évolution d'une chaîne fonctionnelle et du flux de produits, et donc de leurs effets.

Les caractéristiques fondamentales des modèles d'opérations étant désormais avancées, nous nous proposons maintenant de découvrir un autre point clef de notre approche sur lequel le mécanisme d'oubli, présenté dans la suite du manuscrit, s'appuie.

Chapitre 5

Prise en compte de l'observabilité pour le diagnostic

1 Introduction

Comme nous l'avons souligné dans le chapitre 3 de la partie précédente, l'approche proposée dans ce manuscrit est en partie fondée sur les principes du diagnostic de plans. Elle s'appuiera donc sur l'exploitation d'un modèle du fonctionnement passé du sous-système piloté par le niveau coordination. Afin d'anticiper le problème classique de la maîtrise de la taille de ce modèle, un mécanisme d'oubli efficace doit être mis en place.

Ce chapitre propose dans ce sens de poser les hypothèses que nous avons retenues quant à la notion d'observabilité des chaînes fonctionnelles. Le premier paragraphe est dédié à l'introduction du concept d'indice de confiance sur lequel notre mécanisme d'oubli est basé. Après quoi nous proposons une extension du modèle d'opération à cet indice. Enfin la dernière section de ce paragraphe présente le cas particulier des opérations de surveillance chargées d'observer l'évolution du flux de produits.

2 Objectif de l'exploitation de l'observabilité

Le mécanisme d'oubli et de réduction du modèle pour le diagnostic ne peut pas se baser uniquement sur l'observabilité offerte par une chaîne d'acquisition sur le flux de produits et sur la partie opérative. En effet, ceci reviendrait à émettre l'hypothèse qu'une chaîne d'acquisition est exempte de défaillance. Ceci est irréaliste dans un contexte industriel, dans le sens où de nombreuses défaillances sont dues aux capteurs (Sourisse et Boudillon, 1997). Nous avons donc basé le mécanisme d'oubli et de réduction du modèle sur un *indice de confiance* associé aux effets observés des opérations. Cet *indice de confiance* dépend de l'observabilité des variables d'état sur lesquelles l'opération a un effet mais également sur une connaissance a priori du comportement de la variable d'état observée.

Afin d'illustrer ce principe, nous proposons au lecteur de nous accompagner dans cette section au cœur même du fonctionnement d'un module de contrôle commande d'une chaîne fonctionnelle. Aussi, prenons l'exemple d'un vérin initialement en position rentrée et équipé de deux capteurs fin

de course. Considérons que le module de contrôle/commande de la chaîne fonctionnelle possède un modèle temporel du vérin. La Figure 5.1 présente ce modèle ainsi que la position réelle du vérin et les informations capteurs suite à l'ordre du module de contrôle/commande de la chaîne fonctionnelle de la sortie du vérin.

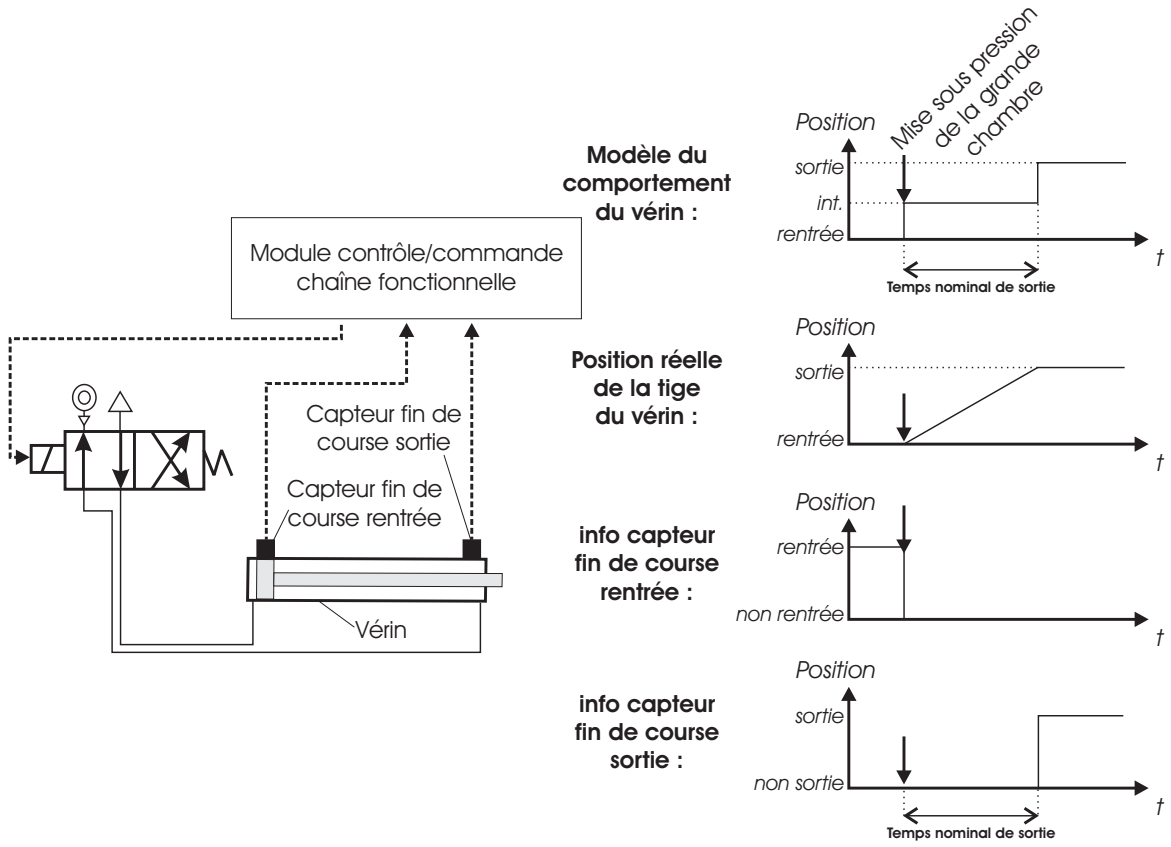


Fig. 5.1 – Détermination de l'indice de confiance

L'exploitation de l'observabilité sur la position de la tige du vérin se fait de la manière suivante :

- S'il y a correspondance entre l'instant de la demande de mise sous pression de la chambre gauche du vérin et l'information issue du capteur fin de course rentrée (indiquant que le vérin a quitté la position rentrée) : une confiance quant au passage de la position rentrée à la position intermédiaire peut être accordée par le module de contrôle/commande de la chaîne fonctionnelle.

- Si l'information du capteur fin de course sortie correspond au temps de sortie nominal du vérin alors une confiance quant au passage de la position rentrée à la position intermédiaire à la position peut être accordée par le module de contrôle/commande de la chaîne fonctionnelle.

Cependant nous devrons noter que la confiance accordée à ces changements d'état dé-

pend de la position des capteurs sur la chaîne d'action. De plus une défaillance pourrait survenir au niveau du capteur et entraîner un changement d'état à l'instant même où il est attendu. Cependant, la probabilité qu'une défaillance survienne et entraîne un changement d'état du capteur à l'instant même où il est attendu et qu'une deuxième défaillance ou propagation de défaillance entraîne que ce changement d'état ne soit pas réalisé est très faible.

Forts de ces hypothèses, il est nécessaire de proposer maintenant un mécanisme permettant de divulguer cette confiance accordée à l'effet sur la chaîne d'action au niveau coordination. Nous proposons ici de définir un *indice de confiance* statique qui sera dans le cadre de ce mémoire, binaire.

3 Prise en compte pour le diagnostic

Il est ainsi proposé dans cette section de formaliser cet indice de confiance que nous noterons désormais IC :

- *L'indice de confiance IC(effet sur chaîne fonctionnelle) est égal à une valeur binaire* 1 *s'il y a correspondance entre l'évolution observée de la variable d'état de la chaîne fonctionnelle et une connaissance de son comportement,* 0 *sinon.*

- *Si l'indice de confiance IC(effet sur chaîne fonctionnelle) est égal à* 1*, le diagnostic qualifiera de correctes la valeur donnée à la variable d'état correspondante, et ne la remettra pas en cause dans son analyse.*

Cet indice de confiance offert au niveau coordination quant à la bonne issue des effets de l'évolution d'une chaîne fonctionnelle, est par nature statique puisque la structure d'une chaîne fonctionnelle n'évolue pas au cours du temps. Il fait donc partie des caractéristiques intrinsèques des services (cf. Figure 5.2).

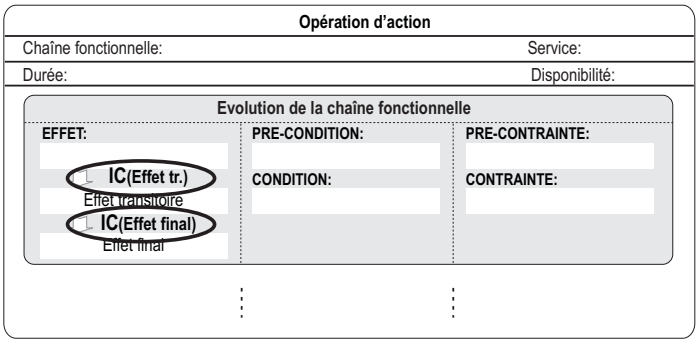


FIG. 5.2 – Description d'une opération d'action avec l'indice de confiance

4 Exploitation de l'observabilité sur le flux de produits

4.1 Principe de l'exploitation de l'observabilité

La prise en compte de l'observabilité des chaînes d'acquisition portant sur le flux de produits est un problème plus complexe. Elle ne peut pas être traitée de la même manière que pour les chaînes d'action, ne serait ce que par la méconnaissance des modules de contrôle/commande sur le comportement du flux de produits. Reprenons l'exemple précédent du vérin : lors d'une requête au service *sortir V1*, la présence ou non d'un produit devant le vérin est une information propre au niveau coordination. Il est donc nécessaire de mettre en place un mécanisme spécifique pour la surveillance du flux de produits. Ce mécanisme est basé sur le lancement d'opérations de surveillance que nous nous proposons de découvrir maintenant.

4.2 Les opérations de surveillance

Les opérations d'acquisition proposées à la section 4.2 du chapitre 4, permettent d'obtenir une information initialement inconnue sur les caractéristiques (spaciales ou physiques) d'un produit ; ce qui d'un point de vue diagnostic est vue comme une modification des valeurs des variables d'état du flux de produits. Toutefois, les services d'une chaîne d'acquisition permettant d'obtenir de l'information sur des variables d'état dont le changement est prévu par le niveau coordination, ne peuvent pas être décrits par une opération d'acquisition. En effet, ces opérations décrivent l'acquisition d'informations sur un produit comme des effets sur ses variables d'état. Nous introduisons donc une nouvelle catégorie d'opération permettant de décrire ces services de surveillance.

4.3 Description d'un service de surveillance

Une opération de surveillance permet de décrire un service de surveillance d'une variable d'état. Elle est composée d'un champ évolution de la chaîne fonctionnelle comme une opération d'acquisition et d'un ou plusieurs *comportements de surveillance*. En effet, les pré-requis à respecter peuvent dépendre de la valeur à surveiller. Chacun de ces comportements de surveillance décrit (cf. Figure 5.3) :

- la pré-condition, la condition, la pré-contrainte et la contrainte à respecter durant la surveillance,

- les valeurs de la variable d'état qui peuvent être surveillées.

4.4 Exploitation des opérations de surveillance

Les opérations de surveillance doivent permettre de transmettre le comportement d'un effet sur une variable d'état à un module de contrôle/commande d'une chaîne fonctionnelle. Ce comportement sera décrit par la valeur que doit prendre la variable d'état, et par le temps au bout duquel cette valeur doit être prise par la variable d'état. La transmission de ces informations doit permettre au module de contrôle/commande de mettre en place un raisonnement tout à fait

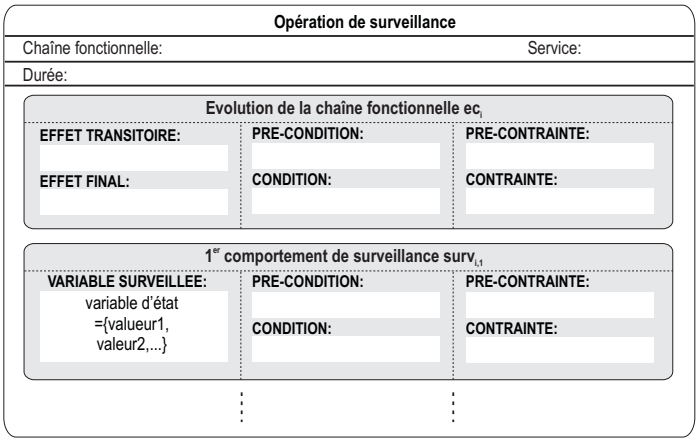


FIG. 5.3 – Opération de surveillance

similaire à celui exposé dans la section 2. Il a été proposé dans (Mauser, 2006), une extension de l'algorithme synthèse de lois de commande afin d'intégrer les opérations de surveillance en leur sein. Les lois de commande et de surveillance ainsi générées par cet algorithme étendu permettent le lancement des opérations de surveillance tout au long de l'exécution de cette loi. Dans un soucis de concision, ces travaux ne seront pas présentés plus en détail ici, mais le lecteur pourra se référer à l'annexe B où une présentation synthétique est proposée.

4.5 Exploitation des comptes-rendus de l'exécution d'une opération de surveillance

Le rôle même d'une opération de surveillance étant de permettre aux chaînes fonctionnelles de corréler une observation du flux de produits avec une connaissance du comportement de celui-ci, l'indice de confiance accordé à la valeur surveillée sera de 1. Dans le cas où un compte rendu d'exécution anormal d'une opération de surveillance est reçu par le niveau coordination, traduisant une incohérence entre la connaissance du comportement du flux de produits et son observation, la fonction diagnostic devra alors en rechercher toutes les origines possibles sans oublier une défaillance de la chaîne fonctionnelle ayant exécuté l'opération de surveillance.

5 Conclusion

Ce chapitre s'est attaché à poser le concept d'observabilité sur lequel notre approche de diagnostic est basée. Ce concept a été présenté sous la forme de la définition d'un indice de confiance accordé à la réalisation des effets d'une opération. Cet indice de confiance est lié non seulement à la connaissance du niveau de captage interne aux chaînes fonctionnelle mais également sur la connaissance intrinsèque de leur comportement. Afin de prendre en compte les spécificités liées au niveau de coordination de ces chaînes fonctionnelles, nous avons été amenés à introduire le concept d'opérations de surveillance du flux de produits enrichies elles aussi d'un indice de confiance accordé quant à la réalisation des effets sur le flux de produits. L'exploitation de cet indice de confiance se révélera être la clef de voute du mécanisme

d'oubli et de réduction du modèle du fonctionnement passé présenté dans le cadre de ce document.

Cependant et avant d'aller plus avant dans la présentation de ce modèle et de sa génération, le chapitre suivant propose de présenter le formalisme qui sera retenu pour le représenter.

Chapitre 6

Formalisation du comportement des opérations

1 Introduction

Jusqu'ici, nous avons présenté démarche et modèles permettant à l'expert de formaliser une connaissance statique d'une partie opérative requise pour les besoins du diagnostic des services à un niveau coordination du CIM.

Cependant, comme nous avons pu le dévoiler section 7.1 du chapitre 3, le mécanisme de diagnostic proposé est basé sur la recherche de conflits dans un modèle représentant le fonctionnement passé. Ce modèle, composé de l'ensemble des opérations exécutées et décrivant donc les évolutions passées des chaînes fonctionnelles et du flux de produits, doit être généré en ligne (Suivi) au rythme du passage des produits qui conditionnent l'exécution de la loi de commande. Afin de préparer la structuration du mécanisme de suivi et de diagnostic qui seront proposés à partir du chapitre 7, un formalisme de représentation de ce modèle créé en ligne à partir des informations contenues dans le modèle de la partie opérative doit être sélectionné.

2 Fonctionnement anormal d'une chaîne fonctionnelle

Afin de pouvoir exprimer le fonctionnement ANormal d'une chaîne fonctionnelle, un prédicat unaire noté AN sera utilisé dans les modèles comportementaux des opérations. Contrairement au travaux développés utilisant la théorie de (Kleer et Williams, 1989), la valeur des pré-requis des opérations exécutées pourront avoir un impact sur la valeur de ce prédicat ; ceci afin de pouvoir modéliser l'impact d'une propagation de défaillance sur le fonctionnement des chaînes fonctionnelles. Pour une opération particulière, $\neg AN$ signifie que la chaîne fonctionnelle exécutant le service associé était en fonctionnement normal au moment du lancement de l'opération ou durant l'opération. Le fonctionnement normal d'une chaîne fonctionnelle est défini comme sa capacité à rendre l'ensemble de ses services. Son fonctionnement anormal pouvant alors être défini comme une perte d'au moins un de ses services, que ceci soit dû à une défaillance ou une propagation de défaillance. Ainsi, dans le cadre du diagnostic, AN signifiera soit que le fonctionnement de la chaîne fonctionnelle exécutant l'opération est à l'origine de la réception d'un CRA, soit qu'une propagation d'une défaillance a affecté le fonctionnement de la chaîne fonctionnelle.

Si le prédicat AN d'une opération précédemment exécutée est vrai, cela signifie que l'exécution de cette opération dans le passé est **suspectée** (potentiellement mal réalisée) ainsi que toutes les opérations offertes par la même chaîne fonctionnelle. Par opposition, suite à une analyse diagnostique, si le prédicat $\neg AN$ pour une opération est vrai, l'opération n'est pas suspectée par rapport au CRA reçu.

3 Formalisation du comportement des opérations d'action

Nous présentons dans cette section uniquement les notations et formalisations correspondant à une opération d'action. La section suivante présentera quant à elle la formalisation pour une opération de surveillance. La formalisation comportementale des opérations induites, d'acquisition et requises sont très proches de celle proposée ici. Le lecteur pourra se référer à l'annexe A pour s'en convaincre. Nous adopterons pour les informations décrites dans le chapitre 4 les notations présentées par la figure 6.1. Une opération Oa_i basé sur une chaîne fonctionnelle CF_k est composée de :

1. Du_i, la durée fixée de l'opération Oa_i.
2. ec_i, l'évolution de la chaîne fonctionnelle, elle-même composée de :
   - $EfT(ec_i)$, l'effet transitoire sur la chaîne fonctionnelle.

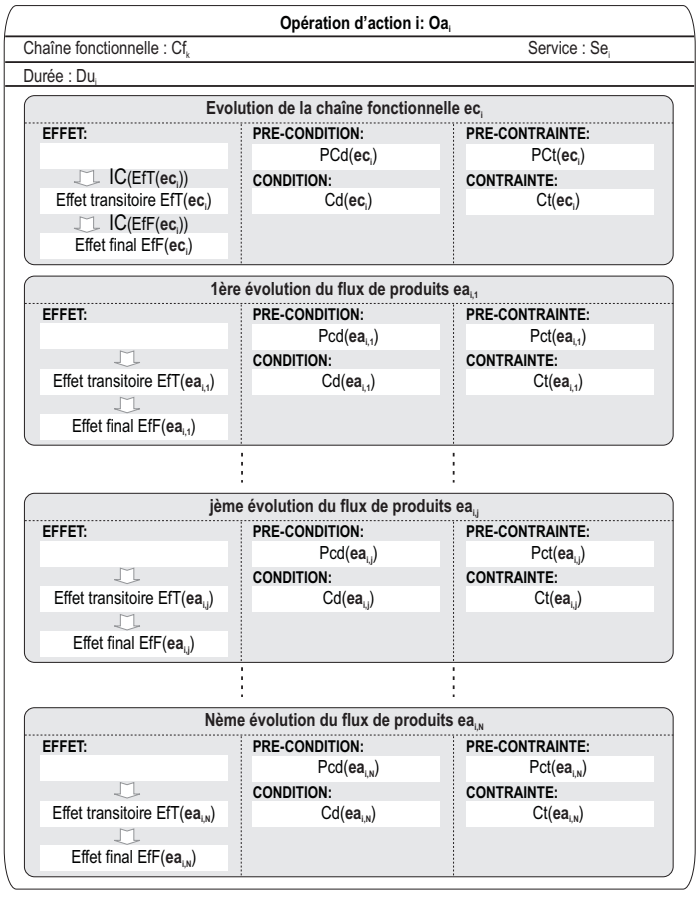


FIG. 6.1 – Notation opération d'action

- $EfF(ec_i)$, l'effet final sur la chaîne fonctionnelle.
- $PCd(ec_i)$, la pré-condition à respecter avant le lancement de l'opération pour que l'effet transitoire sur la chaîne fonctionnelle soit réalisé.
- $Cd(ec_i)$, la condition à respecter pendant l'exécution de l'opération pour que l'effet final sur la chaîne fonctionnelle soit réalisé.
- $PCt(ec_i)$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération si la pré-condition $PCd(ec_i)$ est vraie.
- $Ct(ec_i)$, la contrainte, à respecter durant l'exécution de l'opération.
- $IC(EfT(ec_i))$, l'indice de confiance associé à l'effet transitoire sur la chaîne fonctionnelle.
- $IC(EfF(ec_i))$, l'indice de confiance associé à l'effet final sur la chaîne fonctionnelle.

3. $ea_{i,j}$ pour $j \in [1, N_i]$, les évolutions associées du flux de produits, pour une opération Oa_i avec N_i évolutions possible du flux de produits. Chaque évolution $ea_{i,j}$ est elle même composée de :

   - $EfT(ea_{i,j})$, l'effet transitoire sur le flux de produits.
   - $EfF(ea_{i,j})$, l'effet final sur le flux de produits.
   - $PCd(ea_{i,j})$, la pré-condition à respecter avant le lancement de l'opération pour que l'effet transitoire $EfT(ea_{i,j})$ soit réalisé.
   - $Cd(ea_{i,j})$, la condition à respecter pendant l'exécution de l'opération pour que l'effet final $EfF(ea_{i,j})$ soit réalisé.
   - $PCt(ea_{i,j})$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération si la pré-condition $PCd(ea_{i,j})$ est vraie.
   - $Ct(ea_{i,j})$, la contrainte à respecter durant toute l'évolution $ea_{i,j}$.

La description d'une opération présentée ci-dessus, est réalisée dans un contexte général sans tenir compte de son contexte d'exécution. Un service lancé correspondra à une opération exécutée, composée de l'évolution de la chaîne fonctionnelle et d'un ensemble d'évolutions du flux de produits. Pour chaque opération exécutée, une liste d'indices contiendra les numéros j des évolutions $ea_{i,j}$ attendues du flux de produits, i.e. les $ea_{i,j}$ dont les pré-conditions $PCd(ea_{i,j})$ sont vraies. La Figure 6.2 illustre les différents indices utilisés dans la notation proposée. Il est à noter que d'un point de vue statique :

- 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,

et d'un point de vue dynamique :

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

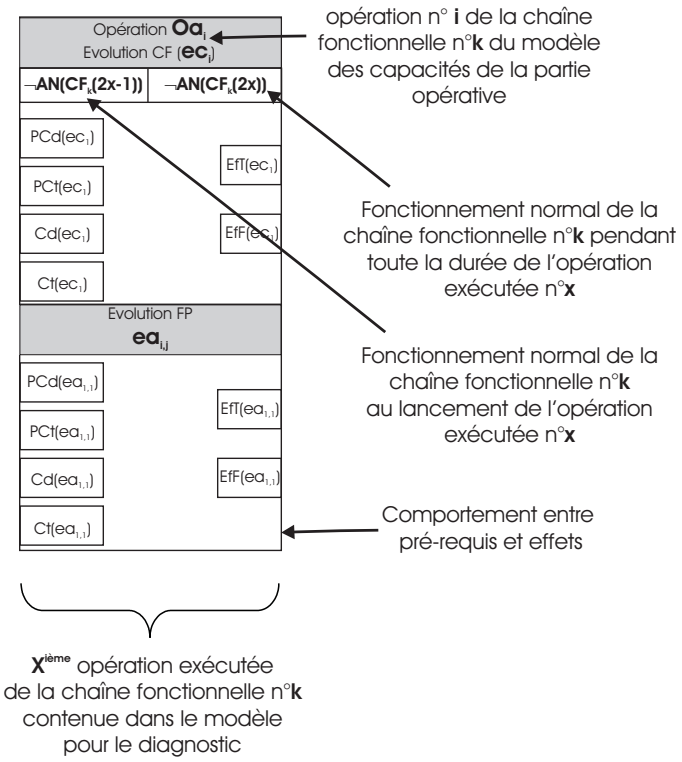


FIG. 6.2 – Illustration graphique d'une opération exécutée

3.1 Formalisation de l'évolution de la chaîne fonctionnelle

L'effet transitoire sur l'état de la chaîne fonctionnelle $EfT(ec_i)$ correspondant à l'évolution ec_i de l'opération Oa_i est réalisé si :

- la chaîne fonctionnelle est en fonctionnement normal au lancement du service. Ceci est noté $\neg AN(CF_k(2x-1))$, où x est la $x^{ième}$ 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 des évolutions associées du flux de produits qui doivent être réalisées sont respectées, autrement dit les pré-contraintes $PCt(ea_{i,j})$ des évolutions $ea_{i,j}$ dont les pré-conditions $PCd(ea_{i,j})$ sont vraies. En effet, le non respect d'une de ces contraintes entraîne une mauvaise exécution de l'opération.

De surcroît, nous considérerons que si l'effet $EfT(ec_i)$ est réalisé l'ensemble des pré-requis ci-dessus sont vrais.

Ceci nous donne donc :

Définition 6.1 *Effet transitoire de l'évolution de la chaîne fonctionnelle*

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

Il est à noter que dans cette définition $\neg AN(CF_k(2x-1))$ correspond à un prédicat unaire, $PCd(ec_i)$, $PCt(ec_i)$, $PCd(ea_{i,j})$ et $PCt(ea_{i,j})$ à des propositions logiques portant sur des tests de variables d'état et $EfT(ec_i)$ à la réalisation ou non de l'effet correspondant.

L'effet final sur l'état de la chaîne fonctionnelle $EfF(ec_i)$ correspondant à l'évolution ec_i de l'opération Oa_i est réalisé si :

- la chaîne fonctionnelle est en fonctionnement normal durant l'exécution du service. Ceci est noté $\neg AN(CF_k(2x))$,
- l'effet transitoire $EfT(ec_i)$ est réalisé,
- la contrainte $Ct(ec_i)$ et la condition $Cd(ec_i)$ sont respectées,
- les contraintes des évolutions associées du flux de produits qui sont réalisées sont respectées, autrement dit si les conditions $Cd(ea_{i,j})$ et les contraintes $Ct(ea_{i,j})$ des évolutions $ea_{i,j}$ dont les pré-conditions $PCd(ea_{i,j})$ sont vraies.

De surcroît, nous considérerons que si l'effet $EfF(ec_i)$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition 6.2 *Effet final de l'évolution de la chaîne fonctionnelle*

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

3.2 Formalisation des évolutions associées du flux de produits

L'effet transitoire sur l'état du flux de produits $EfF(ea_{i,j})$ pour $j \in [1, Ni]$ correspondant à l'évolution associée $ea_{i,j}$ de l'opération Oa_i est réalisé si :

- la chaîne fonctionnelle est en fonctionnement normal au lancement du service,
- la pré-contrainte $PCt(ea_{i,j})$ et la pré-condition $PCd(ea_{i,j})$ de $ea_{i,j}$ sont respectées,
- la pré-contrainte $PCt(ec_i)$ et la pré-condition $PCd(ec_i)$ de ec_i sont respectées, car sans effet sur la chaîne fonctionnelle il ne peut y avoir un effet sur le flux de produits,
- les pré-contraintes des autres évolutions associées du flux de produits qui doivent être réalisées sont respectées, autrement dit si les pré-contraintes $PCt(ea_{i,l})$ des évolutions $ea_{i,l}$ dont les pré-conditions $PCd(ea_{i,l})$ sont vraies pour $l \in [1, N_i] \setminus j$.

De surcroît, nous considérerons que si l'effet $EfT(ea_{i,l})$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition 6.3 *Effet transitoire d'une évolution du flux de produits*

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

L'effet final sur l'état du flux de produits $EfF(ea_{i,j})$ $\forall j \in [1, Ni]$ correspondant à l'évolution $ea_{i,j}$ de l'opération Oa_i est réalisé si :

- la chaîne fonctionnelle est en fonctionnement normal durant l'exécution du service,
- l'effet transitoire $EfT(ea_{i,j})$ est réalisé,
- la contrainte $Ct(ea_{i,j})$ et la condition $Cd(ea_{i,j})$ sont respectées,
- la contrainte $Ct(ec_i)$ et la condition $Cd(ec_i)$ sont respectées,
- les contraintes des autres évolutions associées du flux de produits qui sont réalisées sont respectées.

De surcroît nous considérerons que si l'effet $EfF(ea_{i,l})$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition 6.4 *Effet transitoire d'une évolution du flux de produits*

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

3.3 Conséquence du non respect des pré-contraintes et contraintes

Pour le diagnostic, une des extensions proposées dans le cadre de la modélisation est la prise en compte de l'impact des pré-contraintes et contraintes d'une opération sur le fonctionnement de la chaîne fonctionnelle ayant exécutée l'opération.

La non satisfaction de la pré-contrainte (violation d'une contrainte de sécurité) de l'évolution de la chaîne fonctionnelle ($PCt(ec_i)$), ou d'une des évolutions associées du flux de produits qui doivent être réalisées ($PCt(ea_{i,j})$ pour j tel que $PCd(ea_{i,j})$ est vraie) ont pour conséquence potentielle d'entraîner un fonctionnement anormal de la chaîne fonctionnelle correspondante dès le lancement de l'opération :

Définition 6.5 *Conséquence du non respect des pré-contraintes sur le fonctionnement des chaînes fonctionnelles au lancement des opérations*

$$\neg PCt(ec_i) \bigvee_{\exists j \in [1,Ni]} \neg [PCd(ea_{i,j}) \Rightarrow (PCt(ea_{i,j})] \Rightarrow AN(CF_k(2x-1))$$

De la même manière, la non satisfaction d'une contrainte de l'évolution de la chaîne fonctionnelle ($Ct(ec_i)$), ou des évolutions associées du flux de produits qui doivent être réalisées ($Ct(ea_{i,j})$ pour j tel que $PCd(ea_{i,j})$ est vraie) ont pour conséquence potentielle d'entraîner un fonctionnement anormal de la chaîne fonctionnelle correspondante durant l'exécution de l'opération :

Définition 6.6 *Conséquence du non respect des contraintes sur le fonctionnement des chaînes fonctionnelles durant l'exécution des opérations*

$$\neg Ct(ec_i) \bigvee_{\exists j \in [1,Ni]} \neg \Big[PCd(ea_{i,j}) \Rightarrow Ct(ea_{i,j}) \Big] \Rightarrow AN(CF_k(2x))$$

3.4 Formalisation du lien entre l'exécution des opérations

Nous avons montré dans le chapitre 3 qu'il était nécessaire dans le cadre de notre problématique de modéliser le lien entre la disponibilité des opérations d'une même chaîne fonctionnelle (i.e. le lien entre le bon fonctionnement d'une même chaîne fonctionnelle à plusieurs instants différents). Lorsque le diagnostic vient à suspecter l'exécution d'une opération d'une chaîne fonctionnelle, n'ayant qu'une connaissance du fonctionnement normal de la partie opérative, il est

nécessaire de suspecter les exécutions des opérations postérieures de la même chaîne fonctionnelle. Ceci implique que le prédicat $AN(CF_k(2x))$ est également une "sortie" de chaque opération exécutée du modèle pour le diagnostic afin de suspecter les opérations postérieures, et $AN(CF_k(2x-1))$ un pré-requis afin d'être en mesure de suspecter la disponibilité de l'opération correspondante si la disponibilité d'une opération antérieure a été suspectée. En formalisant le principe évoqué ci-dessus, on obtient :

Définition 6.7 *Fonctionnement anormal d'une même chaîne fonctionnelle*

$$AN(CF_k(x-1)) \Rightarrow AN(CF_k(x))$$

4 Comportement d'une opération de surveillance

La notation associée à une opération de surveillance est la suivante :

1. ec_i, l'évolution de la chaîne fonctionnelle,
2. $Surv_{i,j}$ pour $j \in [1, N_i]$, les différents comportements de produits, pour une opération Os_i avec N_i surveillance possible. Chaque comportement $Surv_{i,j}$ est lui même composé de :
   - $VE(Surv_{i,j})$, la variable d'état surveillée.
   - $valeur1, valeur2, ...$, les valeurs de la variable d'état qui peuvent être surveillées.
   - $PCd(Surv_{i,j})$, la pré-condition à respecter avant le lancement de l'opération pour effectuer la surveillance.
   - $Cd(Surv_{i,j})$, la condition à respecter pendant l'exécution de l'opération.
   - $PCt(Surv_{i,j})$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération si la pré-condition $PCd(Surv_{i,j})$ est vraie.
   - $Ct(Surv_{i,j})$, la contrainte à respecter.

Dans le cadre des défaillances de chaînes d'acquisition utilisées pour la surveillance, il est nécessaire de prendre en compte, dans le modèle pour le diagnostic, les opérations de surveillance une fois exécutées. Nous proposons d'utiliser la définition suivante quant au comportement de surveillance :

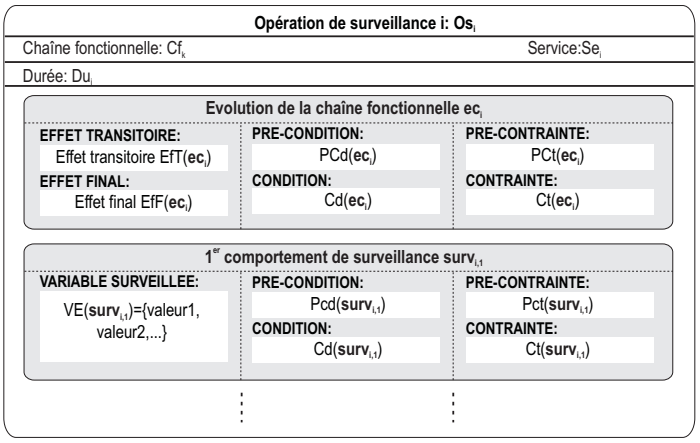


FIG. 6.3 – Notation opération de surveillance

Définition 6.8 *Comportement de surveillance*

$$\neg AN(CF_k(2x-1)) \wedge \neg AN(CF_k(2x)) \wedge (VE(Surv_{i,j}) == valeur) \wedge PCd(Surv_{i,j}) \wedge Cd(Surv_{i,j}) \wedge PCt(Surv_{i,j}) \wedge Ct(Surv_{i,j}) \wedge PCd(ec_i) \wedge Cd(ec_i) \wedge PCt(ec_i) \wedge Ct(ec_i) \bigwedge_{\forall l \in [1,N_i] \setminus j} \Big[PCd(Surv_{i,l}) \Rightarrow (PCt(Surv_{i,l}) \wedge Ct(Surv_{i,l})) \Big] \Leftrightarrow EfF(Surv_{i,j})$$

Cette définition sera utilisée pour mettre en évidence suite à la réception d'un compte rendu d'exécution incorrecte envoyé par la chaîne fonctionnelle (mettant en œuvre la surveillance), qu'une des conditions suivantes n'était pas respectée :

- l'opération Os_i était disponible au moment de son lancement et durant toute son exécution,
- la valeur prévue de $VE(Surv_{i,j})$ (accompagnant la requête de surveillance) est correcte
- la pré-condition $PCd(Surv_{i,j})$, la condition $Cd(Surv_{i,j})$, la pré-contrainte $PCt(Surv_{i,j})$ ou la contrainte $Ct(Surv_{i,j})$ du comportement de surveillance étaient satisfaites,
- la pré-condition $PCd(ec_i)$, la condition $Cd(ec_i)$, la pré-contrainte $PCt(ec_i)$ ou la contrainte $Ct(ec_i)$ du comportement de la chaîne fonctionnelle étaient respectées,
- les pré-contraintes et contraintes des autres comportement de surveillance qui ont été utilisées étaient respectées.

Le lancement d'un service de surveillance n'a pas d'effet sur les variables d'état du flux de produits. $EfF(Surv_{i,j})$ ne contient pas d'expression particulière mais est uniquement une valeur binaire : vrai si la surveillance s'est correctement exécutée, faux si une incohérence a été observée par la chaîne d'acquisition. Ainsi, $EfF(Surv_{i,j})$ permettra au diagnostic de gérer le cas de l'exécution d'une opération de surveillance de façon identique à toutes autres opérations.

5 Conclusion

Dans ce chapitre, nous nous sommes employés à présenter le formalisme sur lequel les mécanismes de suivi et de diagnostic vont s'appuyer pour d'une part générer le modèle du fonctionnement passé et d'autre part, en présence de dysfonctionnements, suspecter les opérations exécutées à l'origine possible du dysfonctionnement constaté ainsi que celles qui ont pu être affectées.

Le formalisme retenu est celui de la logique propositionnelle. Ce dernier nous a permis de définir les évolutions des chaînes fonctionnelles, les évolutions du flux de produits mais également les conséquences du non respect des pré-contraintes et des contraintes sur le fonctionnement des chaînes fonctionnelles. Après quoi le lien entre le bon fonctionnement d'une chaîne fonctionnelle durant l'exécution des opérations, ainsi que les spécificités induites par les opérations de surveillance, a été introduit.

A ce stade de l'étude, tous les éléments requis sont désormais présentés pour envisager de proposer les mécanismes de suivi et de diagnostic de services retenus. C'est ce que nous proposons de présenter dans la partie suivante.

Troisième partie

Processus de Diagnostic

Chapitre 7

Génération et gestion du modèle pour le diagnostic

1 Introduction

Le modèle utilisé dans le cadre du diagnostic est une extension du modèle proposé par (Reiter, 1987). Nous y retrouvons donc une structure identique :

- un ensemble d'opérations précédemment lancées suite à l'exécution de la loi de commande,
- une description des comportements des opérations,
- une description des liens de causalité entre les opérations.

Au travers ce modèle, le diagnostic aura à rechercher à partir de la réception d'un CRA (cf. section 5.1 page 24) ses origines possibles et leurs conséquences sur les capacités opératoires (cf. section 3 page 30). Cette recherche est basée sur l'analyse du comportement de chaque opération exécutée, ainsi que sur celle des liens entre ces différentes opérations. Contrairement au travaux développés dans le diagnostic logique, le nombre d'éléments constituant le modèle n'est pas fixé, mais évolue en fonction de l'exécution de la loi de commande.

Ce chapitre sera ainsi articulé autour de deux paragraphes. Le premier s'attachera à présenter les principes du mécanisme de génération du modèle pour le diagnostic. Sur la base de ces principes, l'algorithme retenu sera alors présenté. Le deuxième paragraphe sera alors consacré quant à lui, à la présentation de règles de réduction du modèle.

2 Génération du modèle

2.1 Principe de génération du modèle

Au niveau coordination des chaînes fonctionnelles, le diagnostic doit être capable de prendre en compte toutes les évolutions passées de la partie opérative et du flux de produits. Ces évolutions dépendent directement de l'exécution de la loi de commande.

2.1.1 Exécution d'une loi de commande

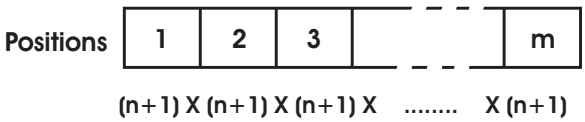
La loi de commande contient l'ensemble des séquences d'opérations lancées par le niveau coordination.

S'il est possible de prévoir à l'avance la séquence ou les séquences d'opérations ainsi que le comportement de chacune des opérations, il est alors tout à fait envisageable de pouvoir construire le modèle pour le diagnostic hors ligne. Ceci permettant de réaliser l'analyse diagnostique hors ligne pour tous les CRA possibles. Dans le cadre d'un système manipulant un flux de produits unique (mono flux) suivant systématiquement le même routage, ceci est tout à fait envisageable. Cependant, dès lors que l'on s'intéresse à des flux de produits différents (multi flux), le routage et leurs transformations peuvent dépendre du type de produits, et donc de l'ordre d'arrivée des produits dans le sous-système piloté. La combinatoire des séquences d'opérations possibles peut alors rapidement exploser. Afin d'illustrer ce problème, prenons l'exemple d'un sous système piloté dont les caractéristiques sont les suivantes :

- manipulation de n types de produits différents sur lesquels un traitement spécifique doit être fait,

- au maximum m produits différents peuvent être présents en même temps.

Ceci nous donne donc $n+1$ combinaisons de produits différents (i.e. n produits ou l'absence de produit) à chacune des m positions :



Ainsi $(n+1)^m$ possibilités différentes d'état du flux de produits sont possibles et donc potentiellement autant de séquences d'opérations avec des comportements différents pour y parvenir. Pour un sous système manipulant 5 types de produits différents, dont seulement 10 peuvent être présents simultanément dans le sous système piloté, nous obtenons 60 millions de combinaisons différentes. Si nous considérons 7 types de produits différents, nous obtenons plus d'un milliard de combinaisons possibles...

Il devient dans ce cas irréaliste de construire l'ensemble des modèles pour chacune des combinaisons possibles. Nos travaux se sont naturellement portés sur une génération en ligne du modèle. Ceci nous amène alors à découper en deux parties le problème de diagnostic : un problème de suivi du sous-système piloté permettant de fournir toutes les informations utiles à une deuxième phase de diagnostic suite à la réception par le niveau coordination d'un CRA.

2.1.2 Fonction suivi

Le suivi a comme tâche d'enregistrer et d'organiser toutes les informations nécessaires au diagnostic durant le fonctionnement normal du sous-système piloté. Il génère le modèle du diagnostic et assure la gestion de sa taille. Cette fonction suivi permet à la fonction diagnostic de pouvoir produire rapidement un résultat de diagnostic suite à la réception par le niveau

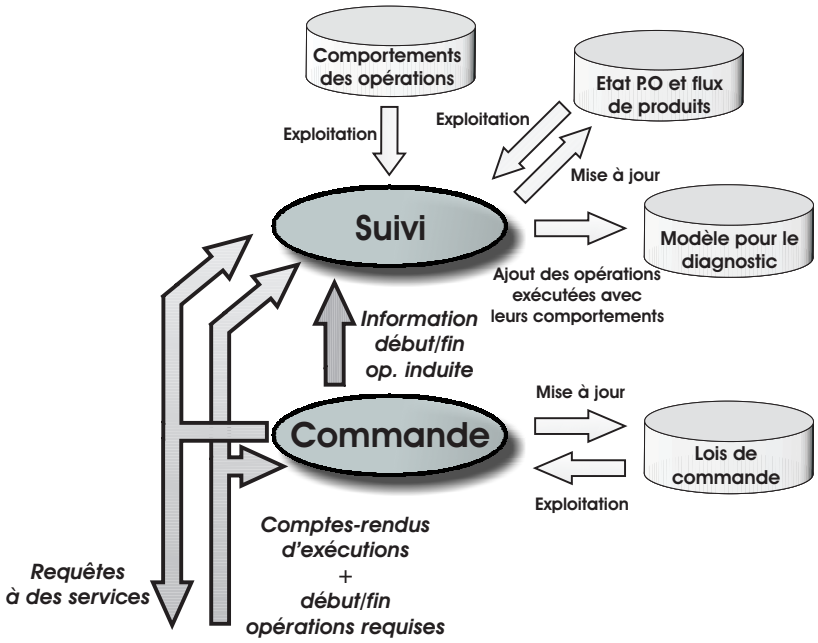


FIG. 7.1 – Principe de génération du modèle

coordination d'un CRA. A chaque réception d'une information de début d'opération (requête d'un service, information de début d'une opération induite ou requise), et de fin d'opération (compte rendu d'un service, information de fin d'une opération induite ou requise), le suivi ajoute une opération exécutée et la connecte aux autres opérations déjà présentes dans le modèle pour le diagnostic (cf. Figure 7.1). Toutefois, afin de pouvoir ajouter le comportement de l'opération dans son contexte d'exécution, il est nécessaire de connaître l'état de la partie opérative et du flux de produits au lancement de l'opération. Dans cet objectif, la fonction suivi met à jour, au fur et à mesure de l'exécution des opérations, la liste des variables d'état. Ainsi, le suivi détermine à partir de l'état courant et pour chaque opération i exécutée, une liste d'indices J contenant les numéros j des évolutions $ea_{i,j}$ du flux de produits, i.e. les $ea_{i,j}$ dont les pré-conditions $PCd(ea_{i,j})$ sont vraies.

Après avoir exposé les principes de la génération du modèle pour le diagnostic, il est maintenant proposé d'étudier chacune des opérations exécutées le constituant ainsi que leurs liens.

2.2 Description comportementale

La modélisation comportementale d'une opération exécutée sera bien entendu basée sur les définitions du comportement des opérations développées au chapitre 6. Toutefois cette formalisation a été faite dans un contexte général sans tenir compte ni de son contexte d'exécution (évolutions du flux de produits réalisées dont les numéros sont contenus dans J), ni des expressions contenues dans chacun des pré-requis (pré-conditions, conditions...). Pour chaque opération exécutée, une description de son comportement sera ajoutée au modèle de diagnostic. Ce comportement correspond à la fusion de trois éléments :

– le modèle de comportement des opérations définissant le lien entre les pré-conditions, les

conditions, les pré-contraintes, les contraintes, et les effets des évolutions, ainsi que le lien entre les disponibilités des opérations,

– la description de l'opération particulière (contenue dans le modèle des capacités de la partie opérative), définissant les expressions logiques entre les tests sur les variables d'état, contenus dans les pré-requis (pré-conditions, conditions, pré-contraintes, contraintes) et dans les effets,

– les informations permettant de déduire le contexte d'exécution d'une opération, c'est à dire l'ensemble J des évolutions attendues au lancement de l'opération.

Le diagnostic est basé sur la recherche des origines possibles en terme d'exécutions non conformes des opérations. De surcroît, le diagnostic est également basé sur la recherche des autres conséquences possibles, c'est à dire partant des opérations dont l'exécution n'est potentiellement pas conforme (origines obtenues), retrouver toutes les conséquences sur l'exécution des opérations (Boufaied, 2000). Afin de dégager les informations pertinentes vis-à-vis du diagnostic, il est proposé ci-après de définir l'exécution non conforme d'une opération.

2.2.1 Caractérisation de l'exécution non conforme d'une opération

Le principe de la propagation de la suspicion va rechercher dans le modèle pour le diagnostic les exécutions potentiellement non conformes des opérations. Le terme non conforme réfère au comportement d'une opération. En fonctionnement normal une opération exécutée, se compose d'une évolution de la chaîne fonctionnelle ec_i et d'un ensemble d'évolutions du flux de produits $ea_{i,j}$ pour $j \in J$. Ceci implique que toutes les évolutions $ea_{i,j}$ pour $j \notin J$ ne sont pas réalisées en fonctionnement normal. La non-conformité du comportement attendu se traduit donc comme :

– la non réalisation de $EfT(ec_i)$ ou,

– la non réalisation de $EfF(ec_i)$ ou,

– la non réalisation de $EfT(ea_{i,j})$ pour $j \in J$ ou,

– la non réalisation de $EfF(ea_{i,j})$ pour $j \in J$ ou,

– la réalisation de $EfT(ea_{i,j})$ pour $j \notin J$ ou,

– la réalisation de $EfF(ea_{i,j})$ pour $j \notin J$.

2.2.2 Origine de l'exécution non conforme d'une opération

Les origines de l'exécution non conforme se retrouvent dans les définitions des comportements des opérations. Prenons par exemple $EfT(ec_i)$:

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

L'origine de la non réalisation de l'effet $EfT(ec_i)$ est due à au moins un des points suivants qui sont donc à suspecter :

- une indisponibilité des opérations de la chaîne CF_k au moment du lancement de l'opération (prédicat $\neg AN(CF_k(2x-1))$ faux),
- non respect de la pré-condition de l'évolution de la chaîne fonctionnelle ($PCd(ec_i)$ $faux$),
- non respect de la pré-contrainte de l'évolution de la chaîne fonctionnelle ($PCt(ec_i)$ $faux$),
- non respect d'une pré-contrainte d'une évolution associée du flux de produits qui est réalisée ($PCt(ea_{i,j})$ faux avec $PCd(ea_{i,j})$ vraie).

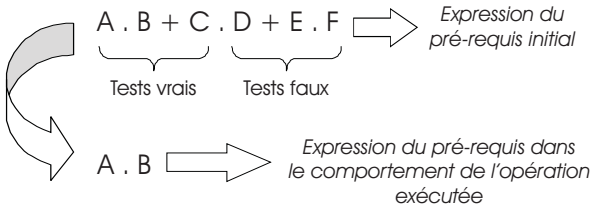
Pour chacun des pré-requis évoqués, l'exécution non conforme est due soit à la non satisfaction de l'expression d'un pré-requis, soit au contraire, à la satisfaction de l'expression $PCd(ea_{i,j})$ pour $j \notin J$, i.e. le lancement non prévu par la loi de commande d'une évolution du flux de produits.

2.2.3 Expression des pré-requis satisfaits et non satisfaits

Afin de faciliter la recherche du diagnostic, lors de la construction du modèle seuls les tests pertinents seront conservés.

Pré-requis prévus satisfaits :

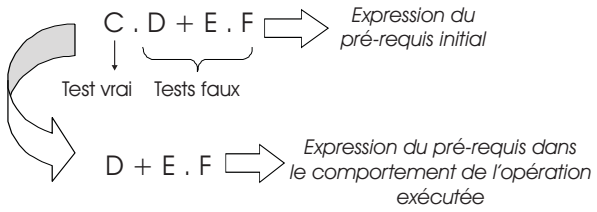
Dans le cadre du diagnostic, il est important dans l'expression d'un pré-requis satisfait de connaître les tests sur les variables d'état dont la remise en cause induit la non satisfaction de l'expression du pré-requis. Ainsi, dans le comportement d'une opération exécutée les pré-conditions PCd, les pré-contraintes PCt, les conditions Cd et les contraintes Ct des évolutions attendues (i.e. pré-requis satisfait) contiendront une expression logique correspondant à l'expression logique initiale à laquelle les conjonctions fausses sont retirées. Ceci peut être illustré par la figure suivante :



En effet, si le pré-requis est non satisfait, uniquement le test A ou le test B est à remettre en cause. Ainsi, lors de la génération seuls ces deux tests sont conservés.

Pré-requis prévus non satisfaits :

L'expression d'un pré-requis non satisfait d'une opération exécutée contiendra une expression logique correspondant à l'expression logique initiale du pré-requis à laquelle les tests vrais sont retirés. Ceci peut être illustré par la figure suivante :



En effet, le pré-requis sera satisfait si le test D ou E ou F est remis en cause. Ainsi, lors de la génération uniquement ces trois tests sont conservés. Une des premières utilisations de ce principe est sur l'expression des pré-conditions des évolutions non attendues du flux de produits ($ea_{i,j}$ pour $j \notin J$).

2.2.4 Expression des évolutions non souhaitées du flux de produits d'une opération exécutée

Pour une évolution du flux de produits $ea_{i,j}$, si $j \notin J$ alors l'évolution en fonctionnement normal n'est pas réalisée. Toutefois son lancement intempestif est également une origine de l'exécution non conforme d'une opération. Deux cas sont à distinguer :

- la réalisation potentiellement incorrecte d'une opération amène le diagnostic à suspecter la transgression d'une pré-contrainte ou contrainte par l'intermédiaire du lancement de cette évolution. Dans ce cas il faut tout d'abord suspecter les variables d'état ne satisfaisant pas la pré-condition de l'évolution mais aussi les variables d'état satisfaisant la pré-contrainte de l'évolution.

- Le diagnostic suspecte la réalisation de l'effet transitoire et final. Dans ce cas nous suspecterons également les variables d'état ne satisfaisant pas la pré-condition de l'évolution mais cette fois-ci nous suspecterons les variables d'état qui ne satisfaisaient pas les pré-requis de l'effet. Aussi, pour que $EfT(ea_{i,j})$ soit réalisé il faut que la pré-condition et la pré-contrainte de $ea_{i,j}$ soient satisfaites. Pour que $EfF(ea_{i,j})$ soit réalisé il faut que $EfT(ea_{i,j})$ soit réalisé et que la condition et la contrainte de $ea_{i,j}$ soient satisfaites.

Lors de la génération du comportement d'une opération contenant une évolution non attendue, il est nécessaire de conserver uniquement la partie non satisfaite de l'expression de la pré-condition et de la condition. Pour la pré-contrainte ou la contrainte, si l'expression est prévue non satisfaite (ou respectivement satisfaite), il est nécessaire de conserver uniquement la partie non satisfaite de l'expression (ou respectivement la partie satisfaite). Ainsi les pré-requis seront structurés au mieux pour la recherche effectuée par le diagnostic. Ils seront déterminés en utilisant les principes de la section précédente et seront notés comme suit :

- $PCd\_NS(ea_{i,j})$ sera composée d'une expression logique correspondant à l'expression de la pré-condition de $ea_{i,j}$ à laquelle les tests vrais portant sur les variables d'état ont été retirés,

- $Cd\_NS(ea_{i,j})$,

- $PCt\_NS(ea_{i,j})$ sera une expression logique correspondant à l'expression de la pré-contrainte de $ea_{i,j}$ à laquelle les tests vrais portant sur les variables d'état ont été retirés,

- ou $PCt\_S(ea_{i,j})$ sera 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,

- $Ct\_NS(ea_{i,j})$,

- ou $Ct\_S(ea_{i,j})$.

Ces pré-requis permettent ainsi de prendre en compte dans le comportement d'une opération exécutée les évolutions non souhaitées du flux de produits. Toutefois, l'ensemble des évolutions non souhaitées du flux de produits n'est pas forcément associé à la réalisation d'une opération. C'est le cas d'une opération induite réalisée (due à une propagation de défaillance) mais non initialement prévue dans la loi de commande.

2.2.5 Évolutions non souhaitées d'une opération induite non prévue

Une propagation de défaillance a pour conséquence potentielle la réalisation d'une opération induite qui n'était pas prévue dans la loi de commande. La prise en compte de ce cas par le diagnostic est une tâche délicate. En effet, toutes les autres évolutions prévues ou non prévues sont associées à une information de début d'opération, ce qui n'est pas le cas pour une opération induite non prévue. Si nous souhaitions intégrer durant la phase de suivi ce type d'évolutions non souhaitées au modèle pour le diagnostic, il serait nécessaire pour chaque opération exécutée de regarder si sa réalisation incorrecte pourrait modifier la pré-condition d'une évolution d'une opération induite, et si tel est le cas ajouter les évolutions non souhaitées de cette opération induite. Ce mécanisme entraînerait une importante augmentation de la taille du modèle. Aussi nous proposons de construire durant la phase de diagnostic uniquement les opérations induites non prévues qui pourraient expliquer la réception d'un CRA, ou qui pourraient être la conséquence d'une propagation des défaillances premières.

2.3 Description des liens de causalité entre les opérations

Un deuxième aspect développé ici concerne la description des liens de causalité entre les opérations exécutées contenues dans le modèle pour le diagnostic. Cette description définit les connexions entre les pré-requis d'une opération et les effets des opérations antérieures. Chacune des variables d'état contenues dans PCd, PCt, Cd, Ct, PCd_NS, Cd_NS, PCt_NS, Ct_NS, PCt_S ou Ct_S du modèle des comportements de l'opération sera reliée avec :

- le dernier effet ayant modifié cette variable d'état, rendant ainsi le test correspondant vrai

(ou faux pour les pré-requis non satisfaits),

– les effets non souhaités des opérations qui auraient pu modifier (en fonctionnement anormal) cette variable d'état à la suite du dernier effet lui ayant donné sa valeur, et ainsi rendre le test faux (ou vrai pour les pré-requis non satisfaits). En effet, une opération qui ne doit pas influer sur une variable d'état en fonctionnement normal, peut voir son comportement modifié suite à une propagation de défaillance, et ainsi avoir un effet sur cette variable d'état.

Cette description a pour objectif de fournir au diagnostic le lien entre l'exécution des opérations. Ainsi, suite à l'analyse d'une opération afin de connaître les pré-requis à l'origine possible de sa mauvaise exécution, le diagnostic pourra propager la suspicion de ces pré-requis sur l'exécution antérieure d'opérations.

2.4 Algorithme de génération du modèle

A la réception d'une information de début (respectivement de fin d'une opération) (cf. section 2.1.1 page 76) la procédure de suivi présentée dans l'algorithme 1 (respectivement dans l'algorithme 2) est lancée pour réaliser les actions suivantes :

– horodatage du début de l'opération (ou de la fin) afin que le diagnostic puisse connaître l'ordre des opérations.

– Pour la réception d'une information de début d'opération uniquement, détermination à partir de l'état courant de la partie opérative et du flux de produits du comportement de l'opération (liste des indices J des évolutions réalisées).

– Ajout des effets transitoires réalisés et non souhaités de l'opération exécutée (ou effets finaux) : pour cela non seulement le comportement (dans le contexte d'exécution) liant les pré-requis aux effets est enregistré mais également le lien de causalité avec les autres opérations. Il est à noter que si l'effet porte sur une variable d'état d'un produit, cet effet fait référence à un produit à une position. Ainsi l'algorithme transposera automatiquement cette référence à l'identifiant du produit (cf. section 2.3 page 50).

– Mise à jour des variables d'état de la partie opérative et du flux de produits en fonction des effets réalisés.

– Pour la réception d'une information de fin d'opération uniquement, appel de la procédure de réduction de modèle... En effet, au fur et à mesure de la construction du modèle, l'exploitation de l'indice de confiance permet d'en réduire la taille. Cette procédure est décrite en détail dans la prochaine section.

Ajouter Opération O_i
Enregistrer l'instant $Début(O_i)$
Enregistrer x, le numéro d'opération exécutée de CF_k (k=0 si opération non exécutée par une chaîne fonctionnelle du sous-système piloté, i.e. pour les opérations induites et requises)
//Détermination de $J_{k,x}$ représentant le comportement de la x^{me} opération exécutée de la chaîne fonctionnelle CF_k//
Pour j allant de 1 à Ni **faire**
  Si ($PCd(ea_{i,j})$ == vrai) **Alors**
    $J_{k,x} = J_{k,x} \cup \{j\}$
  Fin Si
Fin Pour
//Ajout du début de l'évolution de la chaîne fonctionnelle//
Si (ec_i existe) **Alors**
  Ajouter évolution ec_i
  Ajouter $PCd(ec_i)$, $PCt(ec_i)$, $EfT(ec_i)$
  Rechercher les derniers EfF tel que $PCd(ec_i)$ soit vraie et ajouter aux connexions
  Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCd(ec_i)$ et ajouter aux connexions
  Rechercher les derniers EfF tel que $PCt(ec_i)$ soit vraie et ajouter aux connexions
  Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCt(ec_i)$ et ajouter aux connexions
Fin Si
//Ajout du début des évolutions souhaitées du flux de produits//
Pour tout $j \in J_{k,x}$ **faire**
  Ajouter évolution $ea_{i,j}$ Ajouter $PCd(ea_{i,j})$, $PCt(ea_{i,j})$, $EfT(ea_{i,j})$
  Rechercher les derniers EfF tel que $PCd(ea_{i,j})$ soit vraie et ajouter aux connexions
  Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCd(ea_{i,j})$ et ajouter aux connexions
  Rechercher les derniers EfF tel que $PCt(ea_{i,j})$ soit vraie et ajouter aux connexions
  Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCt(ea_{i,j})$ et ajouter aux connexions
Fin Pour
//Ajout du début des évolutions non souhaitées du flux de produits//
Pour tout $j \notin J_{k,x}$ **faire**
  Ajouter évolution $ea_{i,j}$ Ajouter $PCd\_NS(ea_{i,j})$, $PCt\_NS(ea_{i,j})$, $PCt\_S(ea_{i,j})$, $\neg EfT(ea_{i,j})$
  Rechercher les derniers EfF portant sur $PCd\_NS(ea_{i,j})$ ajouter aux connexions
  Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCd\_NS(ea_{i,j})$ et ajouter aux connexions
  Rechercher les derniers EfF portant sur $PCt\_S(ea_{i,j})$ et ajouter aux connexions
  Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCt\_S(ea_{i,j})$ et ajouter aux connexions
  Rechercher les derniers EfF portant sur $PCt\_NS(ea_{i,j})$ et ajouter aux connexions
  Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCt\_NS(ea_{i,j})$ et ajouter aux connexions
Fin Pour

Alg. 1: Génération du modèle suite à la réception d'une information de début d'opération

Enregistrer l'instant $fin(O_i)$
//Ajout de la fin de l'évolution de la chaîne fonctionnelle//
Si (ec_i existe) **Alors**
Ajouter $Cd(ec_i)$, $Ct(ec_i)$, $EfF(ec_i)$ dans ec_i de $(O_i(y))$
Rechercher les derniers EfF tel que $Cd(ec_i)$ soit vraie et ajouter aux connexions
Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Cd(ec_i)$ et ajouter aux connexions
Rechercher les derniers EfF tel que $Ct(ec_i)$ soit vraie et ajouter aux connexions
Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Ct(ec_i)$ et ajouter aux connexions
Fin Si
//Ajout de la fin des évolutions souhaitées du flux de produits//
Pour tout $j \in J_{k,x}$ **faire**
Ajouter $Cd(ea_{i,j})$, $Ct(ea_{i,j})$, $EfF(ea_{i,j})$ dans $ea_{i,j}$
Rechercher les derniers EfF tel que $Cd(ea_{i,j})$ soit vraie et ajouter aux connexions
Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Cd(ea_{i,j})$ et ajouter aux connexions
Rechercher les derniers EfF tel que $Ct(ea_{i,j})$ soit vraie et ajouter aux connexions
Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Ct(ea_{i,j})$ et ajouter aux connexions
Fin Pour
//Ajout de la fin des évolutions non souhaitées du flux de produits//
Pour tout $j \notin J_{k,x}$ **faire**
Ajouter évolution $ea_{i,j}$ Ajouter $Cd\_NS(ea_{i,j})$, $Ct\_NS(ea_{i,j})$, $Ct\_S(ea_{i,j})$, $\neg EfF(ea_{i,j})$
Rechercher les derniers EfF portant sur $Cd\_NS(ea_{i,j})$ ajouter aux connexions
Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Cd\_NS(ea_{i,j})$ et ajouter aux connexions
Rechercher les derniers EfF portant sur $Ct\_S(ea_{i,j})$ et ajouter aux connexions
Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Ct\_S(ea_{i,j})$ et ajouter aux connexions
Rechercher les derniers EfF portant sur $Ct\_NS(ea_{i,j})$ et ajouter aux connexions
Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Ct\_NS(ea_{i,j})$ et ajouter aux connexions
Fin Pour
Réduction_modèle(O_i)

Alg. 2: Génération du modèle suite à la réception de l'information de fin d'une opération

3 Maîtrise de la taille du modèle

3.1 Exploitation des indices de confiance

La section précédente a présenté un processus de construction d'un modèle pour le diagnostic. Ce modèle est basé sur les opérations exécutées ainsi que leurs liens. Au fur et à mesure du fonctionnement du SAP, des opérations sont ajoutées à ce modèle. Il est donc nécessaire de

mettre en place un processus de gestion de la taille de ce modèle. Dans ce cadre, il est proposé ici d'exploiter l'indice de confiance des opérations défini au chapitre 5.

Il est proposé d'utiliser l'hypothèse formulée à la section 3 page 61 : "toute évolution de la partie opérative ou du flux de produits correspondant à un effet dont l'indice de confiance est de 1 sera qualifiée de correcte". Dans un deuxième temps, des règles de réduction présentées ci-après permettent de propager cette confiance au sein du modèle (i.e. de qualifier d'autres informations de correctes). D'une façon générale, l'ensemble des informations ayant été qualifiées de correctes ne seront pas conservées dans le modèle afin de limiter la recherche du diagnostic. Ceci paraît tout à fait justifié dans le sens où ces informations correspondent soit à des informations ayant été corrélées avec une connaissance comportementale du système, soit à des informations dont la nature correcte était indispensable pour que les informations corrélées soient correctes. Cette démarche peut être assimilée à l'hypothèse d'exonération utilisée dans de nombreux travaux de diagnostic (Dubuisson, 2001), revenant à dire que toutes défaillances ou compositions de défaillances sont potentiellement observables par un module de contrôle/commande.

3.2 Règles de réduction

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. Il est à noter que cette propagation de la confiance doit se faire par un parcours arrière du modèle mais également par un parcours avant.

Afin d'illustrer ce principe prenons l'exemple d'une opération Oa_1, $1^{ère}$ opération exécutée par la chaîne fonctionnelle CF_1. L'opération Oa_1 exécutée est composée d'une évolution de la chaîne fonctionnelle ec_1 et d'une évolution du flux de produits $ea_{1,1}$, seule évolution possible du flux de produits. Considérons que le lancement d'une opération de surveillance Os_2, $1^{ère}$ instance des opérations exécutées par la chaîne fonctionnelle CF_2 permet de vérifier l'effet final $EfF(ea_{1,1})$ par l'intermédiaire d'un captage sur la valeur de la variable d'état $VE(Surv_{2,1})$ ($Surv_{2,1}$ est le seul comportement de surveillance de l'opération Os_2) et donc de lui accorder un niveau de confiance de 1 en l'absence de compte rendu de mauvaise exécution.

Les définitions associées à l'opération Oa_1 sont donc :

$$\neg AN(CF_1(1)) \wedge PCd(ec_1) \wedge PCt(ec_1) \wedge [PCd(ea_{1,1}) \Rightarrow PCt(ea_{1,1})] \Leftrightarrow EfT(ec_1) \tag{7.1}$$

$$\neg AN(CF_1(2)) \wedge EfT(ec_1) \wedge Cd(ec_1) \wedge Ct(ec_1) \wedge [PCd(ea_{1,1}) \Rightarrow Ct(ea_{1,1})] \Leftrightarrow EfF(ec_1) \tag{7.2}$$

$$\neg AN(CF_1(1)) \wedge PCd(ea_{1,1}) \wedge PCt(ea_{1,1}) \wedge PCd(ec_1) \wedge PCt(ec_1) \Leftrightarrow EfT(ea_{1,1}) \quad (7.3)$$

$$\neg AN(CF_1(2)) \wedge EfT(ea_{1,1}) \wedge Cd(ea_{1,1}) \wedge Ct(ea_{1,1}) \wedge Cd(ec_1) \wedge Ct(ec_1) \Leftrightarrow EfF(ea_{1,1}) \quad (7.4)$$

L'opération de surveillance Os_2 est quant à elle définie par :

$$\neg AN(CF_2(1)) \wedge PCd(ec_2) \wedge PCt(ec_2) \wedge [PCd(Surv_{2,1}) \Rightarrow PCt(Surv_{2,1})] \Leftrightarrow EfT(ec_2) \quad (7.5)$$

$$\neg AN(CF_2(2)) \wedge EfT(ec_2) \wedge Cd(ec_2) \wedge Ct(ec_2) \wedge [PCd(Surv_{2,1}) \Rightarrow Ct(Surv_{2,1})] \Leftrightarrow EfF(ec_2) \quad (7.6)$$

$$\neg AN(CF_2(1)) \wedge \neg AN(CF_2(2)) \wedge (VE(Surv_{2,1}) == valeur) \wedge PCd(Surv_{2,1}) \wedge Cd(Surv_{2,1}) \wedge PCt(Surv_{2,1}) \wedge Ct(Surv_{2,1}) \wedge PCd(ec_2) \wedge Cd(ec_2) \wedge PCt(ec_2) \wedge Ct(ec_2) \Leftrightarrow vraie \quad (7.7)$$

Avec : $IC(Surv_{2,1}) = 1$

Suite à l'exécution de l'opération Oa_1 et de l'opération de surveillance Os_2, considérons que les deux chaînes fonctionnelles CF_1 et CF_2 nous renvoient des comptes-rendus d'exécutions correctes. L'exécution de l'opération de surveillance a un indice de confiance de 1. Dans un premier temps, le processus de réduction propagera en arrière la confiance accordée à cette exécution comme représenté schématiquement dans la Figure 7.2 :

1. selon la définition 7.7 (pour que l'exécution de l'opération soit correcte toutes les conditions requises doivent être correctes), $\neg AN(CF_2(1))$, $\neg AN(CF_2(2))$ ainsi que toutes les valeurs des variables d'état, $(VE(Surv_{2,1}) == valeur)$, $PCd(Surv_{2,1})$, $Cd(Surv_{2,1})$, $PCt(Surv_{2,1})$, $Ct(Surv_{2,1})$, $PCd(ec_2)$, $Cd(ec_2)$, $PCt(ec_2)$, et $Ct(ec_2)$ sont qualifiées de correctes.

2. La valeur de la variable d'état surveillée $VE(Surv_{2,1})$ ainsi que l'effet correspondant $EfF(ea_{1,1})$ sont donc qualifiés de corrects. Selon la définition 7.4 et par propagation arrière de la confiance accordée aux pré-requis $AN(CF_1(2))$, ainsi que les valeurs des variables d'état contenues dans $EfT(ea_{1,1})$, $Cd(ea_{1,1})$, $Ct(ea_{1,1})$, $Cd(ec_1)$ et $Ct(ec_1)$ sont donc qualifiées de correctes.

3. L'effet $EfT(ea_{1,1})$ est donc qualifié de correct. Selon la définition 7.3 et par propagation arrière de la confiance accordée aux pré-requis, $\neg AN(CF_1(1))$, $PCd(ea_{1,1})$, $PCt(ea_{1,1})$, $PCd(ec_1)$ et $PCt(ec_1)$ sont qualifiés correctes.

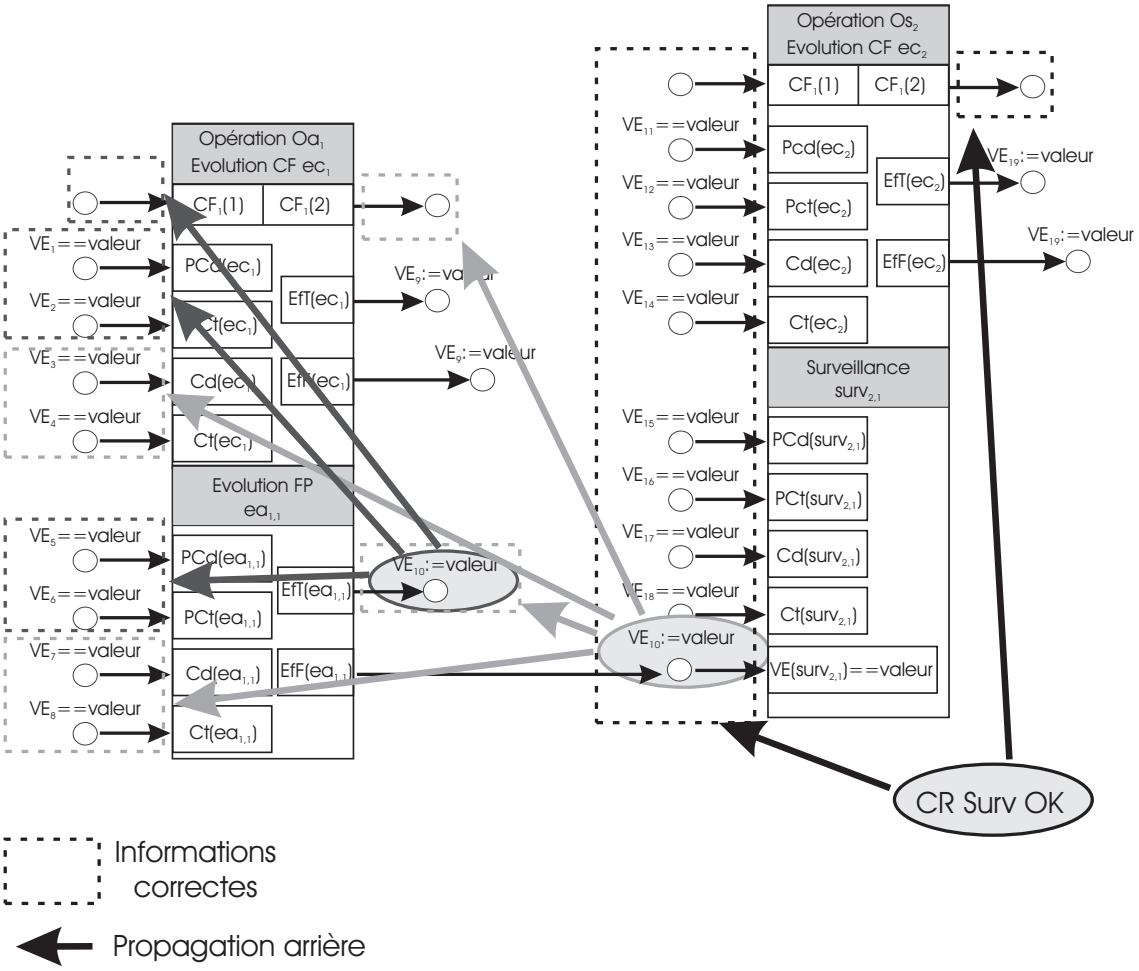


FIG. 7.2 – Propagation arrière de la confiance accordée aux informations

Dans un deuxième temps le processus de réduction va propager vers l'avant la confiance accordée aux informations comme représenté schématiquement par la Figure 7.3 :

1. Comme les pré-requis correspondants à $\neg AN(CF_2(1))$, $PCd(ec_2)$, $PCt(ec_2)$, $PCd(Surv_{2,1})$ et $PCt(Surv_{2,1})$ ont été qualifiés de corrects et par application de la définition 7.5, $EfT(ec_2)$ est qualifié correct.

2. Comme les pré-requis correspondants à $\neg AN(CF_2(2))$, $EfT(ec_2)$, $Cd(ec_2)$, $Ct(ec_2)$, $PCd(Surv_{2,1})$ et $Ct(Surv_{2,1})$ ont été qualifiés de corrects et par application de la définition 7.6, $EfF(ec_2)$ est qualifié correct.

3. Comme les pré-requis correspondants à $\neg AN(CF_1(1))$, $PCd(ec_1)$, $PCt(ec_1)$, $PCd(ea_{1,1})$ et $PCt(ea_{1,1})$ ont été qualifiés de corrects et par application de la définition 7.1, $EfT(ec_1)$ est qualifié correct.

4. Comme les informations correspondantes à $\neg AN(CF_1(2))$, $EfT(ec_1)$, $Cd(ec_1)$, $Ct(ec_1)$, $PCd(ea_{1,1})$ et $Ct(ea_{1,1})$ ont été qualifiées de correctes et par application de la définition 7.2, $EfF(ec_1)$ est qualifié correct.

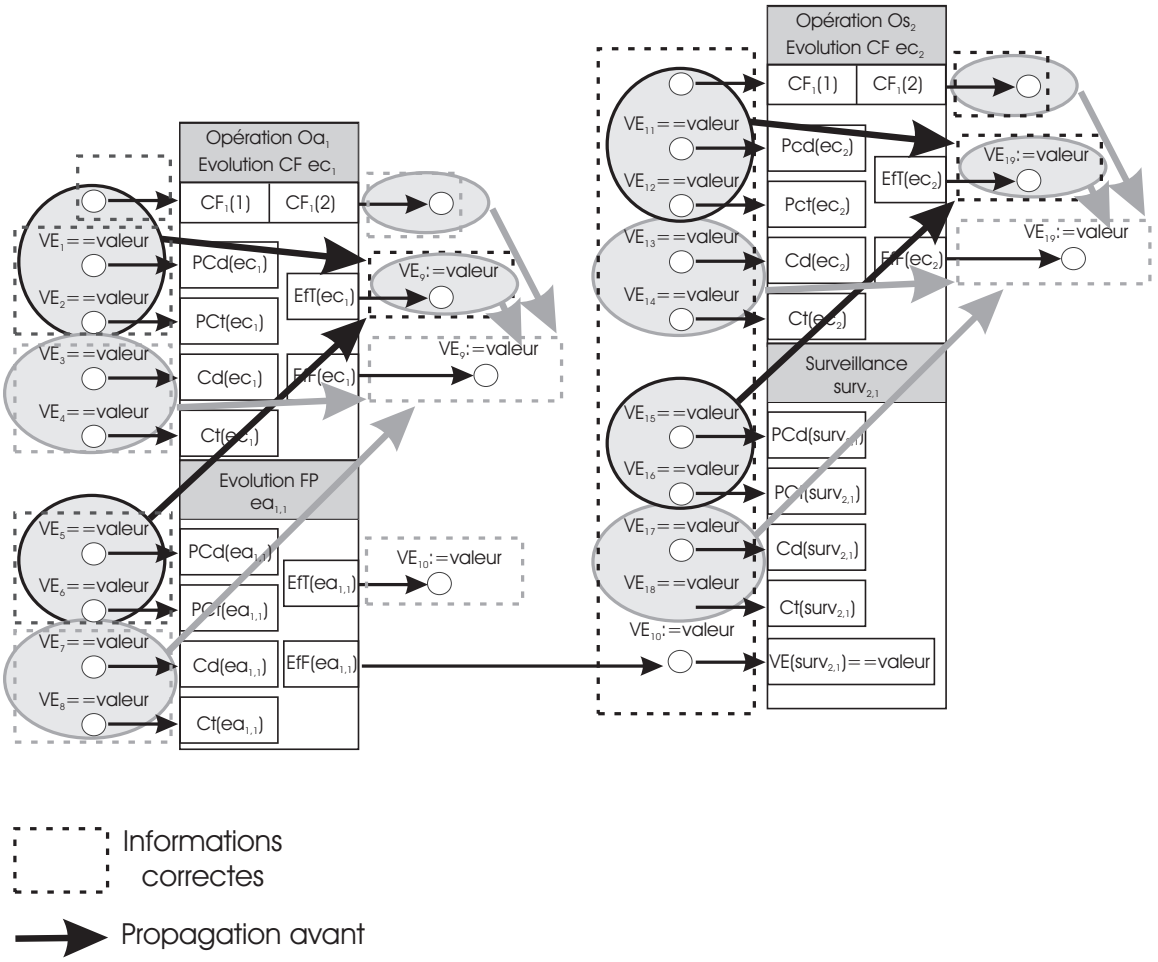


FIG. 7.3 – Propagation avant de la confiance accordée aux informations

Une propagation par l'arrière puis une propagation vers l'avant sont tout à fait suffisantes pour couvrir l'ensemble des cas. La propagation par l'arrière permet de mettre en évidence l'ensemble des pré-requis qui doivent être corrects pour que l'information corrélée soit correcte. La propagation par l'avant qualifie de corrects les effets dont les pré-requis sont corrects. Il est alors inutile de relancer la propagation arrière puisque celle-ci reviendrait à rechercher les pré-requis qui doivent être corrects pour que ces effets soient corrects, pré-requis qui sont justement ceux qui nous ont permis de qualifier les effets de corrects pendant la propagation avant. L'ensemble des informations correctes est alors supprimé du modèle afin de limiter la recherche du diagnostic, à l'exception des opérations dont les effets sont les derniers à avoir agi sur une variable d'état particulière. Il est en effet indispensable de les conserver car lors de la construction du modèle, un pré-requis est toujours connecté aux derniers effets ayant agi sur les

variables d'état présentes dans le pré-requis. Ainsi, si le dernier effet pour une variable d'état qui est correcte n'est pas conservé, une connexion pourrait être créée sur un effet antérieur à l'effet qualifié de correct et biaiser ainsi le résultat de diagnostic. Prenons l'exemple d'un pré-requis d'une opération qui est exécuté à une date $t3$ portant sur l'absence d'un produit à une position donnée. Considérons que l'absence du produit (induit par un effet) a été qualifiée de correcte à une date $t2$ (antérieure à $t3$) mais pas à une date $t1$ (autre effet antérieur à $t2$). Si l'effet ayant induit l'absence du produit à la date $t2$ est supprimé du modèle, le pré-requis portant sur l'absence du produit de l'opération exécuté à la date $t3$ sera connecté à l'effet réalisé à la date $t1$. Ceci signifie que le pré-requis pourra être, à tort, remis en cause car l'information correspondante à l'absence de produit à bien été qualifiée de correcte à la date $t2$.

3.3 État initial de la partie opérative et du flux de produits

L'état initial de la partie opérative et du flux de produits au démarrage du SAP est a priori connu. Toutefois, si aucune certitude ne peut être accordée quant aux valeurs des variables d'état le caractérisant, elles doivent également pouvoir être remises en cause suite à une analyse diagnostique. Il est donc important de le représenter. Pour des raisons de simplification algorithmique nous avons décidé de le représenter comme une opération ne possédant qu'un ensemble d'effets, chacun correspondant à l'affectation des valeurs des variables d'état. Il en est de même pour l'arrivée de nouveaux produits dans le sous système piloté. En sus, ces produits ont éventuellement subi un ensemble de transformations en amont du sous-système piloté. Certaines de ces transformations ont pu être qualifiées de correctes, et donc au moment de l'arrivée d'un nouveau produit dans le sous-système piloté, l'ensemble de ces informations doit être fourni par le module de coordination pilotant le sous-système piloté dont le produit est issu. Par extension, lorsqu'un produit sort du sous-système piloté, le module de coordination doit transmettre ces mêmes informations. Ces principes devront à terme être pris en compte dans l'approche de diagnostic.

3.4 Algorithme de réduction

A l'arrivée d'un compte rendu d'exécution correct d'une opération, si un indice de confiance de 1 est accordé, soit à un effet, soit à l'exécution d'une opération de surveillance, le processus de réduction représenté dans l'algorithme 3 sera lancé.

3.5 Convergence de l'algorithme de réduction du modèle et temps de calcul

La nature même du modèle assure une convergence de l'algorithme. Les pré-requis associés à un effet sont forcément connectés à des effets antérieurs, assurant ainsi l'absence de "bouclages" dans le modèle. La propagation ne se faisant que dans un sens puis dans l'autre ne peut en aucun cas se retrouver bloquée dans une boucle durant une propagation.

Le temps de calcul des algorithmes proposés est en partie fonction de leur complexité. Les mécanismes de propagation avant et arrière sont de très faible complexité puisque le nombre d'itérations est simplement proportionnel au nombre d'opérations. Si le modèle pour le diagnostic est constitué de n effets (sur les chaînes fonctionnelles et sur le flux de produits), le mécanisme de réduction du modèle propagera la confiance des effets vers les pré-requis (et vice versa), au

Qualifier correct les effets (EfT et/ou EfF) dont l'IC est de 1
$liste := EfT$ et/ou EfF dont l'IC est de 1
//propagation arrière ://
Répéter
  Rechercher les pré-requis de l'effet le plus récent contenu dans *liste* en utilisant la définition de son comportement et retirer l'effet de la liste
  Rechercher les connexions amont des pré-requis à des effets EfT ou EfF ou à des non effets $\neg EfT$ ou $\neg EfF$ et qualifier ces effets de corrects
  Ajouter les EfT ou EfF obtenus à *liste*
jusqu'à ce que (liste $= \emptyset$)
//propagation avant ://
Pour tous les EfT et les EfF dans l'ordre d'enregistrement **faire**
  Rechercher les pré-requis de l'effet en utilisant sa définition
  Si (tous les EfT, EfF, $\neg EfT$ et $\neg EfF$ connectés en amont de tous les pré-requis sont corrects) **Alors**
    l'effet est qualifié de correct
  Fin Si
Fin Pour
Supprimer tous les effets corrects qui ne sont pas les derniers à agir sur une variable d'état particulière. Supprimer les opérations dont tous les effets ont été supprimés

Alg. 3: Réduction du modèle

plus une fois par effet. Le nombre maximum d'itérations de l'algorithme sera donc égale à $2n$. La complexité de l'algorithme est donc polynomiale en $\mathcal{O}(n)$. Ceci permet de garantir l'efficacité de l'algorithme d'un point de vue temps de calcul (Garay, 1979).

4 Conclusion

Ce chapitre s'est attaché à présenter les mécanismes de génération et de réduction du modèle pour le diagnostic. Nous avons tout d'abord montré, par une étude de complexité, qu'il n'était pas réaliste de générer le modèle hors ligne. Une fonction suivi a ainsi été introduite afin de générer et de réduire le modèle au fur et à mesure de l'exécution de la loi de commande.

La génération du modèle est basée d'une part sur une traduction des pré-requis, afin de n'enregistrer que les informations pertinentes pour le diagnostic, et d'autre part sur la connexion des tests conservés dans ces pré-requis aux effets des opérations ; ceci afin de permettre la propagation de la confiance accordée aux effets des opérations et la recherche des origines d'un symptôme de défaillances et de leurs conséquences.

La réduction du modèle est quant à elle basée sur l'exploitation conjointe des définitions des comportements des opérations et des indices de confiance. Le mécanisme de réduction propage alors la confiance par les connexions créées entre les pré-requis et les effets des opérations.

Ainsi l'ensemble des mécanismes mis en place permet in fine d'obtenir un modèle du fonctionnement passé incluant le lien entre les différentes opérations exécutées. L'exploitation de ce modèle est présentée dans le chapitre suivant.

Chapitre 8

Diagnostic de services

1 Introduction

A cette étape de l'étude, les mécanismes proposés permettent, en ligne, de générer et de gérer un modèle structuré des évolutions passées.

Sur cette base, ce chapitre se propose de développer la méthode de diagnostic que nous proposons, permettant de mettre en exergue suite à l'occurrence d'un dysfonctionnement avéré de la partie opérative, quelles sont les opérations exécutées qui doivent être suspectées d'être à l'origine de ce dysfonctionnement ainsi que celles qui ont pu être affectées.

Les mécanismes généraux proposés portent sur des techniques de propagation arrière et avant de la suspicion dans le modèle de fonctionnement passé. Ils font l'objet des paragraphes suivants.

2 Principes

2.1 Liens entre la suspicion des opérations et le diagnostic logique

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. Une deuxième phase en diagnostic logique est de déterminer les diagnostics minimaux à partir des conflits minimaux. Un diagnostic minimal correspond à un ensemble de composants défaillants expliquant toutes les observations. Dans le cadre de la fonction diagnostic développée pour la reconfiguration du système de commande, il n'est pas nécessaire de connaître les diagnostics minimaux dans le sens où la suspicion d'une opération implique qu'elle ne doit pas être utilisée dans une nouvelle loi de commande.

2.2 Obtention des opérations suspectes

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 évoluent, 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. Dans ce cas, il n'est pas nécessaire de suspecter les pré-conditions et conditions 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 et pour finir sur la disponibilité future de l'ensemble des opérations. Cette recherche s'appuiera sur les modèles des comportements des opérations exécutées, présentés dans les deux chapitres précédents.

La recherche des origines d'un CRA est assurée par une propagation arrière. Partant de ce 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 (cf. section 3.2 page 31). En revanche, la recherche des autres conséquences possibles est quant à elle assurée par un mécanisme de propagation avant. Partant des opérations dont l'exécution n'est potentiellement pas conforme, il s'agit de retrouver toutes les conséquences possibles. La réalisation d'une propagation arrière suivie d'une propagation avant est suffisante pour traiter tous les cas. En effet la propagation arrière permet d'obtenir l'ensemble des origines possibles d'un CRA, et la propagation avant l'ensemble des conséquences. Relancer une propagation arrière consisterait en quelque sorte à expliquer les conséquences dont on connaît déjà les origines. En résumé, la fonction suivi fournit un modèle pour le diagnostic contenant des informations qui sont a priori non suspectées. Suite à la réception d'un CRA, le processus de diagnostic qualifiera soit de suspectes soit de non suspectes ces informations vis-à-vis du compte rendu reçu.

3 Propagation arrière

3.1 Principe

Comme expliqué à la section 5.1.2 page 25, un CRA peut être renvoyé par une chaîne fonctionnelle dans deux cas distincts :

1. l'impossibilité pour une chaîne fonctionnelle de réaliser un service,
2. suite au changement d'état non attendu d'une chaîne d'acquisition portant sur le flux de produits.

Un troisième cas à prendre en compte est l'envoi par un autre module de niveau coordination de la suspicion d'un effet d'une opération correspondant à une opération requise pour un autre module. Ce cas est tout à fait similaire à la réception d'un compte-rendu de mauvaise exécution d'une opération, dans le sens où il suffira de suspecter l'effet de cette opération puis de lancer la propagation arrière.

3.1.1 Propagation suite à la mauvaise réalisation d'une opération

Dans le premier cas la méthode de propagation de la suspicion est très similaire à la méthode de propagation de la confiance au sein du modèle. En effet, de la même façon elle va utiliser les définitions des comportements des opérations afin de propager la suspicion (i.e. retrouver les différentes propagations de défaillances possibles). Afin de ne pas surcharger l'explication de la méthode et l'écriture des algorithmes, il sera implicitement sous-entendu qu'aucune propagation de la suspicion ne sera faite sur un effet qui a été qualifié de correct et qui est encore contenu dans le modèle. Au cours de la propagation arrière, six cas différents sont à distinguer (cf. Figure 8.1) :

- Suite à la suspicion de l'effet transitoire sur la chaîne fonctionnelle $EfT(ec_i)$. En s'appuyant sur la définition (6.1), l'origine de la non réalisation de cet effet peut provenir de la non satisfaction de la pré-condition $PCd(ec_i)$ et de la pré-contrainte $PCt(ec_i)$ de l'évolution de la chaîne fonctionnelle ec_i, d'une indisponibilité de la chaîne fonctionnelle au moment du lancement de l'opération ($\neg AN(CF_k(2x-1))$, ou de la transgression d'une pré-contrainte d'une évolution $ea_{i,j}$. Cette dernière peut être la transgression d'une pré-contrainte d'une évolution attendue $ea_{i,j}$ pour $j \in J$ ou d'une évolution non attendue $ea_{i,j}$ pour $j \notin J$. Toutefois pour qu'une contrainte soit transgressée, il est nécessaire que l'évolution correspondante soit lancée. Ainsi les pré-conditions des évolutions non attendues seront suspectées, à la condition qu'une pré-contrainte existe. De plus seront suspectées les contraintes satisfaites de ces évolutions qui sont susceptibles d'avoir été transgressées.

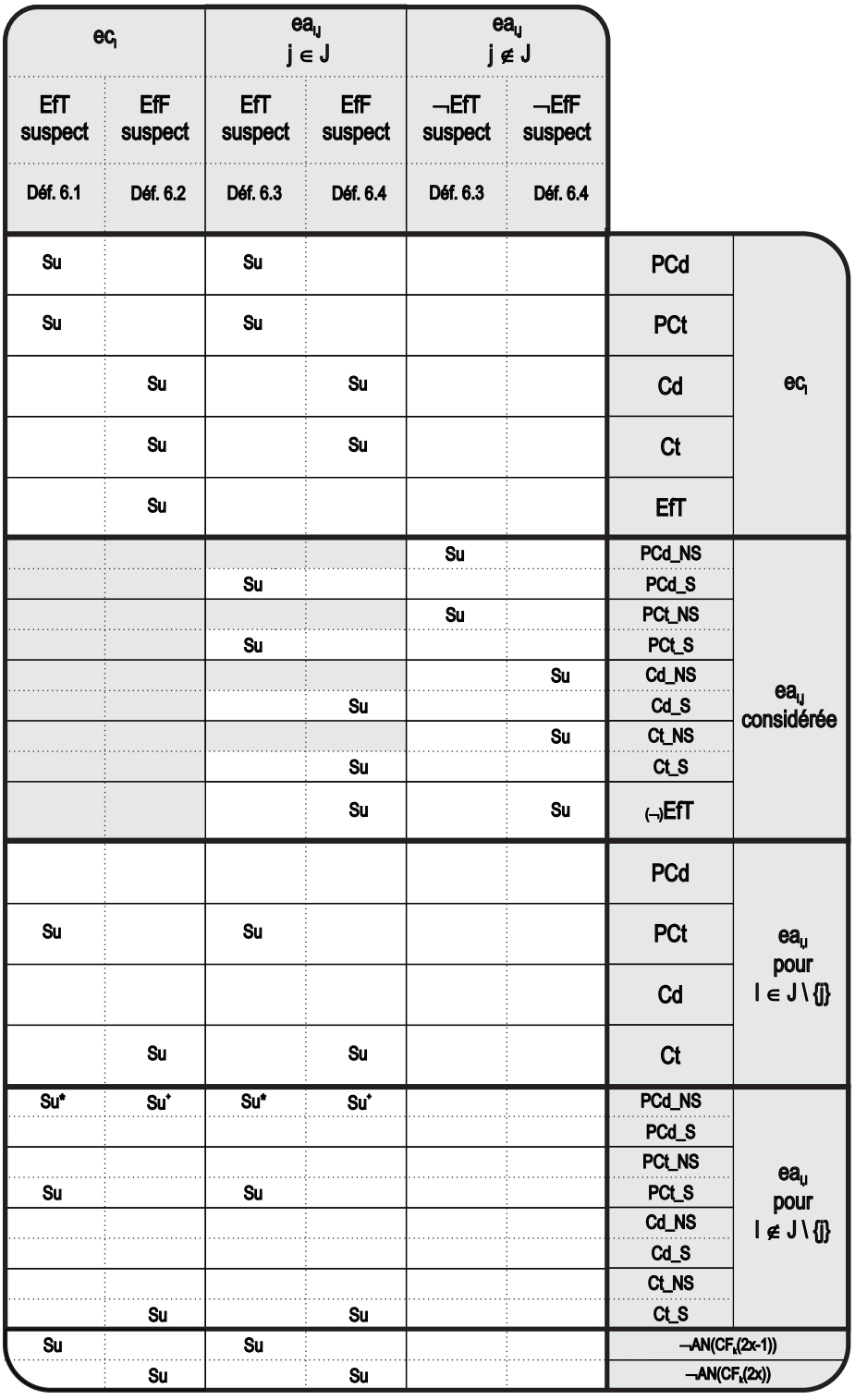
- La propagation suite à la suspicion de l'effet final sur la chaîne fonctionnelle $EfF(ec_i)$ est très similaire à la précédente et s'appuiera sur la définition (6.2).

- Suite à la suspicion de l'effet transitoire $EfT(ea_{i,j})$ d'une évolution du flux de produits $ea_{i,j}$. En s'appuyant sur la définition (6.3), l'origine de la non réalisation de cet effet peut provenir de la non satisfaction de la pré-condition $PCd(ea_{i,j})$ et de la pré-contrainte $PCt(ea_{i,j})$ de l'évolution du flux de produits $ea_{i,j}$, d'une indisponibilité de la chaîne fonctionnelle au moment du lancement de l'opération ($\neg AN(CF_k(2x-1))$), ou de la transgression d'une pré-contrainte d'une autre évolution du flux de produits $ea_{i,l}$.

- La propagation suite à la suspicion de l'effet final $EfF(ea_{i,j})$ d'une évolution du flux de produits est très similaire à la précédente et s'appuiera sur la définition (6.4).

- Suite à la suspicion d'un effet transitoire non attendu du flux de produits d'une opération exécutée $\neg EfT(ea_{i,j})$. Dans ce cas il est nécessaire de suspecter en amont la partie de la pré-condition et de la pré-contrainte non satisfaites par application de la définition (6.4).

- La propagation suite à la suspicion d'un effet final non attendu du flux de produits $\neg EfF(ea_{i,j})$ est similaire au dernier cas. Il est cependant à noter que pour réaliser l'effet final, l'effet transitoire doit être réalisé, d'où la suspicion de $\neg EfT(ea_{i,j})$.



Su : suspect
Su* : suspect si $PCt\_S(ea_{ij}) \neq \emptyset$ OU $PCt\_NS(ea_{ij}) \neq \emptyset$
Su$^+$: suspect si $Ct\_S(ea_{ij}) \neq \emptyset$ OU $Ct\_NS(ea_{ij}) \neq \emptyset$

FIG. 8.1 – Règles de la propagation arrière de la suspicion

Il est à noter que la suspicion d'une expression logique contenue dans une pré-condition, condition, pré-contrainte ou contrainte se traduit par la suspicion de tous les tests sur les variables d'état. Ceci est tout à fait normal car nous ne savons pas a priori quelle partie de l'expression est fausse (ou vraie).

Un deuxième point à noter est que si un effet d'une opération requise est suspecté, l'information doit être transmise au niveau supérieur.

3.1.2 Propagation suite au changement d'état non prévu du flux de produits

Comme nous l'avons vu à la section 4.1 du chapitre 4, une opération induite peut être déclenchée intempestivement suite à une propagation de défaillance. Ceci signifie donc que cette opération induite n'apparaît pas dans le modèle pour le diagnostic dans le sens où l'opération n'était pas prévue dans la loi de commande. Toutefois, l'exécution intempestive d'une opération induite entraîne un changement d'état non prévu du flux de produits. Lorsqu'un changement d'état non prévu du flux de produits est observé par une chaîne d'acquisition, elle en informe le niveau coordination par l'envoi d'un compte rendu de son observation (cf. section 5.1.2 page 25). Suite à la réception d'un tel compte rendu, le diagnostic doit être en mesure de retrouver les évolutions non souhaitées d'opérations induites qui auraient pu conduire au changement d'état du flux de produits observé par la chaîne d'acquisition. Dans ce contexte le diagnostic devra également prendre en compte une possible défaillance de la chaîne d'acquisition.

Captage de l'information du changement d'état du flux de produits

En premier lieu le diagnostic doit s'intéresser à la chaîne d'acquisition ayant fourni l'information sur le changement d'état du flux de produits. Ses capacités à capter de l'information sur le produit est traduite au niveau coordination par une opération de surveillance. Ainsi la première opération à ajouter au modèle est celle de surveillance, si ce n'est pour vérifier les pré-requis à la surveillance de la variable d'état (cf. section 4.3 page 62). Cette opération sera donc ajoutée avec ses pré-requis et les connexions qui les lient aux effets antérieurs comme décrit à la section 2.4 du chapitre 7. Ainsi, les mécanismes de propagation pourront prendre en compte le contexte dans lequel le captage de l'information sur le flux de produits a été fait, et donc les conséquences éventuelles sur cette chaîne d'acquisition de la modification de l'état du flux de produits observé.

Opération induite ayant conduit au changement d'état observé

Le diagnostic, dans un second temps, doit être capable à partir de la variable d'état dont le changement de valeur n'était pas prévu, de rechercher l'ensemble des opérations induites à l'origine possible de ce changement. Il est d'ailleurs tout à fait possible que ce changement soit la conséquence de plusieurs opérations induites s'étant exécutées à la suite.

Le principe de recherche des opérations induites est le suivant :

- Recherche des opérations induites qui ont pu conduire à la valeur observée de la variable d'état. Ceci nous donne donc des pré-requis en terme de valeurs de variables d'état pour le lancement de ces opérations induites. Une comparaison est faite entre la variable d'état caractérisant le produit avant l'opération et celles présentes dans le modèle.

- Si cette variable d'état est identique à une du modèle (non seulement au niveau de sa valeur mais aussi à la date où cette valeur est prise) alors l'opération induite peut expliquer l'observation faite par la chaîne d'acquisition. Elle pourra donc être ajoutée et connectée au modèle.

- S'il n'y a pas de correspondance sur une variable, la recherche d'opérations induites continue à partir de cette nouvelle variable d'état.

Ainsi, la recherche d'opérations induites va construire des séquences d'opérations induites afin d'expliquer l'observation du changement non prévu de l'état du flux de produits. Pour chacune des séquences cette recherche s'arrête lorsqu'il n'est plus possible de connecter la séquence au modèle. En d'autres mots lorsqu'il n'y a plus d'effet antérieur (portant sur le flux de produits) à la dernière opération induite obtenue dans une séquence, la séquence est supprimée de la recherche.

Cette reconstruction se traduit par l'algorithme suivant en partant de la variable d'état et de la valeur observée $VE == valeur$:

Rechercher Opération de surveillance Os_i correspondant à $VE == valeur$
$op\_1 := (Os_i, J)$
$VE\_1 := (VE, valeur)$
$m := 1$
Répéter
  Pour pour chaque VE_indice créée **faire**
    Rechercher Opérations induites telles qu' un effet final EFT correspond à la valeur et à la variable d'état de VE_indice
    Pour Pour chaque *opération* obtenus **faire**
      $m = m + 1$
      $op\_indice, m := (opération\ m, J)$
      $VE\_indices, m := (VE, valeur\ pré\text{-}requis\ à\ l'opération\ m)$
      Si (la VE et sa valeur caractérisant le produit avant l'opération correspond à une VE du modèle(valeur&date)) **Alors**
        Ajouter_Séquence$(op\_indices, m)$
      Fin Si
      Si (si il n'y a pas d'effet antérieur à l'opération m) **Alors**
        Supprimer la séquence $op\_indices, m$
      Fin Si
    Fin Pour
  Fin Pour
jusqu'à ce que (aucune VE_indice ne soit créé)

Alg. 4: Recherche d'opérations induites non souhaitées

L'algorithme d'ajout de la séquence n'est pas présenté ici dans le sens où la méthode est identique à l'ajout d'opérations classiques présenté dans le chapitre précédent. Suite à cet ajout dans le modèle, le mécanisme de propagation est le même que dans le cas de la mauvaise réalisation d'une opération.

3.1.3 Exploitation des liens de causalité entre les opérations pour la propagation

Suite à la suspicion d'un pré-requis à la réalisation d'une opération, le diagnostic va utiliser les connexions de ce pré-requis à l'effet antérieur lui ayant donné sa valeur, ou aux effets non attendus qui auraient pu modifier sa valeur pour les suspecter. Toutefois, il est non seulement nécessaire que le diagnostic s'intéresse aux origines en termes d'exécutions non conformes des opérations mais également en termes d'évolutions non attendues de l'état des chaînes fonctionnelles ou du flux de produits non liées à l'exécution d'une opération (cf. section 3.2 page 31).

Recherche d'évolutions non attendues du flux de produits

Lors de la suspicion d'un pré-requis portant sur une variable d'état du flux de produits, le diagnostic doit de la même manière que dans la section précédente rechercher les évolutions du flux de produits issues d'opérations induites non attendues (i.e. non présentes dans le modèle) qui auraient pu conduire au non respect du pré-requis. La méthode est identique à celle développée dans la section 3.1.2 à l'exception de deux points :

- le point de départ de la recherche réside cette fois-ci dans les valeurs de la variable d'état transgressant le pré-requis,

- la connexion de la séquence obtenue ne dépend pas d'une date, mais d'un intervalle. En effet dans la section 3.1.2, la date du changement non prévu de l'état du flux de produits est connue. Dans le cadre du non respect d'un pré-requis, la valeur de la variable d'état du flux de produits transgressant ce pré-requis a pu être modifiée entre deux dates : la date du dernier effet ayant agi sur cette variable et ayant été qualifiée de correcte, et la date de l'effet dont le pré-requis est suspecté.

Modification de l'état d'une chaîne fonctionnelle

Le non respect d'un pré-requis peut être la conséquence du changement intempestif de l'état d'une chaîne fonctionnelle, par exemple un vérin simple effet qui passerait de la position sortie à la position rentrée suite à la rupture d'un flexible pneumatique. Même si le dernier effet ayant agi sur le vérin a été qualifié de correct, il n'en demeure pas moins que sa position a pu être modifiée depuis. Afin de prendre en compte ce cas dans la propagation arrière mais également les autres conséquences possibles de ce changement d'état intempestif dans la propagation avant, nous proposons de rajouter au modèle un ensemble d'effets non souhaités (donnant comme valeurs celles qui transgressent le pré-requis suspecté) durant la propagation arrière et dont le seul pré-requis est le fonctionnement anormal de la chaîne fonctionnelle dont la variable d'état a pu changer ($AN(CF_k)$). Ces effets non souhaités seront également reliés à tous les pré-requis portant sur cette variable d'état et qui pourraient être transgressés. Cet effet sera daté à la même date que le dernier effet ayant été qualifié de correct, dans le sens où ce changement de variable d'état n'a pas pu avoir lieu avant.

3.2 Algorithme

Le mécanisme de propagation arrière est lancé suite à :

- la réception d'un compte rendu d'une chaîne fonctionnelle suite à l'impossibilité d'exécuter un service. Ce compte rendu ayant été élaboré à partir d'une observation sur une chaîne d'action (cf. section 5.1.3 page 26), l'effet final de l'évolution de la chaîne fonctionnelle correspondante est suspecté ($EfF(ec_i) = suspect$).

- La réception de la part d'une chaîne d'acquisition d'un compte rendu traduisant une incohérence entre l'état du flux de produits attendu et celui observé. Dans ce cas $EfF(Surv_{i,l})$ doit être suspecté (cf. section 4 page 71).

- La réception du niveau supérieur de la mauvaise réalisation d'un effet d'une opération correspondant à une opération requise d'un autre module de coordination. Dans ce cas l'effet en question est suspecté.

La propagation arrière se formalise au travers de l'algorithme suivant :

liste := Effet suspecté
Répéter
  Rechercher les pré-requis à suspecter de l'effet le plus récent contenu dans *liste* en utilisant les définitions des comportements des opérations (cf. Figure 8.1 page 96)
  Retirer l'effet de la liste
  Qualifier de suspects les effets EfT ou EfF ou les effets non attendus $\neg EfT$ ou $\neg EfF$ connectés en amont des pré-requis obtenus
  Pour chaque pré-requis portant sur l'état d'une chaîne fonctionnelle **faire**
    Ajout du changement intempestif de l'état de la chaîne fonctionnelle
  Fin Pour
  Pour chaque pré-requis portant sur le flux de produits **faire**
    Recherche d'opérations induites non attendues
  Fin Pour
  Si (un effet appartient à une opération requise) **Alors**
    transmettre au niveau supérieur
  Fin Si
  Ajouter les EfT ou EfF ou $\neg EfT$ ou $\neg EfF$ obtenus à *liste*
jusqu'à ce que (liste = $\emptyset$)

Alg. 5: Propagation arrière de la suspicion

Suite à la propagation arrière, l'ensemble des origines possibles d'un CRA est suspecté. Le mécanisme de propagation prend en compte la propagation de défaillances multiples (cf. section 3.4 page 32). Prenons par exemple le cas d'une propagation due au non respect d'une contrainte d'une évolution $ea_{i,j}$ non attendue du flux de produits ($j \notin J_{k,x}$). Considérons que même si l'évolution du flux de produits n'est pas attendue, la contrainte $Ct(ea_{i,j})$ soit satisfaite (i.e. $Ct\_NS(ea_{i,j}) = \emptyset$). Deux défaillances, l'une entraînant la satisfaction de la pré-condition

$PCd\_NS(ea_{i,j})$, et une autre entraînant la non satisfaction de la pré-contrainte $Ct\_S(ea_{i,j})$ sont par exemple des origines prises en compte pour l'exécution non conforme de l'effet final de l'évolution de la chaîne fonctionnelle ec_i.

4 Propagation avant

4.1 Principe

La propagation arrière a permis de suspecter l'ensemble des origines possibles en terme d'exécutions non conformes des opérations présentes dans le modèle pour le diagnostic. En accord avec les principes proposés dans la section 3 du chapitre 2, la propagation avant va quant à elle rechercher et suspecter l'ensemble des conséquences possibles de ces exécutions non conformes en recherchant les propagations multiples d'une défaillance. De la même manière que la propagation arrière, la propagation avant est basée sur l'exploitation des définitions des comportements des opérations. La figure 8.2 résume l'ensemble des règles de propagation permettant de déterminer les conséquences sur chacun des effets et sur la bonne exécution des opérations, en fonction de la suspicion des pré-requis de l'opération.

Nous retrouvons des similitudes avec les règles de propagation arrière, toutefois elles se distinguent sur deux points :

- ces règles prennent en compte les définitions relatives à la conséquence du non-respect des pré-contraintes et contraintes mais également les liens entre la disponibilité des opérations d'une même chaîne fonctionnelle (Définitions 6.5, 6.6 et 6.7),

- le terme suspect utilisé pour les pré-requis ne se réfère plus à une liste de variables d'état à suspecter comme dans la propagation arrière, mais cette fois-ci, à l'impact d'un pré-requis suspect sur les effets. Il est donc nécessaire à partir de la suspicion d'un test dans une expression logique de définir dans quel cas l'expression est elle-même suspecte.

La suspicion de l'expression d'un pré-requis sera établie en utilisant les tables de vérité données ci-dessous. Elles correspondent aux tables de vérité classiques des conjonctions et des disjonctions, où le terme suspect correspond soit à faux pour les expressions satisfaites, ou à vrai pour les expressions non satisfaites. Ceci permet de savoir si une expression satisfaite est suspecte (i.e. potentiellement non satisfaite) ou si une expression non satisfaite est suspecte (i.e. potentiellement satisfaite).

		ec_i		ea_{ij} $j \in J$		ea_{ij} $j \notin J$		$\neg AN(CF_k(2x-1))$	$\neg AN(CF_k(2x))$	$\neg AN(CF_k(y))$ pour $y \geq 2x$
		EfT	EfF	EfT	EfF	¬EfT	¬EfF			
		Déf. 6.1	Déf. 6.2	Déf. 6.3	Déf. 6.4	Déf. 6.3	Déf. 6.4	Déf. 6.5	Déf. 6.6	Déf. 6.7
ec_i	PCd suspecte	Su		Su						
	PCt suspecte	Su		Su				Su		Su
	Cd suspecte		Su		Su					
	Ct suspecte		Su		Su				Su	Su
	EfT suspecte		Su							
ea_{ij} considéré pour $i \in J$	PCd suspecte			Su						
	PCt suspecte			Su				Su		Su
	Cd suspecte				Su					
	Ct suspecte				Su				Su	Su
	EfT suspecte				Su					
ea_{ij} considéré pour $i \notin J$	PCd_NS suspecte									
	PCd_S suspecte									
	PCt_NS existe					Su*		Su*		Su*
	PCt_S suspecte							Su*		Su*
	Cd_NS suspecte									
	Cd_S suspecte									
	Ct_NS existe								Su*	Su*
	Ct_S suspecte								Su*	Su*
	(¬)EfT suspecte						Su+			
ea_{ij} pour $i \in J \setminus \{j\}$	PCd suspecte									
	PCt suspecte	Su		Su				Su		Su
	Cd suspecte									
	Ct suspecte		Su		Su				Su	Su
ea_{ij} pour $i \notin J \setminus \{j\}$	PCd_NS suspecte									
	PCd_S suspecte									
	PCt_NS existe	Su*		Su*				Su*		Su*
	PCt_S suspecte	Su*		Su*				Su*		Su*
	Cd_NS suspecte									
	Cd_S suspecte									
	Ct_NS existe		Su*		Su*				Su*	Su*
	Ct_S suspecte		Su*		Su*				Su*	Su*
$\neg AN(CF_k(2x-1))$ suspect		Su		Su					Su	Su
$\neg AN(CF_k(2x))$ suspect			Su		Su					Su

Su : suspect
Su* : suspect si $PCd\_NS(ea_{ij})$ = Su
Su+ : suspect si $Cd\_NS(ea_{ij})$ = Su et $Ct\_NS(ea_{ij})$ = Su

FIG. 8.2 – Règles de propagation avant de la suspicion

Pour les expressions satisfaites :

A	B	A ∨ B
vrai	vrai	vrai
vrai	suspect	vrai
suspect	vrai	vrai
suspect	suspect	suspect

A	B	A ∧ B
vrai	vrai	vrai
vrai	suspect	suspect
suspect	vrai	suspect
suspect	suspect	suspect

Pour les expressions non satisfaites :

A	B	A ∨ B
faux	faux	faux
faux	suspect	suspect
suspect	faux	suspect
suspect	suspect	suspect

A	B	A ∧ B
faux	faux	faux
faux	suspect	faux
suspect	faux	faux
suspect	suspect	suspect

Nous devons noter que dans le mécanisme de propagation, il est important de traiter dans l'ordre les conséquences sur les effets du plus ancien vers le plus récent. Un effet peut être le pré-requis à un effet plus ancien, et donc en cas de suspicion, peut modifier la règle de propagation du plus ancien. Enfin, tout comme la propagation arrière, la propagation avant doit s'intéresser au cas des opérations induites dont l'exécution non attendue (et donc non contenue dans le modèle pour le diagnostic) pourrait être une conséquence non encore mise en exergue des opérations mal exécutées.

4.2 Cas de propagation par des opérations induites non prévues

Dans la recherche des propagations multiples par le diagnostic, les évolutions d'opérations induites non prévues doivent être prises en compte. Toutefois ces évolutions non souhaitées du flux de produits ne sont pas présentes dans le modèle pour le diagnostic (cf. section 2.2.5 page 81). Le déclenchement intempestif d'une opération induite est obtenu via la satisfaction non prévue de sa pré-condition. Si à un moment donné, une pré-condition d'une évolution du flux de produits d'une opération induite est suspecte, alors les évolutions non souhaitées de l'opération induite sont ajoutées dans le modèle. Pendant la phase de propagation avant, à chaque fois qu'un effet est suspecté, l'algorithme de recherche et d'ajout d'opérations induites sera lancé. Ainsi, les effets non souhaités de ces opérations induites pourront être traités comme tout autres effets du modèle suite à leurs ajouts. La fonction utilisée peut être traduite de la manière suivante :

Fonction RechercheEffetOind($Effet\_suspect$,$date$) :
- **Pour** chaque opération induite dont une pré-condition est modifiée par un effet de $Effet\_suspect$ **faire**
  - Rechercher si une pré-condition $PCd(ea_{i,j})$ d'une évolution de l'opération induite $Oind_i$ est suspecte entre $date$ et l'instant courant.
  - **Si** (une pré-condition des évolutions est suspectée) **Alors**
    - Ajouter l'opération $Oind_i$ au modèle
    - $début(Oind_i) := date$ *ou* $PCd(ea_{i,j}) = suspect$
    - $fin(O_ind) := début(Oind_i) + Du(Oind_i)$
    - **Pour** $j = 1..N_i$ **faire**
      - Ajouter évolution $ea_{i,j}$
      - Ajouter $PCd\_NS(ea_{i,j})$, $PCt\_NS(ea_{i,j})$, $PCt\_S(ea_{i,j})$, $\neg EfT(ea_{i,j})$
      - Rechercher les derniers EfF portant sur $PCd\_NS(ea_{i,j})$ ajouter aux connexions
      - Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCd\_NS(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les derniers EfF portant sur $PCt\_S(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCt\_S(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les derniers EfF portant sur $PCt\_NS(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $PCt\_NS(ea_{i,j})$ et ajouter aux connexions Ajouter $Cd\_NS(ea_{i,j})$, $Ct\_NS(ea_{i,j})$, $Ct\_S(ea_{i,j})$, $\neg EfF(ea_{i,j})$
      - Rechercher les derniers EfF portant sur $Cd\_NS(ea_{i,j})$ ajouter aux connexions
      - Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Cd\_NS(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les derniers EfF portant sur $Ct\_S(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Ct\_S(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les derniers EfF portant sur $Ct\_NS(ea_{i,j})$ et ajouter aux connexions
      - Rechercher les $\neg EfT$ et $\neg EfF$ ultérieurs portant sur $Ct\_NS(ea_{i,j})$ et ajouter aux connexions
    - **Fin Pour**
    - **Retourner** ($\neg EfT(ea_{i,j})$ pour $j = 1..N_i$ de $Oind_i$)
  - **Fin Si**
- **Fin Pour**
- **Retourner** ()

Fin

Alg. 6: Recherche du lancement intempestif d'une opération induite

4.3 Conséquence du non respect d'une pré-contrainte ou contrainte sur l'environnement de la chaîne fonctionnelle

Un dernier point reste à prendre en compte. Il s'agit de la conséquence du non respect d'une pré-contrainte ou d'une contrainte sur l'environnement d'une chaîne fonctionnelle exécutant une opération. La conséquence n'est pas limitée à la chaîne fonctionnelle transgressant une contrainte mais peut impacter les chaînes fonctionnelles environnantes ainsi que le flux de produits. Prenons l'exemple de deux vérins orthogonaux ne devant pas être sortis simultanément pour éviter une collision. En cas de transgression de cette contrainte de sécurité, non seulement le premier vérin peut être endommagé mais également le second. Lors de l'analyse diagnostique, il est nécessaire de pouvoir couvrir de telles situations. Toutefois, n'ayant qu'une connaissance du fonctionnement normal de la partie opérative, nous sommes dans l'obligation de prendre en compte le cas le plus défavorable. Ainsi, lors d'une propagation de défaillance, si une contrainte portant sur une variable d'état d'une chaîne fonctionnelle environnante est transgressée, la disponibilité des opérations offertes par cette chaîne fonctionnelle devra être suspectée à partir de la date de transgression de la contrainte. Si la variable d'état porte sur un produit, l'ensemble des valeurs des variables d'état caractérisant le produit doit être suspecté. Pour cela, chacun des derniers effets ayant modifié les variables d'état du produit avant la violation de la contrainte sera suspecté.

L'intégration de ce principe dans le mécanisme de propagation avant permet la prise en compte de l'ensemble des conséquences de la mauvaise exécution d'une opération sur son environnement direct.

4.4 Algorithme

En résumé la propagation avant prend la forme proposée dans l'algorithme suivant :

$liste$:= {l'ensemble des EfT, EfF, $\neg EfT$ et $\neg EfF$ présents dans le modèle}
Répéter
  Rechercher l'effet le plus ancien dans $liste$
  Si (Effet trouvé est un effet transitoire) **Alors**
    Propager la suspicion des pré-requis de l'opération correspondante sur les effets transitoires souhaités ou non souhaités de cette opération, et sur la disponibilité de la chaîne fonctionnelle (cf. Figure 8.2 102)
    $Effets\_suspects := \{effets\ transitoires\ suspects\}$
    $liste := liste \setminus Effet\_suspects$
    Pour chaque transgression d'une pré-contrainte portant sur une autre chaîne fonctionnelle CF_k **faire**
      suspecter les prédicats $\neg AN(CF_k(x))$ pour tous les effets ultérieurs.
    Fin Pour
    Pour chaque transgression d'une pré-contrainte portant sur une variable d'état d'un produit **faire**
      Suspecter chacun des derniers effets ayant modifié les variables d'état du produit avant la violation de la contrainte
    Fin Pour
  Fin Si
  Si (Effet obtenu est un effet final) **Alors**
    Propager la suspicion des pré-requis de l'opération correspondante sur les effets finaux souhaités ou non souhaités de cette opération, et sur la disponibilité de la chaîne fonctionnelle
    $Effets\_suspects := \{effets\ finals\ suspects\}$
    $liste := liste \setminus Effet\_suspects$
    Pour chaque transgression d'une contrainte portant sur une autre chaîne fonctionnelle CF_k **faire**
      suspecter les prédicats $\neg AN(CF_k(x))$ pour tous les effets ultérieurs.
    Fin Pour
    Pour chaque transgression d'une contrainte portant sur une variable d'état d'un produit **faire**
      Suspecter chacun des derniers effets ayant modifié les variables d'état du produit avant la violation de la contrainte
    Fin Pour
  Fin Si
  Pour Chaque effet contenu dans $liste$ **faire**
    Ajouter **RechercheEffetOind(effet,date effet) à** $liste$
  Fin Pour
jusqu'à ce que ($liste = \emptyset$)

Alg. 7: Propagation avant de la suspicion

5 Propriété des algorithmes

5.1 Convergence

Tout comme le processus de réduction du modèle, la nature même du modèle assure la convergence de l'algorithme. L'absence de "boucles" dans le modèle, et les mécanismes de propagation fait en arrière et avant garantissent l'absence de blocage. Il en est de même pour l'ajout d'opérations induites puisque partant d'un instant donné dans le fonctionnement passé, seulement un nombre fini d'opérations peut être ajouté jusqu'à l'instant courant correspondant à la date de réception d'un CRA.

5.2 Temps de calcul

Les mécanismes de propagation avant et arrière sont de très faible complexité puisque le nombre d'itérations est proportionnel au nombre d'effets présents dans le modèle pour le diagnostic (cf. section 3.5 page 89). Les fonctions de recherche d'opérations induites intempestives sont par contre basées sur la construction d'une arborescence, où chacune des opérations induites ne peut être utilisée qu'une seule fois par chemin. Ce type d'algorithme a une complexité exponentielle (nombre d'itérations en $n!$), avec n le nombre d'opérations induites. Toutefois, de par la décomposition du système de pilotage en une structure hiérarchique et modulaire, le nombre d'opérations induites offertes à un module du niveau de coordination reste limité. Dans l'exemple d'application proposé dans la prochaine partie, l'arborescence des opérations induites qui peut être construite dans le cas le plus défavorable est quant à elle limitée à un seul chemin constitué de deux opérations induites. Le temps de calcul en pratique sera "essentiellement" fonction des mécanismes de propagation arrière et avant, assurant ainsi une bonne efficacité du temps d'exécution de l'algorithme.

6 Conclusion

Ce chapitre nous a conduit à apporter une brique supplémentaire à l'édifice du processus de reconfiguration. Il nous a en effet conduit sur les traces des origines opératoires d'un dysfonctionnement retranscrit par une chaîne fonctionnelle. La recherche de ces origines est basée sur un principe "d'honnêteté" d'analyse de la situation, consistant in fine, durant le parcours du modèle de fonctionnement passé, à propager la suspicion sur les opérations représentées.

Cette propagation s'effectue dans un premier temps en arrière sur l'occurrence d'un compte-rendu traduisant soit l'exécution non conforme d'une opération, soit le changement imprévu de l'état du flux de produits. Ce parcours arrière exploite les définitions encadrant le comportement des opérations, et permet d'obtenir les origines possibles du dysfonctionnement avéré.

Dans un deuxième temps, une propagation avant recherche les conséquences sur l'exécution des opérations, des défaillances possibles à l'origine du dysfonctionnement. Cette recherche exploite non seulement le modèle généré mais prend également en compte les conséquences d'une éventuelle transgression de pré-contraintes ou contraintes sur l'environnement des chaînes fonctionnelles. Enfin les opérations induites intempestivement exécutées sont également

recherchées afin de pouvoir prendre compte toutes les évolutions non prévues du flux de produits.

Forts des résultats obtenus, il est maintenant possible de revenir dans le modèle de partie opérative afin d'y projeter les suspicions mises en exergue et présenter ainsi à une fonction décision/synthèse un modèle honnête de la vision des capacités opératoires que nous pouvons avoir à l'instant t, sans faire appel à un opérateur humain.

Chapitre 9

Exploitation du résultat de diagnostic pour la reconfiguration

1 Introduction

Disposant désormais d'une vision "honnête", dans les temps impartis, du fonctionnement passé de la partie opérative, ce chapitre se propose de présenter le mécanisme de projection des résultats du diagnostic au sein du modèle de partie opérative proposé dans la partie II de ce mémoire. Après quoi, nous proposerons des pistes à suivre pour envisager les prises de décision et la synthèse de nouvelles lois de commande tenant compte des éventuelles pertes de services.

2 Le processus de reconfiguration

Revenons au processus de reconfiguration du système de commande. Suite à la réception d'un CRA, ce processus consiste en la mise en place d'un ensemble de fonctionnalités dans l'objectif de confiner la ou les défaillances à l'origine de ce dysfonctionnement (cf. Figure 9.1). Nous y retrouvons le diagnostic, la décision et la synthèse de loi de commande.

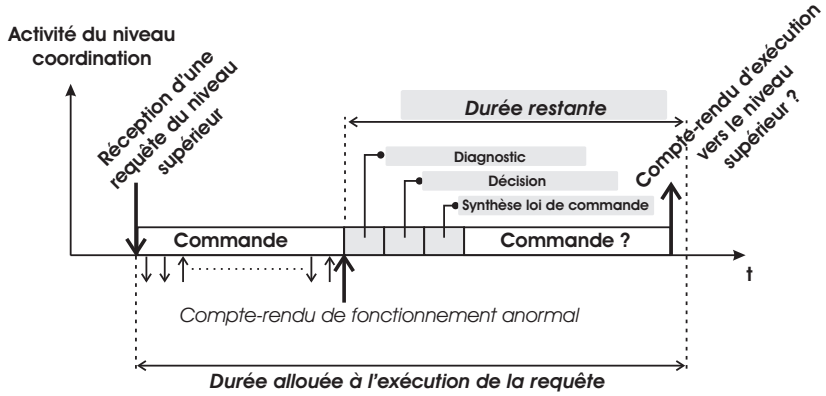


Fig. 9.1 – Fonction mise en place pour la reconfiguration

La fonction diagnostic proposée dans ce manuscrit permet de retrouver dans le fonctionnement passé les origines possibles du CRA. Dans un deuxième temps, le diagnostic

dégage l'ensemble des conséquences potentielles sur le fonctionnement passé du sous-système piloté des possibles défaillances. Ces informations doivent maintenant être exploitées par le reste du processus de reconfiguration. Ce processus exploitant davantage la description des capacités opératoires proposées dans le chapitre 4, il est nécessaire de projeter les suspicions présentes dans le modèle du fonctionnement passé sur le modèle des capacités de la partie opérative.

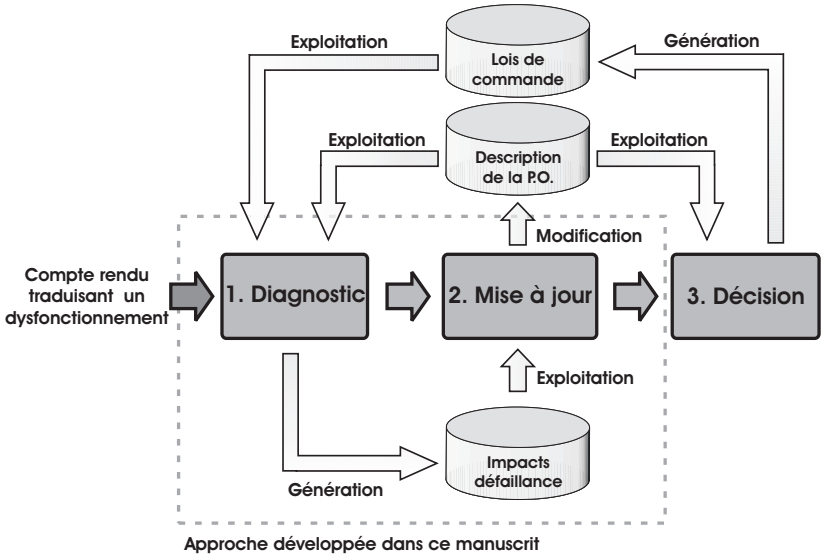


FIG. 9.2 – Principe de la reconfiguration

3 Projection des informations suspectes

3.1 Disponibilité des services

Afin de pouvoir décrire pour chacun des services sa disponibilité, le champ disponibilité est introduit dans les opérations (cf. Figure 9.3), et sera noté Di. Nous devons noter qu'une opération induite n'étant pas liée au lancement d'un service, elle ne possède pas de champ D_i.

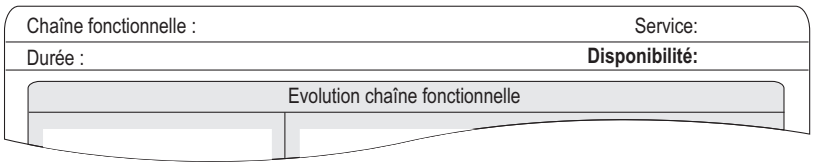


FIG. 9.3 – Champ disponibilité d'une opération

La mise à jour du modèle des capacités opératoires qualifiera la disponibilité d'un service de correcte, non suspecte ou suspecte. Avant de proposer le mécanisme de mise à jour une définition est proposée pour chacun des qualificatifs suivants :

– la disponibilité d'un service est qualifiée de correcte si, lors de son dernier appel, son effet sur la chaîne fonctionnelle offrant ce service a été qualifié de correct,

- la disponibilité d'un service est qualifiée de suspecte si, lors de son dernier appel, l'effet de ce service sur la chaîne fonctionnelle a été qualifié de suspect,
- la disponibilité d'un service est qualifiée de non suspecte, si elle n'a ni été qualifiée de correcte, ni de suspecte.

Nous notons ici une différence fondamentale entre le mécanisme qualifiant une disponibilité de correcte, et celui qualifiant une disponibilité de suspecte. La qualification des disponibilités correctes pour les services d'une chaîne fonctionnelle est traitée séparément car un service peut très bien être disponible et un autre non. Cependant, pour la qualification des disponibilités suspectes, les services sont traités d'une façon globale pour une chaîne fonctionnelle, car la suspicion de la disponibilité d'un service d'une chaîne fonctionnelle doit entraîner la suspicion de la disponibilité de tous ses services. Ainsi, le mécanisme de mise à jour pourra directement exploiter la suspicion du prédicat $\neg AN(CF_k(x))$ pour la plus grande valeur de x afin de déterminer si les disponibilités des services de la chaîne fonctionnelle k doivent être suspectées.

Notons toutefois un cas particulier dans ces mécanismes : la disponibilité d'un service (noté Sc) peut être qualifiée de correcte, alors que l'effet d'un autre service (noté Ss) offert par la même chaîne fonctionnelle a été qualifié de suspect, lors de son dernier lancement. Dans ce cas, si le service dont la disponibilité peut être qualifiée de correcte a été lancé avant le deuxième (Sc avant Ss), sa disponibilité est qualifiée de suspecte, sinon elle est qualifiée de correcte puisque son effet a été constaté correct à une date ultérieure (Ss avant Sc).

Afin de mettre en œuvre ces principes, nous proposons la démarche suivante :

- la suspicion de la disponibilité des services pour chaque chaîne fonctionnelle est réalisée à partir du prédicat $\neg AN(CF_k(x))$.
- Pour chaque service, le mécanisme de mise à jour recherche si à son dernier lancement (dernière "opération exécutée" du modèle) son effet final sur la chaîne fonctionnelle est correct. Si oui, et qu'il n'y a pas de service (offert par la même chaîne fonctionnelle), lancé après, dont l'effet final sur la chaîne fonctionnelle est suspect, la disponibilité du service est qualifiée de correcte.
- Tous les services dont la disponibilité n'est pas qualifiée de correcte ou de suspecte, verront leur disponibilité qualifiée de non suspecte.

Ce mécanisme peut être illustré par l'algorithme suivant :

Pour chaque chaîne fonctionnelle CF_k **faire**
  Si ($\neg AN(CF_k(x)) = suspect$ pour la dernière valeur de x) **Alors**
    toutes les disponibilités D_i des services offerts par CF_k sont suspectées
  Fin Si
Fin Pour
Pour chaque service **faire**
  Si (l'effet final lors de son dernier lancement sur la chaîne fonctionnelle est qualifié de correct) **Alors**
    Si (il n'y a pas de service offert par la même chaîne fonctionnelle lancé ultérieurement dont l'effet final sur la chaîne fonctionnelle est suspect) **Alors**
      la disponibilité D_i est qualifiée de correcte
    Fin Si
  Fin Si
  Si (sa disponibilité D_i n'est pas qualifiée de correcte ou de suspecte) **Alors**
    sa disponibilité D_i est qualifiée de non suspecte
  Fin Si
Fin Pour

Alg. 8: Mise à jour de la disponibilité des opérations

3.2 Variables d'état

Les variables d'état n'apparaissent pas directement dans le modèle pour le diagnostic. Cependant, leurs valeurs peuvent être retrouvées via le dernier effet les ayant modifiées. Il est également nécessaire de tenir compte des effets non attendus qui auraient pu éventuellement les modifier.

La mise à jour des variables d'état se traduit par l'algorithme suivant :

Pour chaque variable d'état **faire**
  Rechercher le dernier effet du modèle ayant modifié la variable et les effets non attendus ultérieurs qui auraient pu la modifier
  Si (un effet est suspect) **Alors**
    la valeur de la variable d'état est qualifiée de suspecte
  Sinon
    Si (tous les effets sont qualifiés de corrects) **Alors**
      la valeur de la variable d'état est qualifiée de correcte
    Sinon
      la valeur de la variable d'état est qualifiée de non suspecte
    Fin Si
  Fin Si
Fin Pour

Alg. 9: Mise à jour des variables d'état

4 Décision et synthèse de loi de commande

L'objectif de cette section n'est pas de proposer une approche de décision, ceci sortant largement du cadre de ce manuscrit, mais d'ouvrir des pistes d'exploitation des techniques proposées jusqu'ici.

La fonction décision, dans un premier temps, pourra rechercher s'il est possible de synthétiser une loi de commande répondant aux objectifs de production et exploitant les capacités opératoires encore offertes par la partie opérative. L'exploitation de la description des opérations dans le cadre de la synthèse de loi de commande nécessite forcément une modification de l'algorithme de synthèse proposé dans (Henry, 2005). En effet celui-ci doit être capable de prendre en compte la suspicion des opérations mais également la suspicion des variables d'état. Plusieurs cas de figures peuvent être pris en compte :

- Il est possible de trouver une loi de commande satisfaisant les objectifs et n'utilisant pas d'opération dont la disponibilité est suspecte et dont aucun pré-requis ne porte sur des variables d'état suspectes. Dans ce cas, et s'il existe plusieurs lois de commande possibles, plusieurs critères peuvent être pris en compte dans le choix de la loi de commande. En premier celui qui est déjà pris en compte dans l'algorithme proposé dans (Henry et al., 2004a), c'est à dire le temps de cycle de la loi de commande. Ensuite il est possible afin de minimiser le risque en terme de défaillance durant l'exécution d'une loi de commande, de minimiser le nombre de chaînes fonctionnelles utilisées dont la disponibilité des opérations est qualifiée de non suspecte et ainsi privilégier celles qui sont qualifiées de correctes. On peut également minimiser le nombre de variables d'état utilisées dans les pré-requis dont la valeur est non suspectée.

- Il n'est pas possible de trouver une loi de commande satisfaisant les objectifs et n'utilisant pas d'opération dont la disponibilité est suspecte et dont aucun pré-requis ne porte sur une variable d'état suspecte. Cette fois-ci, en sus du temps de cycle, il serait intéressant de minimiser le nombre de chaînes fonctionnelles utilisées dont la disponibilité des opérations est qualifiée de suspecte. On peut également minimiser le nombre variables d'état utilisées dans les pré-requis dont la valeur est suspecte.

Une dernière piste d'utilisation du résultat de diagnostic peut être proposé dans le cadre d'une décision quant à l'intervention de la maintenance. Si une loi de commande satisfaisant les objectifs et n'utilisant pas d'opération dont la disponibilité est suspecte et dont aucun pré-requis ne porte sur une variable d'état suspecte ne peut pas être obtenu, il serait intéressant de proposer un plan d'intervention afin de minimiser le nombre de chaînes fonctionnelles à vérifier et à réparer afin de pouvoir générer une loi de commande satisfaisant les objectifs.

5 Conclusion

Ce dernier chapitre de cette partie était centré sur l'utilisation des résultats issus de la fonction diagnostic proposée. Il nous a ainsi permis de présenter l'algorithme de mise à jour du modèle de partie opérative tant sur le plan de la disponibilité des services que sur celui des variables d'état

du sous-système piloté. Après quoi nous avons proposé quelques pistes qui mériteraient d'être suivies pour envisager de poursuivre la construction du processus complet de reconfiguration et ainsi imaginer disposer d'une approche complète capable de rendre autonome un système de coordination de chaînes fonctionnelles face aux aléas de fonctionnement.

Nous nous proposons maintenant de dévoiler la dernière partie de ce document dont l'objectif est d'étudier la pertinence et la validité de l'approche de suivi/diagnostic sur un cas d'étude.

Quatrième partie

Exemple d'application

Chapitre 10

Présentation du cas d'étude

1 Introduction

Dans cette partie, nous avons souhaité développer un exemple d'application de notre approche de suivi et de diagnostic de services sur la base d'une cellule de vernissage de cartes électroniques. Cette cellule a été imaginée afin de couvrir l'ensemble des situations de dysfonctionnement à traiter (cf. section 3 page 30).

Dans le cadre de ce premier chapitre nous allons plus particulièrement nous intéresser à la présentation de cette cellule. 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 illustre sur un service notre démarche d'acquisition de la connaissance de ses caractéristiques afin de construire le modèle de la cellule de vernissage au niveau coordination. Enfin, la dernière section sera dédiée à la présentation de la plate-forme logicielle de test que nous avons utilisée afin d'expérimenter le système de suivi et de diagnostic.

2 Caractéristiques techniques du sous-système étudié

2.1 Objectif de production

La cellule considérée représentée schématiquement dans la Figure 10.1 est dédiée au vernissage de cartes électroniques permettant de mettre à l'abri les soudures et les circuits de toute oxydation éventuelle. Chacun des trois postes de travail est en mesure d'assurer d'une part un nettoyage côté pistes des cartes électroniques afin d'assurer l'adhérence du vernis et d'autre part une dépose du vernis par pulvérisation.

2.2 Description de la partie opérative

2.2.1 Système de transitique

Le système de transitique assure le transfert des palettes supports de cartes électroniques de l'entrée du système vers les différents postes de travail puis vers la sortie. Les transferts sont réalisés par trois convoyeurs à rouleaux entraînés par des moteurs. Ces derniers sont commandés (marche/arrêt) dans un seul sens de rotation. La vitesse de déplacement d'une palette sur le convoyeur est de 5cm/s. Un vérin est également utilisé pour transférer une carte électronique de

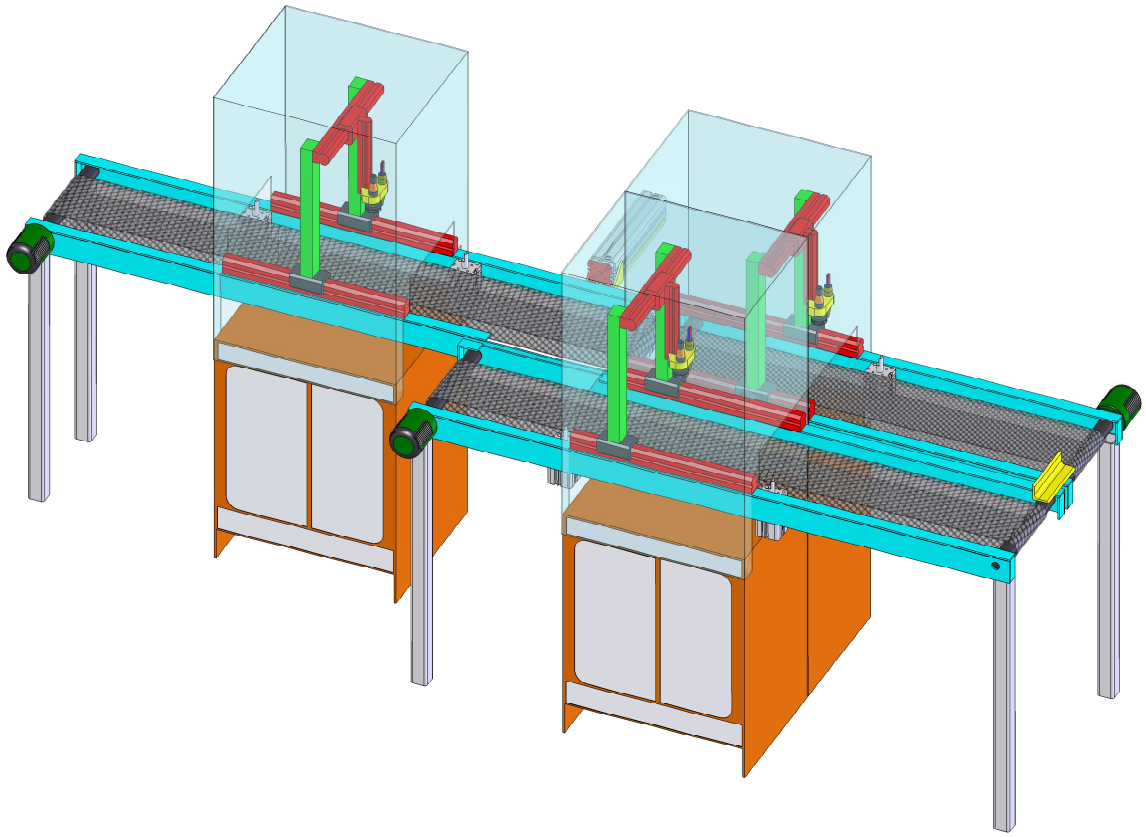


FIG. 10.1 – Partie opérative du système de vernissage de cartes électroniques

la fin du premier convoyeur vers le troisième. Le temps de transfert de la carte électronique est estimé à 5s (temps de sortie du vérin). Enfin, six butées (vérins) assurent le blocage des palettes en amont de chacun des postes de travail et donc leur maintien (cf. Figure 10.2) ; les convoyeurs peuvent ainsi rester en rotation permanente. Le temps de sortie et de rentrée des butées est de 1s.

2.2.2 Postes de travail

Chacun des postes de travail est équipé d'un pulvérisateur pouvant se déplacer selon deux axes perpendiculaires afin de pouvoir parcourir la surface de la carte électronique. Le pulvérisateur peut être alimenté soit en solvant, soit en vernis. Chaque poste de travail est ainsi capable de réaliser des opérations de nettoyage ou de vernissage des cartes. Cependant, le changement de solvant/vernis nécessite la réalisation d'un cycle de nettoyage du pulvérisateur, consistant à évacuer l'ancien produit en pulvérisant le nouveau pendant 30 secondes. La durée d'une opération de nettoyage est de une minute (pulvérisation plus temps d'évaporation du solvant) et la durée de l'opération de vernissage est de deux minutes (pulvérisation plus temps de séchage du vernis). Pour ne pas surcharger l'exemple, l'approvisionnement en solvant et vernis des machines ne sera pas considéré.

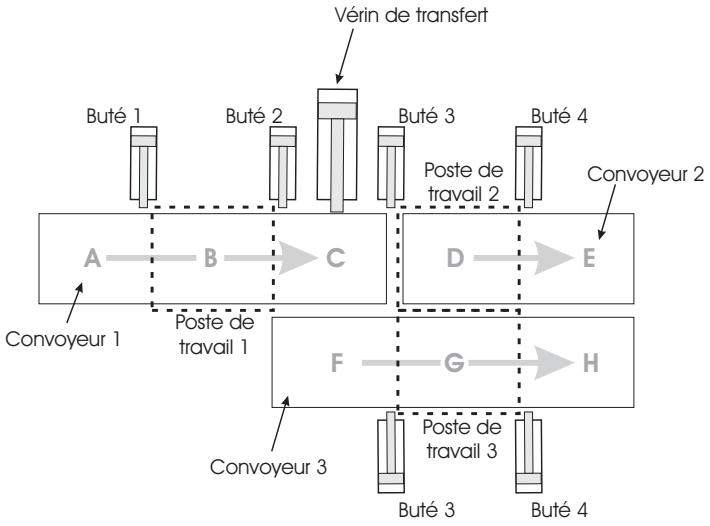


FIG. 10.2 – Synoptique du système de transitique

2.2.3 Captage d'information

Nous considérerons que le système est muni de trois capteurs permettant de détecter la présence d'une carte électronique dans chacun des postes de travail. De plus, une caméra est placée dans chacun des postes 2 et 3, permettant de vérifier, par application de techniques de traitement d'image, si le vernis a été correctement déposé.

2.2.4 Alimentation en cartes électroniques et évacuation après vernissage

Les cartes électroniques sont issues d'un système de mise en place et de soudage des composants. Elles sont automatiquement acheminées sur le système de vernissage. De même, les cartes électroniques vernies sont évacuées puis orientées vers une machine de conditionnement avant l'envoi chez le client. Cependant, l'approvisionnement en cartes électroniques à venir et leurs évacuations une fois vernies n'est pas prise en charge par le module de coordination que nous considérons ici.

3 Système de pilotage considéré

3.1 Architecture de pilotage

La structure de pilotage est présentée dans la Figure 10.3. Elle se compose d'un module de coordination capable de piloter et de coordonner un ensemble de chaînes fonctionnelles. Nous retrouvons 18 chaînes fonctionnelles : chaque poste de travail est vu comme une chaîne fonctionnelle permettant d'agir sur les caractéristiques physiques des cartes. Nous y retrouvons également les chaînes fonctionnelles n'agissant pas directement sur le flux de cartes mais permettant leur transport (convoyeurs, butées, vérin de transfert). Enfin, la remontée d'information au niveau coordination est assurée par cinq chaînes fonctionnelles composées uniquement d'une chaîne d'acquisition ; elles seront exploitées dans le cadre de la surveillance.

Chacune de ces chaînes fonctionnelles se voit attribuer un module de contrôle/commande basé sur une approche de surveillance et de supervision intégrée à la commande.

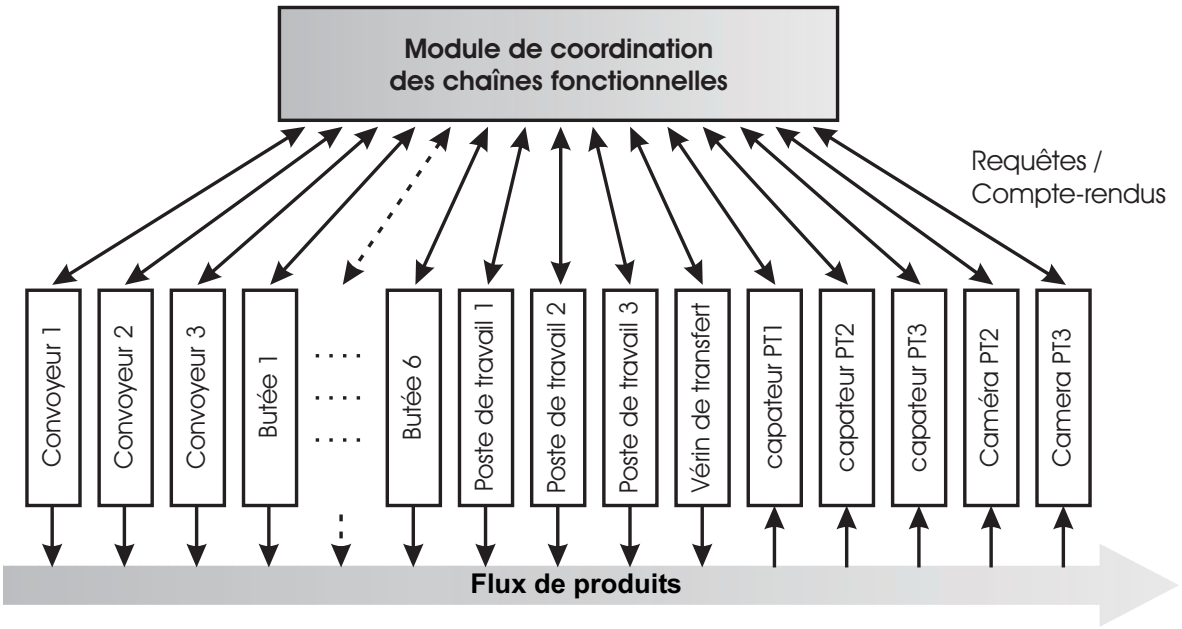


FIG. 10.3 – Architecture de pilotage considérée

3.2 Loi de commande considérée au niveau coordination

La loi de commande retenue dans le cadre de cet exemple est présentée sous forme d'un réseau de Petri dans la Figure 10.4. Nous noterons que la loi de commande intègre des parallélismes entre les opérations, induisant la présence possible de plusieurs cartes électroniques simultanément dans le sous-système piloté. De plus, afin de minimiser le temps de cycle, le poste de travail 1 sera utilisé exclusivement pour le nettoyage des cartes électroniques et les postes de travail 2 et 3 pour le vernissage. Nous retrouvons dans la Figure 10.4 :

- des parties en noir, pour l'envoi des requêtes d'appel aux service et pour l'attente des compte-rendu de leur exécution,
- des parties grisées, pour le lancement des services de surveillance,
- des parties grisées en italique, pour l'envoi des requêtes et comte-rendus fictifs des opérations induites ; la durée de ces opérations induites étant simulée par des temporisations.

FIG. 10.4 – Loi de commande considérée

4 Description des opérations

La démarche d'acquisition de la connaissance des capacités offertes par la partie opérative se base sur le concept du modèle d'opérations présenté dans le chapitre 4. La démarche d'acquisition sera préalablement basée sur une distinction des types d'opérations. Ainsi nous pouvons dénombrer :

- vingt six opérations d'action (ouvrir butée i, fermer butée i, vernissage carte par PT2, etc),
- six opérations induites (transfert de la pièce de la position A à la position B, etc),
- trois opérations requises (placer carte en A, évacuer carte en E, évacuer carte en H),
- cinq opérations de surveillance (surveiller présence carte en A, surveiller vernissage de la carte en D, etc).

L'ensemble de ces opérations est décrit dans l'annexe C. Dans un soucis de concision seule une d'entre elle sera reprise ici.

Nous proposons ainsi de décrire le service *Sortir vérin de tranfert* par une opération d'action (cf. Figure 10.5). Comme décrit dans le cahier des charges, la durée de sortie du vérin est de 5s. Le lancement du service ayant comme effet de modifier la position de la tige du vérin, nous utilisons la variable d'état $Vt.position$ afin de décrire l'évolution de la chaîne fonctionnelle. Cette évolution ne doit être lancée que si la tige du vérin n'est pas dans la position sortie. Nous notons cette information dans le champ pré-condition de l'évolution de la chaîne fonctionnelle. Afin de s'assurer que la sortie du vérin ne soit pas bloquée par une éventuelle carte, nous imposons dans la pré-contrainte et contrainte, l'absence de carte entre la position B et C et entre la position C et D, et ceci quelque soit l'évolution associée du flux de produits.

Nous nous focalisons maintenant sur les évolutions possibles du flux de produits. Si une carte est présente soit à la positon C soit entre les positions C et F, la sortie du vérin aura pour effet de modifier la position de cette carte. Ainsi, nous décrivons deux évolutions du flux de produits avec comme pré-conditions $CarteEnC == non\ vide$ et $CarteEnCF == non\ vide$. Pour chacune de ces deux évolutions, afin de garantir l'absence de collision entre deux cartes, nous imposons comme pré-contrainte et/ou contrainte l'absence de carte sur la trajectoire de celle transférée.

Cette démarche est tout à fait similaire pour la description des services présentés dans l'annexe C.

5 Environnement de test

Bien que la partie opérative soit virtuelle, nous l'avons simulée sur ordinateur (cf. Figure 10.6) et développé réellement son architecture de pilotage (le module de coordination et les 18 modules de contrôle/commande des chaînes fonctionnelle) sur la plate-forme logicielle CERBERE déve-

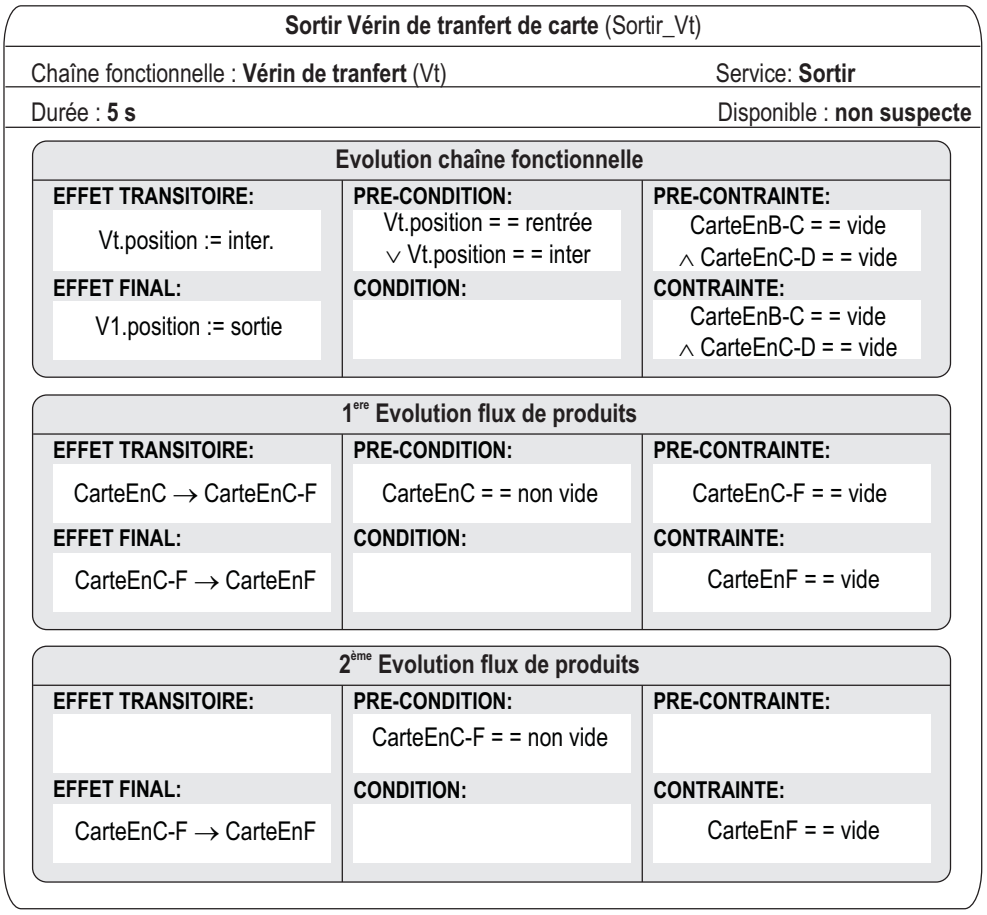


FIG. 10.5 – Opération d'action *Sortir vérin de transfert de carte*

loppée dans notre équipe de recherche au sein du Laboratoire d'Automatique de Grenoble. Cette plate-forme logicielle s'appuie sur un PC pentium 1 (fréquence processeur 200MHz, mémoire RAM 64 Mo), le système d'exploitation multi tâches temps réel VxWorks et d'un ensemble de fonctionnalités issues de la mise en œuvre de résultats de recherches. Parmi celles-ci, nous avons utilisé la structure générique d'un module de contrôle/commande, un joueur de réseau de Petri (Chuiton, 2005), ainsi que des mécanismes de routage des informations inter niveaux présentés dans (Granier, 2003). En sus de ces composants logiciels existants que nous avons paramétrés pour ce cas d'étude, nous avons lancé le développement logiciel de notre approche de suivi et de diagnostic de services. Ce travail a été réalisé dans le cadre du mémoire CNAM de M. Freddy Murand (Murand, 2007). Ainsi, les mécanismes exposés, à l'exception de la mise à jour du modèle des capacités de la partie opérative, ont été intégrés à l'architecture logicielle existante (cf. Figure 10.7)

6 Conclusion

Dans ce chapitre nous avons présenté la cellule virtuelle sur laquelle nous allons évaluer réellement notre approche de diagnostic. En effet, autant la cellule sera simulée autant son architecture de pilotage est réelle. Nos algorithmes ont été quant à eux mis en œuvre en C++

et intégrés sous VxWorks aux autres fonctions de pilotage déjà présents dans la plate-forme logicielle CERBERE.

Le chapitre suivant présente une analyse critique des résultats obtenus par la fonction suivi testée.

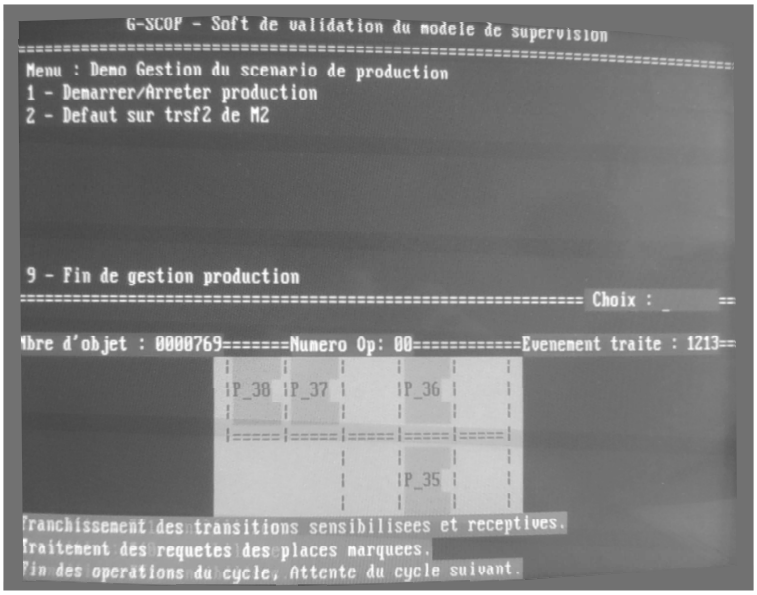


FIG. 10.6 – Simulateur du système de vernissage

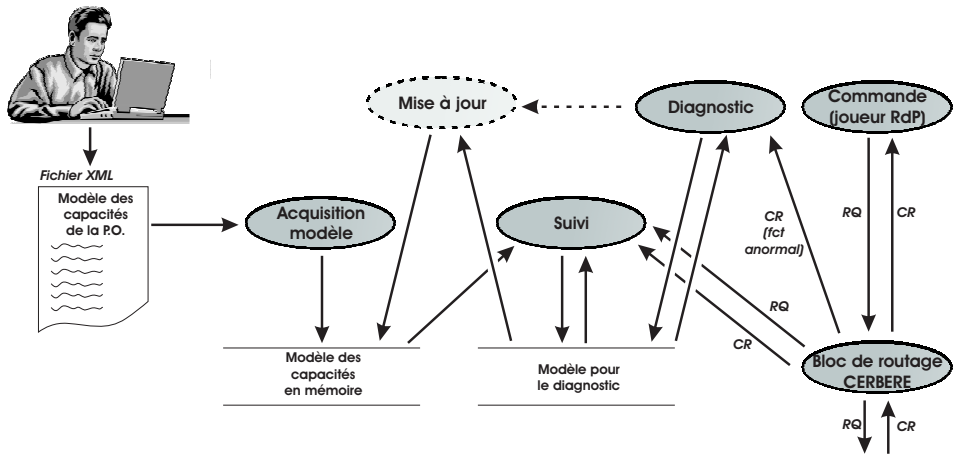


FIG. 10.7 – Architecture générale du logiciel

Chapitre 11

Suivi : génération du modèle et réduction

1 Introduction

Dans ce chapitre nous nous proposons de tester en situation réelle notre algorithme de suivi au sein de la plate-forme logicielle de test CERBERE.

La première section de ce chapitre donne les conditions initiales de la simulation. Dans la deuxième section, nous aurions pu restituer l'étude que nous avons menée sur l'évolution, dans le temps, du contenu du modèle de fonctionnement passé afin de le comparer avec celui prévu "à la main". Cependant, pour des raisons de concisions nous avons préféré ne retenir ici que les résultats d'une autre étude réalisée, orientée quant à elle sur l'analyse des performances exprimées en termes de taille du modèle et des temps nécessaires à la génération et à la réduction du modèle

1.1 Conditions initiales du test

Avant de dévoiler l'expérimentation menée dans la suite du manuscrit, nous nous proposons ici de décrire l'état initial dans lequel nous considérons le sous-système piloté :

- convoyeurs en rotation (Convi.rot=oui pour i allant de 1 à 3)
- butées fermées (Bi.position = fermée pour i allant de 1 à 6),
- vérin de transfert rentré (Vt.position=rentrée),
- postes de travail 1, 2 et 3 libres (PTi.occ=libre pour i allant de 1 à 3),
- chaînes d'acquisition libres (CamD.occ=libre, CamG.occ=libre, CB.occ=libre, CD.occ=libre, CG.occ=libre)
- pas de carte présente dans le sous système piloté (CarteEnA=vide, CarteEnA-B=vide, CarteEnB=vide, CarteEnB-C=vide, etc)

2 Évolution de la taille du modèle

La structure du modèle pour le diagnostic est implantée sous le logiciel développé par M. Freddy Murand sous forme d'une structure de données dynamique illustrée dans la fenêtre droite de la Figure 11.1. Nous y retrouvons dans la fenêtre de gauche un exemple de code correspondant à la connexion de pré-requis aux opérations. Nous proposons dans cette section de mener des tests quant à l'évolution de la taille du modèle pour le diagnostic.

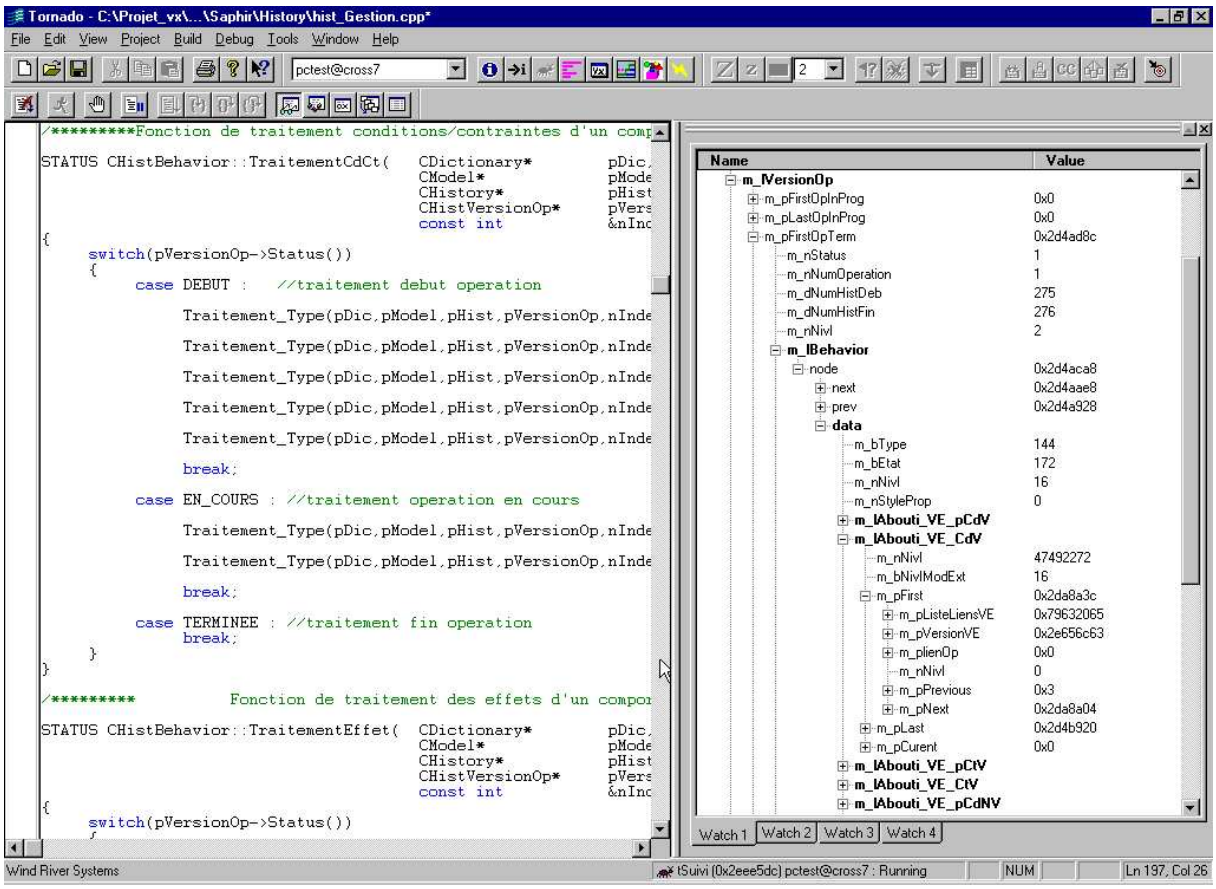


FIG. 11.1 – Illustration de la structure de données du modèle pour le diagnostic

2.1 Variation de la taille du modèle

La variation de la taille du modèle est illustrée au travers de la Figure 11.2. Cette courbe représente la taille du modèle en nombre d'effets en fonction de l'arrivée des événements à la fonction suivi. Cette simulation a été réalisée en injectant dix cartes électroniques à la suite dans la cellule de vernissage. La courbe montre des augmentations de la taille du modèle (ajout de versions d'opérations exécutées) puis des diminutions correspondant à l'action du mécanisme de réduction. Dans la tendance générale de la courbe, trois phases semblent se distinguer. Ces différentes phases peuvent être interprétées de la manière suivante :

– Une première phase d'injection progressive des cartes dans la cellule. Ainsi, progressive-

ment les postes de travail 1 puis 2 puis 3 sont alimentés en carte à nettoyer ou à vernir. Ceci se traduit par une augmentation de versions d'opérations contenues dans le modèle pour le diagnostic.

– Une deuxième phase montre un fonctionnement cyclique de la cellule, où chaque poste de travail reste alimenté de façon régulière en cartes.

– Une troisième phase montre une diminution progressive de la taille du modèle. L'alimentation des postes de travail en cartes est progressivement interrompu, et le mécanisme de réduction, basé sur l'exploitation des opérations de surveillance des chaînes d'acquisition placées au niveau des postes 2 et 3, supprime progressivement les anciennes versions d'opération. Une fois l'ensemble des cartes évacuées, un nombre résiduel de versions d'opérations reste présent dans le modèle. Ceci s'explique par le fait que certains effets ne seront qualifiés de corrects qu'après le passage du produit suivant.

Une analyse plus globale de la courbe montre que la taille atteint une limite supérieure. Les diminutions du nombre d'effets montrent l'intervention du mécanisme de réduction. Dans ce sens nous voyons par l'expérimentation que le nombre d'effets contenus dans le modèle pour le diagnostic est borné. Bien entendu, comme nous le verrons dans les perspectives, les résultats de cette expérimentation devront être validés sur le plan théorique.

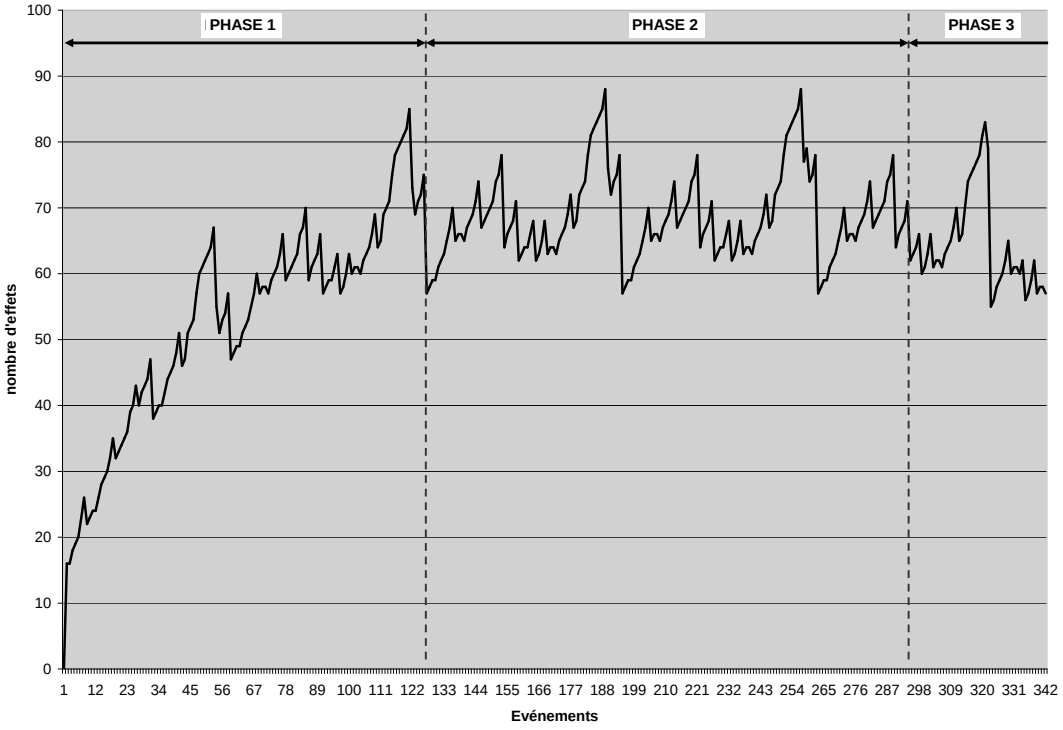


FIG. 11.2 – Taille du modèle pour le diagnostic en fonction de l'arrivée des requêtes des comptes rendus

La borne supérieure mise en évidence dans cette expérimentation dépend du niveau d'observabilité intégré dans les chaînes fonctionnelles. Il va s'en dire qu'une cellule ne présentant aucune observabilité entraînerait une explosion de la taille du modèle. Ainsi, nous proposons dans la prochaine section d'étudier l'impact du niveau de captage sur la borne supérieure de la taille du modèle.

2.2 Borne supérieure de la taille du modèle

Cette section propose d'étudier, en fonction du captage disponible, si une borne supérieure relative à la taille du modèle existe. Pour cela, nous injectons deux cents produits successivement dans la cellule. Nous limiterons notre analyse aux cinq chaînes d'acquisitions décrites dans le chapitre 10. Les résultats obtenus sont synthétisés dans le tableau suivant 11.1.

Présence capteur	cas 1	cas 2	cas 3	cas 4	cas 5	cas 6	cas 7	cas 8
Capteur carte à la position B	indifférent	indifférent	non	non	non	oui	non	oui
Capteur carte à la position D	indifférent	indifférent	non	non	oui	non	oui	oui
Capteur carte à la position G	indifférent	indifférent	non	oui	non	non	oui	oui
Caméra à la position D	non	indifférent	oui	oui	oui	oui	oui	oui
Caméra à la la position G	indifférent	non	oui	oui	oui	oui	oui	oui
Taille maximum	dépassement mémoire	dépassement mémoire	169	151	147	130	110	88

Tab. 11.1 – Borne supérieure de la taille du modèle en fonction du niveau d'implantation des chaînes d'acquisition

Une première analyse de ces résultats confirme que l'augmentation du nombre de chaînes d'acquisition entraîne une diminution de la borne supérieure que prend la taille du modèle. Cette borne peut effectivement varier de la valeur zéro si toutes les chaînes fonctionnelles sont équipées d'un moyen d'observation, à une valeur très grande dépassant les capacités mémoires non seulement si le nombre de chaînes d'acquisition est insuffisant mais aussi en fonction de leurs positions dans la cellule. Ainsi nous observons qu'il est nécessaire de disposer d'au moins des deux chaînes d'acquisition permettant de surveiller le vernissage des cartes aux positions D et G. Il s'ouvre ici des perspectives quant à la problématique du placement de capteurs.

2.3 Performances temporelles de la fonction suivi

Afin d'évaluer les performances temporelles de l'algorithme de génération du modèle et de réduction, nous proposons deux mesures quant aux performances temporelles de l'algorithme de génération et de réduction du modèle. La première mesure concerne les temps maximum et

moyens de l'ajout des effets correspondants au début ou à la fin d'une opération exécutée. Ce temps correspond à la création de l'opération mais également à la connexion de ses pré-requis au reste du modèle. Nous obtenons après avoir injecté dix cartes électroniques dans le système : $t1_{max} = 20\ ms$ et $t1_{moy} = 4\ ms$.

La deuxième mesure proposée ici, est le temps maximum et moyen de l'application (suite à la réception d'un compte-rendu d'exécution d'une opération) des mécanismes de parcours arrière et avant du modèle complet afin d'assurer sa réduction. Cette mesure a été effectuée sur le cas numéro trois de la section précédente qui présentait la taille la plus grande du modèle. Nous obtenons après avoir injecté dix cartes électroniques dans le système : $t2_{max} = 15\ ms$ et $t2_{moy} = 10\ ms$.

3 Conclusion

Dans ce chapitre nous avons présenté les résultats principaux de l'étude que nous avons menée quant à la validité du mécanisme proposé. Cette étude nous a permis, dans un premier temps de valider, sur l'exemple retenu, le modèle généré automatiquement par rapport à celui prévu (Murand, 2007). Dans un deuxième temps, nous avons souhaité valider la mise en œuvre informatique de notre algorithme au travers d'une étude de la taille du modèle généré. Les résultats ont confirmé nos attentes. D'une part la taille du modèle n'explose pas pour un niveau de captage réaliste et d'autre part reste fonction du nombre de capteurs présents au sein même des chaînes fonctionnelles. Enfin, nous avons souhaité mesurer les temps mis pour gérer l'ajout d'un effet dans le modèle. Ces dernières s'avèrent être tout à fait réalistes pour un niveau coordination. Bien entendu ces résultats se doivent dans un avenir proche d'être complétés par une étude théorique.

Chapitre 12

Diagnostic en ligne des services : application à la cellule de vernissage

1 Introduction

Dans ce dernier chapitre du manuscrit, nous nous proposons de réaliser des tests permettant de valider par simulation notre approche de diagnostic de services. Ces tests seront basés sur 3 scenarii de défaillances résumés dans la Figure 12.1 et permettant d'illustrer les différentes situations énumérées section 3 page 30. L'étude de ces trois scenarii permet de mettre en valeur les apports de notre approche.

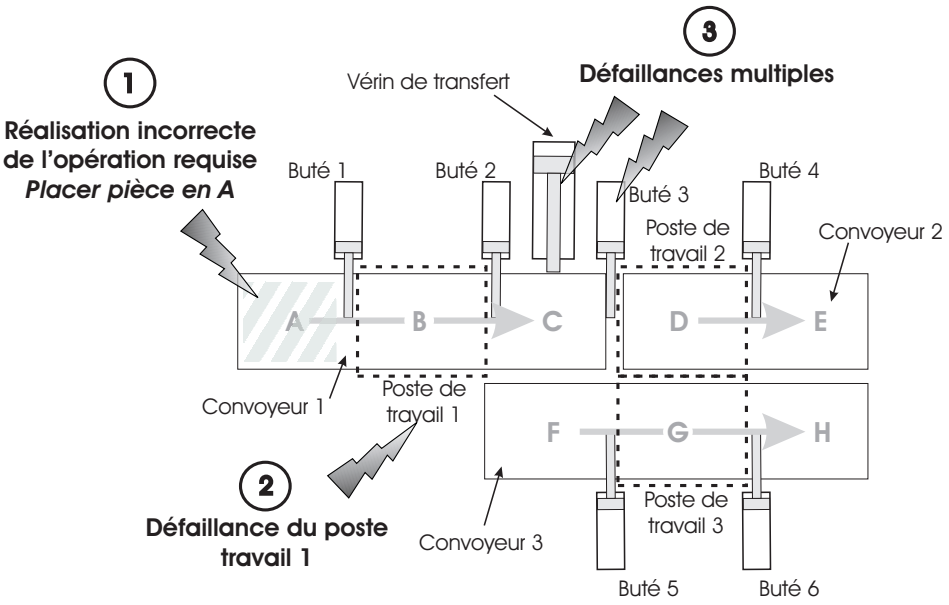


FIG. 12.1 – Scenarii considérés

2 Scénario 1 : Mauvaise réalisation d'une opération requise

Ce premier scénario simule une réalisation incorrecte de l'opération requise *Placer une carte en A*. Nous considérons que deux cartes sont présentes dans le système de vernissage.

La première est en cours de nettoyage sur le poste de travail 1, et la deuxième en cours de vernissage sur le poste de travail 2. Un compte-rendu normal de fin de l'opération requise *Placer une carte en A* est reçu par le niveau coordination, alors qu'aucune carte n'a été déposée à la *position A* (cf. Figure 12.2).

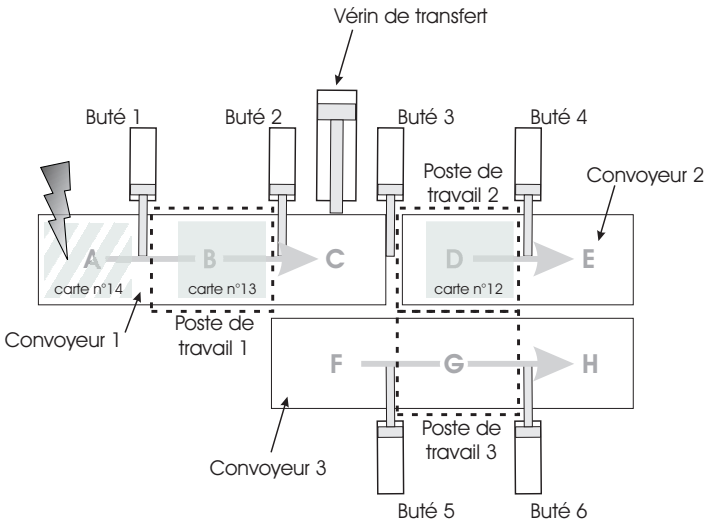


FIG. 12.2 – Synoptique du système pour le scénario 1

L'objectif de ce scénario est de vérifier que la version de l'opération requise sera suspectée à juste titre. Compte tenu du fait que l'incohérence entre le fonctionnement réel et celui attendu ne peut être révélé que par la chaîne d'acquisition de l'information *CarteEnB*, le diagnostic doit alors suspecter le transfert de la pièce dont le nettoyage vient d'être effectué sur le poste de travail 1.

2.1 Fonctionnement du système suite à l'occurrence de la défaillance

Le fonctionnement du système de vernissage à partir du début du scénario peut être décrit comme suit :

- Fin de l'opération de *nettoyage carte dans le poste 1*, ouverture de la *butée 2, et début de l'opération induite de transfert de la carte de la position B à la position C.*
- *Fin de l'opération induite de transfert, fermeture de la butée 2 début de l'opération sortir vérin de transfert, ouverture de la butée 1, début de l'opération induite de transfert de la carte de la position A à la position B.*
- *Fin de l'opération de transfert, réception du CRA de la chaîne fonctionnelle CB suite à l'absence de l'information capteur sur la présence de la pièce en B.*

2.2 Diagnostic des services et mise à jour

Nous proposons dans le cadre de ce scénario de détailler le mécanisme de diagnostic permettant d'aboutir aux opérations exécutées suspectes dans le modèle.

Le mécanisme de diagnostic est lancé par la réception du CRA de la chaîne fonctionnelle CB, ainsi l'algorithme 5 débute par la suspicion des effets connectés au pré-requis de l'opération de surveillance de la *présence d'une carte à une position B*. Ainsi, le fonctionnement normal de la chaîne fonctionnelle d'acquisition CB ainsi que l'effet final de l'opération induite *transférer carte de la position A à la position B* sont suspectés. Ensuite, le mécanisme de propagation arrière, recherche les effets connectés aux pré-requis de ce dernier effet :

- pour *Conv1.rot==vrai*, le dernier effet ayant été qualifié de correct, un changement intempestif de la chaîne fonctionnelle est ajouté au modèle,
- pour *Butée1.position==ouverte*, l'effet final de la dernière opération *ouvrir butée1* est suspecté,
- pour *CarteEnA==non vide*, l'effet final de l'opération requise *Placer une carte en A* est suspecté,
- pour *CarteEnB==vide* aucun effet ou séquence d'opération induite n'est trouvé,

Ainsi repartant de ces effets suspectés, l'algorithme poursuit sa propagation arrière de la suspicion, et ainsi de suite. Après quoi, la phase de propagation avant est effectuée dans le modèle. Partant par exemple du changement intempestif de l'état de la chaîne fonctionnelle $Conv1$, qui a également été connecté, lors de son insertion dans le modèle, au pré-requis de l'opération induite *transférer carte de la position B à la position C*, l'effet transitoire puis final de cette opération sont suspectés par l'algorithme 5.

Une partie de ces résultats de diagnostic est illustrée au travers de la Figure 12.3. Nous y retrouvons les versions d'opérations suspectes et non suspectes (i.e. $\neg AN(CF_k(2x-1))$ suspect/non suspect). Les opérations exécutées non contenues dans la Figure correspondent aux opérations dont l'ensemble des pré-requis de ses évolutions ainsi que les effets sont corrects. Nous retrouvons également un exemple de connexion des pré-requis d'une opération. D'un point de vue performance temporelle de l'algorithme de diagnostic, une mesure permet de connaître le temps d'exécution des phases de propagation avant et arrière dans le modèle (la mise à jour n'étant pas pour le moment mise en œuvre) : nous avons obtenu un temps de 23 ms mesuré entre la date d'occurrence du CRA et celle de la fin d'activité de l'algorithme de propagation.

Dans un deuxième temps nous avons mis en œuvre "à la main" le mécanisme de mise à jour, le résultat quant à la suspicion des disponibilités des opérations est repris dans le tableau 12.1.

Nous nous sommes limités à reprendre uniquement les disponibilités des opérations et des variables d'état suspectes dans le sens où elles seules permettent d'observer le résultat des mécanismes de suspicion du diagnostic, les informations correctes et étant issues du mécanisme de réduction.

Parmi les disponibilités suspectes, nous retrouvons bien l'opération requise *Placer une carte en A*, initialement à l'origine de la propagation de défaillance. Toutefois, la disponibilité d'autres

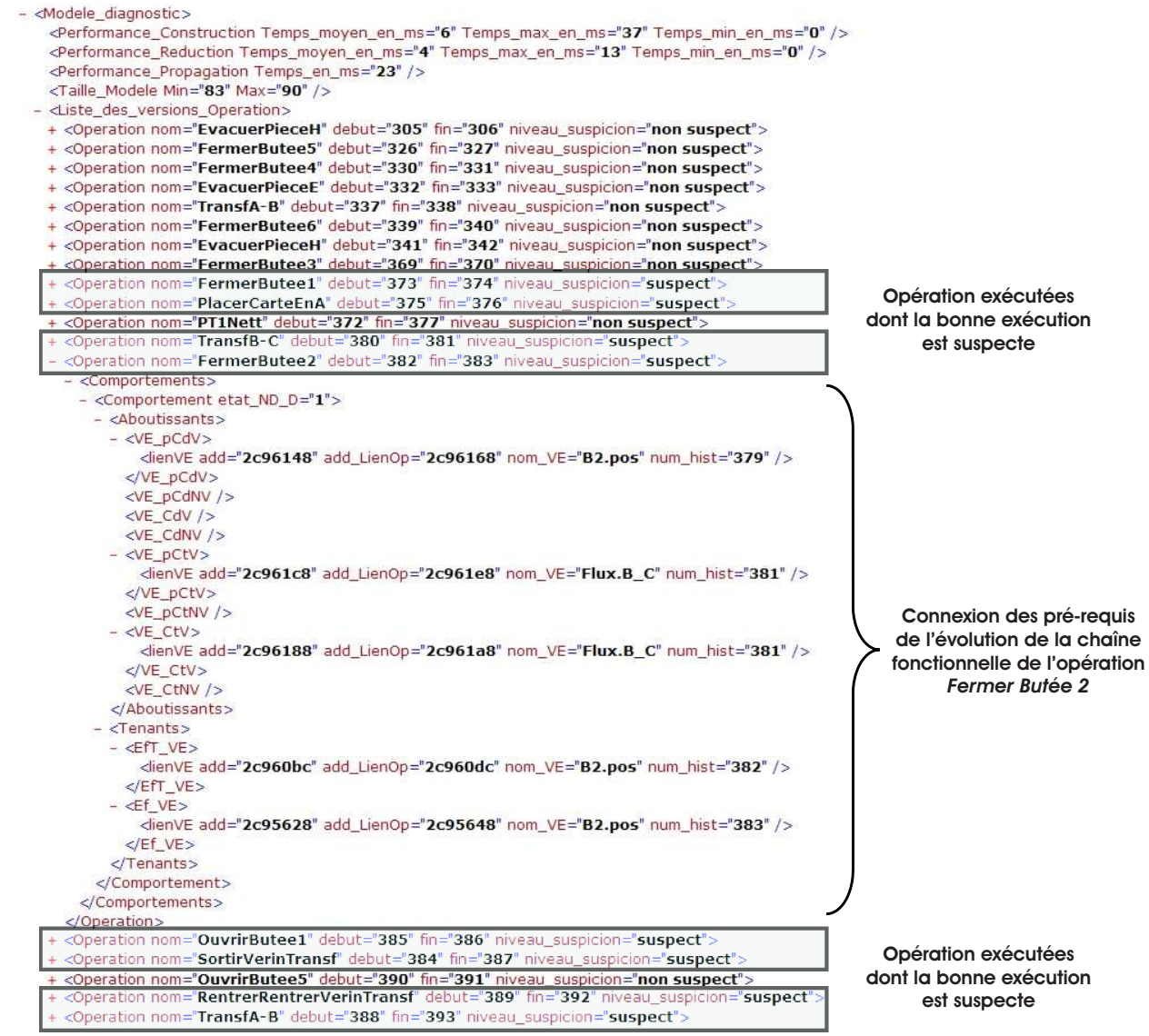


FIG. 12.3 – Modèle pour le diagnostic après application des mécanismes de propagation de la suspicion

opérations a été suspectée. Après analyse de la propagation complète de la suspicion, les différentes origines obtenues par le mécanisme de propagation arrière sont :

- défaillance de la chaîne fonctionnelle *butée 1* empêchant ainsi le transfert de la carte numéro douze de la *position A* vers la *position B*, position du captage,
- défaillance de la chaîne fonctionnelle *butée 2* empêchant ainsi le transfert complet de la carte numéro treize de la *position B* vers la *position C*,
- défaillance du *convoyeur 1* (changement intempestif de son état),
- défaillance de l'environnement (mauvaise réalisation de l'opération requise *Placer une carte en A*),
- défaillance de la chaîne fonctionnelle *capteur présence carte en B*,

Variables d'état suspectes	Opérations dont la disponibilité est suspecte
Carte12.nettoyage	Ouvrir Butée 1
Carte12.vernis	Fermer Butée 1
Carte13.nettoyage	Ouvrir Butée 2
Carte13.vernis	Fermer Butée 2
CarteEnA	Placer une carte en A
CarteEnA-B	Sortir vérin de transfert
CarteEnB	Rentrer vérin de transfert
CarteEnB-C	Surveillance de la présence d'une carte en B
CarteEnC	Démarrer convoyeur 1
CarteEnC-F	Arrêter convoyeur 1
Conv1.rot	
B1.position	
B2.position	
Vt.position	
CB.occ	

Tab. 12.1 – Résultat de la mise à jour pour le scénario 1

et les conséquences obtenues par le mécanisme de propagation avant sont :

- l'arrêt possible du *convoyeur 1* pendant le transfert de la carte dont l'identificateur est treize, puis la fermeture de la *butée 2* sur cette carte, entraînant non seulement une suspicion des capacités offertes par la *butée 2* mais aussi des variables d'état de la carte numéro treize,
- l'arrêt possible du *convoyeur 1* pendant le transfert de la carte dont l'identificateur est treize, puis la sortie du *vérin de transfert* sur cette carte à cheval sur les deux positions, entraînent une suspicion des capacités offertes par le *vérin de transfert*,
- un changement intempestif de l'état de la *butée 1* pendant le transfert de la carte douze durant son transfert entre la *position A* et la *position B*, entraîne la suspicion des variables d'état de cette carte,
- une mauvaise réalisation de l'opération requise *Placer une carte en A*,
- la suspicion des variables d'état caractérisant les chaînes fonctionnelles pouvant être soit à l'origine du compte-rendu traduisant le fonctionnement anormal, soit affectées par les défaillances possibles,
- les variables d'état caractérisant la présence ou non d'une pièce dans les zones suspectées du système de vernissage.

Nous pouvons constater que les origines possibles mises en exergue par le mécanisme de propagation arrière correspondent bien à des défaillances réalistes qui auraient conduit à l'absence de la présence de la carte en B. Afin d'affiner ce résultat, il serait nécessaire de rentrer dans un raisonnement plus profond, basé sur une connaissance plus fine des chaînes fonctionnelles en intégrant leur comportement en fonctionnement anormal.

Nous observons d'ailleurs, sur le résultat du mécanisme de propagation avant de la suspicion, la limite du raisonnement. Les variables d'état caractérisant le nettoyage des cartes ont été

suspectées alors que la fermeture d'une butée à un moment inapproprié ne peut en aucun cas dégrader les cartes qui sont en réalité maintenues sur une palette.

2.3 Reconfiguration du système de commande envisageable

L'ensemble des chaînes fonctionnelles composant la première partie du système de transitique étant suspectées, il n'est pas envisageable de réaliser une reconfiguration du système de commande. Toutefois, il serait tout de même intéressant de terminer le vernissage de la carte numéro douze, puisque les opérations nécessaires n'ont quant à elles pas été suspectées.

3 Scénario 2 : défaillance du *poste de travail 1*

Ce deuxième scénario consiste à simuler une défaillance de la chaîne fonctionnelle *poste de travail 1*. Suite à l'obstruction de la buse de pulvérisation, le nettoyage des cartes n'est plus réalisé correctement sur ce poste de travail (cf. Figure 12.4). Nous considérons qu'une carte est présente dans le système sur le *poste de travail 1*, une deuxième carte sera ensuite introduite dans le système durant le vernissage de la première.

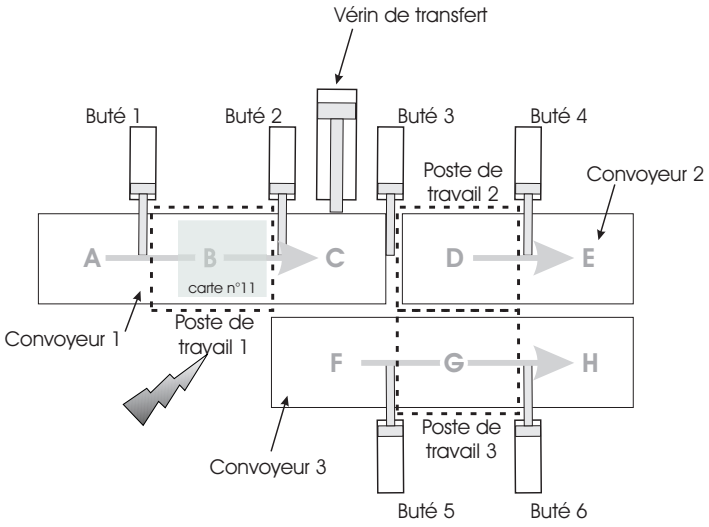


FIG. 12.4 – Synoptique du système pour le scénario 2

L'objectif principal de ce scénario est de valider la prise en compte de la propagation multiple d'une défaillance. En effet, les deux cartes présentes ne seront pas nettoyées correctement et seront orientées l'une sur le *poste de travail 2* et l'autre sur le *poste de travail 3*. La défaillance se propage donc sur ces deux produits. Nous souhaitons également estimer la couverture de la suspicion, et les possibilités en terme de reconfiguration.

3.1 Fonctionnement du système suite à l'occurrence de la défaillance

Suite à la défaillance du *poste de travail 1*, nous considérons le fonctionnement suivant :

– fin de l'opération de *nettoyage carte dans le poste 1* sur la carte numéro onze (opération non réalisée correctement), transfert de cette carte jusqu'à la position D (réduction du

modèle par exploitation de la surveillance de l'arrivée de la carte en D), puis début de l'opération *vernissage carte dans le poste 2* sur la carte.
- Arrivée d'une carte (identifiant numéro treize) à la *position A*, transfert à la *position B*, nettoyage de la carte, puis transfert vers la *position C* puis vers la *position F*.
- Réception de la chaîne fonctionnelle *caméra en D d'un compte rendu traduisant un fonctionnement anormal (suite à la détection d'un vernissage non conforme par le module de pilotage de cette chaîne fonctionnelle).*

3.2 Résultats

Par application des mécanismes de propagation et de mise à jour le résultat suivant est obtenu :

Variables d'état suspectes	**Opérations dont la disponibilité est suspecte**
Carte11.nettoyage	Vernissage dans poste de travail 2
Carte11.vernis	Nettoyage dans poste de travail 2
Carte12.nettoyage	Configuration de poste de travail 2 pour vernissage
Carte12.vernis	Configuration de poste de travail 2 pour nettoyage
PT1.occ	Vernissage dans poste de travail 1
PT2.occ	Nettoyage dans poste de travail 1
CamD.occ	Configuration de poste de travail 1 pour vernissage
	Configuration de poste de travail 1 pour nettoyage
	Surveillance du vernissage d'une carte en D

TAB. 12.2 – Résultat de la mise à jour pour le scénario 2

Le temps d'exécution des phases de propagation avant et arrière dans le modèle est de : $t_1 = 6\ ms$.

Les résultats obtenus sont tout à fait conformes à ceux qui étaient attendus. Les variables d'état des deux cartes ont bien été suspectées montrant bien la prise en compte de la propagation multiple. Nous pouvons encore une fois noter une limite du raisonnement basé sur une connaissance exclusive du fonctionnement normal, il s'agit de la suspicion par le mécanisme de propagation arrière de la disponibilité des services offerts par le *poste de travail 2*. Cette suspicion est issue de la transgression possible de la pré-contrainte portant sur le nettoyage de la carte pour le vernissage. Cependant, la transgression de ce pré-requis n'entraîne en réalité que la mauvaise réalisation du vernissage sans affecter le fonctionnement du poste de travail. Ceci correspondant à l'impact du non respect de la pré-contrainte, qui est une connaissance du fonctionnement anormal, et donc non pris en compte dans notre approche.

3.3 Reconfiguration du système de commande envisageable

Les deux cartes présentes dans le système de vernissage doivent tout d'abord être évacuées, car la suspicion de leurs variables d'état interdit de procéder à d'autres traitements. Ensuite,

les opérations offertes par le *poste 3* ainsi que toutes les opérations permettant le transfert des pièces restant disponibles, il est possible de réaliser le nettoyage et le vernissage des cartes sur ce poste. Cependant, cette solution nécessite de lancer des opérations de configuration du poste entre chacune de ces phases, entraînant une forte augmentation du temps de traitement global des cartes. Il est donc possible de faire fonctionner le système de vernissage de cartes en mode dégradé, au prix d'une baisse de la performance.

4 Scénario 3 : défaillance de la *buté 3* et du *vérin de transfert*

L'objectif de ce troisième scénario est double : illustrer la prise en compte de la propagation de défaillances multiples et de la propagation due à une évolution spontanée du flux de produits (non prévue dans la loi de commande).

Pour les besoins de l'exemple considérons que suite à une indisponibilité du *poste de travail 2* (suite au scénario 3 par exemple), toutes les cartes traitées par le système sont nettoyées dans le *poste de travail 1* et vernies dans le *poste de travail 3*. Le scénario consiste à simuler non seulement une défaillance de la *buté 3* (passage intempestif du distributeur du côté butée ouverte) et du *vérin de transfert* (rupture du flexible pneumatique permettant la mise sous pression de la chambre assurant la sortie du vérin). Une carte électronique est alors en cours de transfert de la *position B* à la *position C* en accord avec la Figure 12.5

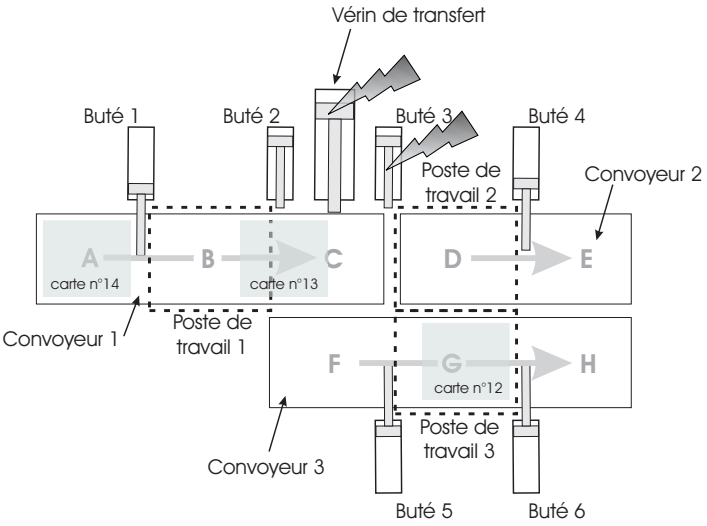


FIG. 12.5 – Synoptique du système pour le scénario 3

4.1 Fonctionnement du système suite à l'occurrence des défaillances

Suite à l'occurrence des défaillances, la carte numéro treize termine son transfert vers la *position C*. Ensuite, la carte n'étant pas transférée vers la *position F*, et la *butée 3* étant ouverte, la carte est spontanément transférée vers la *position D*. La chaîne fonctionnelle *capteur présence carte en D* envoie alors au niveau coordination un compte-rendu sur ce changement intempestif de l'état du flux de produits.

4.2 Résultat du diagnostic

Nous devons noter ici que ce scénario a été traité "à la main", l'algorithme permettant de rechercher les opérations induites non prévues à partir d'un compte-rendu traduisant un changement intempestif du flux de produits n'étant encore qu'en cours de tests et de débbugage.

Par application des mécanismes de propagation et de mise à jour le résultat suivant est obtenu :

Variables d'état suspectes	**Opérations dont la disponibilité est suspecte**
B3.position	Ouvrir butée 3
Vt.position	Fermer butée 3
CD.occ	Sortir Vérin de transfert
CarteEnC	Rentrer vérin de transfert
CarteEnC-F	Surveillance de la présence d'une carte en D
CarteEnF	
CarteEnC-D	
CarteEnD	

Tab. 12.3 – Résultat de la mise à jour pour le scénario 4

A partir de l'information *CarteEnD=non vide*, nous avons recherché la séquence d'opérations induites ayant pu conduire cette variable d'état à cette valeur. La séquence obtenue est composée d'une seule opération induite, *Transférer carte de la position C à la position D*. Le temps de transfert de la pièce entre ces deux positions ajouté à l'instant d'arrivée de la carte à la *position C* correspondant à l'instant du CRA, cette opération a été ajoutée. Ainsi les pré-requis à cette opération ont été suspectés, c'est à dire la position du *vérin de transfert* et de la *butée 3*. Ensuite le mécanisme de propagation avant est identique aux scenarii précédents.

Nous notons ici que l'application du mécanisme a conformément à ce qui était attendu, suspecté non seulement les services offerts par la chaîne fonctionnelle *la butée 3* mais également ceux offerts par la chaîne fonctionnelle *Vérin de transfert*.

5 Conclusion

L'exemple que nous venons de traiter dans cette partie est basé sur une simulation du comportement d'une cellule de vernissage de cartes électroniques. Il s'agit notamment de démontrer à la fois la pertinence du mécanisme de suivi proposé ainsi que celui du diagnostic de services. Dans ce dernier cas, trois scenarii propres à l'évaluation de la fonction diagnostic ont été proposés. Le premier nous a permis de vérifier qu'une opération requise effectivement en cause a bien été suspectée par le diagnostic. Le deuxième scénario a été l'occasion de tester notre algorithme face à une situation de propagation multiple d'une défaillance. Le troisième a mis notre algorithme à l'épreuve d'une propagation de défaillances multiples ainsi qu'a une évolution intempestive du flux de produits. Dans tous ces cas, le mécanisme de diagnostic a

fourni des résultats confirmant nos attentes.

Comme nous avons pu le constater, les mécanismes mis en place sont en mesure de fournir un résultat de diagnostic à partir des seules informations d'un compte-rendu traduisant un dysfonctionnement de la partie opérative et d'un modèle représentant son fonctionnement normal. Ce résultat ne nécessitant aucune intervention de l'opérateur humain, il permet d'envisager très rapidement une reconfiguration du système de commande. Cependant nous devons souligner que d'une part ce résultat ne peut être fourni qu'à la condition que l'observabilité sur la partie opérative soit suffisante, sans quoi l'explosion de taille du modèle remettrait en cause la validité de l'approche. Une approche complémentaire permettant de définir le niveau d'observabilité souhaitable serait un complément incontestable à l'approche proposée. Or, la validité du résultat de diagnostic dépend très fortement des caractéristiques des opérations, connaissance acquise auprès d'un expert. De surcroît, rien ne permet de garantir à ce jour la validité de ces connaissances ; une méthode de validation doit aussi être envisagée.

Bien que le mécanisme de diagnostic appliqué à cet exemple d'application permette, in fine, une reconfiguration du système de commande, nous constatons que près de 20% (moyenne sur les trois scenarii) des services offerts par les chaînes fonctionnelles se voient leur disponibilité suspectée, alors que la responsabilité du fonctionnement incombe seulement, à un ou deux services. Ainsi, bien qu'ayant répondu à la problématique dans un temps très court et sans intervention de l'opérateur humain, il n'en demeure pas moins que la proportion de services suspectés risque de conduire une éventuelle fonction décision / synthèse dans un état d'échec. La pertinence de ces propos reste bien entendu à vérifier sur d'autres exemples auxquels des algorithmes de synthèse de lois de commande tels que ceux proposés dans (Henry, 2005) devront être appliqués.

Enfin, nous souhaiterions revenir sur deux hypothèses à la base de nos travaux :

- une chaîne fonctionnelle est toujours en mesure de fournir un compte-rendu de fonctionnement normal ou anormal : ceci ce justifie pleinement dans un cadre théorique sous l'hypothèse d'un module de contrôle/commande exempt de défaillances (cf. section 3.1 page 20). Cependant, dans un cadre réel, l'impact d'une défaillance à ce niveau entraînerait inéluctablement le blocage du module de coordination en charge de gérer ce module de contrôle/commande. Ainsi, bien que la fonction détection soit sans intérêt au niveau coordination dans le cadre théorique de nos travaux, il ne faut pas pour autant en négliger l'importance dans un contexte réel à venir,

- l'indice de confiance est binaire : afin d'améliorer la pertinence du résultat de diagnostic, nous avons souligné qu'il serait intéressant de lever cette hypothèse et ainsi définir un indice de confiance variant entre 0 et 1. Bien que ce point de vue semble intéressant à développer, il ne faut surtout pas perdre également de vue que augmenter le pouvoir d'expression de cet indice se traduira rapidement par une dégradation à la hausse de la taille du modèle généré.

Conclusion générale

Les travaux que nous avons présentés dans ce document traitent de la surveillance, de la commande et de la supervision des procédés industriels complexes. Ils font suite à ceux déjà réalisés (Zamai, 1997), (Mendez, 2002) et (Henry, 2005) dans ce domaine dans l'équipe Conduite et Optimisation (COSP) du Laboratoire d'Automatique de Grenoble (LAG) puis au Laboratoire des Sciences pour la Conception, l'Optimisation et la Production (G-SCOP).

La contribution de nos travaux réside dans la proposition d'une approche globale de suivi, de diagnostic et de mise à jour de modèles de parties opératives. L'approche proposée s'appuie sur une démarche d'acquisition de la connaissance au travers de l'interface des modèles d'opérations issus de la planification automatique. Ensuite, en ligne et en fonctionnement normal, un mécanisme de suivi construit le modèle des évolutions passées. Afin de s'affranchir du problème de la maîtrise de la taille du modèle, nous avons doté cette fonction suivi de règles d'exonération lui permettant de réduire le modèle des évolutions passées à celles de qualifiées de non suspectes. En présence d'aléas de fonctionnement retranscrits par les modules de contrôle/commande des chaînes fonctionnelles, le mécanisme de diagnostic proposé est doté de règles permettant d'évaluer avec "honnêteté", compte tenu du temps imparti et des seules informations remontées, le niveau de suspicion à attribuer aux opérations exécutées à l'origine possible du dysfonctionnement observé ainsi qu'à celles susceptibles d'avoir été affectées.

Après quoi, un processus de mise à jour du modèle de partie opérative a été proposé afin d'offrir au processus de reconfiguration un modèle "honnête" de la disponibilité des capacités opératoires.

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, comme nous l'avons souligné dans le manuscrit, la validité de l'approche dépend d'une part de la taille du modèle des évolutions passées, et d'autre part des performances temporelles des algorithmes proposés. Aussi devons nous envisager une étude théorique de ces performances. Nous avons montré au travers de l'expérimentation le fort impact sur la taille du modèle du nombre de capteurs, de leurs positions mais également du type d'information surveillée. Une étude théorique doit permettre en fonction de ces paramètres de définir une borne supérieure quant à la taille du modèle pour un ordre d'arrivée des produits donné.

– Deuxièmement, nous avons proposé une méthode de diagnostic, en ligne, s'intéressant aux SAP dont le déterminisme au flux de produits n'est pas assuré (ordre d'arrivée des produits). Cependant il existe d'autres SAP, basés quant à eux sur un flux de produits déterministe (cf. lignes de production de la Société Nationale de Roulements à Annecy). Dans ce cadre, il serait fortement question de réviser nos positions et de développer une approche hors ligne de diagnostic sur les mêmes principes que ceux proposés dans ce manuscrit.

– Troisièmement, nos travaux permettent rapidement de suspecter la disponibilité d'un ensemble de services, ainsi, serait-il intéressant d'exploiter ce résultat afin d'orienter le déclenchement de diagnostics locaux, au sein des modules de contrôle/commande des chaînes fonctionnelles.

– Quatrièmement, les résultats obtenus par les mécanismes de diagnostic ne peuvent être exploités par l'algorithme de synthèse qu'après son extension afin de prendre en compte la suspicion de la disponibilité des services. Il doit alors être en mesure de prendre plusieurs critères en compte tels que le niveau de suspicion et le temps de cycle de la loi de commande synthétisée.

– Enfin, et sur un plan technique, le module logiciel de diagnostic proposé doit être connecté au module de synthèse de lois de commande développé dans (Henry, 2005; Chuiton, 2005) afin de valider en quasi totalité la plate-forme logicielle CERBERE.

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

– Premièrement, nous avons souligné qu'il était intéressant d'étendre l'indice de confiance à une valeur comprise entre 0 et 1. Dans ce cadre et afin de conserver la validité du principe proposé, deux voies doivent être ouvertes consistant soit à définir une graduation plus fine du niveau de suspicion, soit de proposer une hiérarchisation du modèle du fonctionnement passé.

– Une deuxième perspective dégagée dans l'étude de la taille du modèle pour le diagnostic à travers l'exemple d'application, 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 de sous-ensembles de services par exemple. En effet, nous avons montré dans notre cas d'étude, l'impact du captage sur la taille du modèle. Ainsi une configuration du captage devrait pouvoir garantir l'impossibilité dans le modèle du fonctionnement passé, de propager une suspicion entre les sous-ensembles d'opérations.

– Troisièmement, il serait à notre sens pertinent d'étudier l'intérêt d'un tel concept dans le cadre de l'ordonnancement temps réel de l'architecture CIM. Il s'agirait ici d'orienter les solutions d'ordonnancement calculées par le biais des machines suspectées et non suspectes.

– 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, 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., et Biswas, G. (2003). System diagnosis using hybrid failure propagation graphs. Technical report, Vanderbilt University, États-Unis.

AFNOR (1991). *NF X50-151 Analyse de la valeur - Analyse fonctionnelle - Expression fonctionnelle du besoin et cahier des charges fonctionnel*.

Agard, B. et Tollenaere (2002). Conception d'assemblages pour la customisation de masse (design of assembly for mass customization). *Mécanique & industries*, 3(2) :113–119.

Benveniste, A., Haar, S., Fabre, E., et Jard, C. (2005). Distributed and asynchronous discrete event systems diagnosis. In *41st IEEE Conf. on Decision and Control*, pages 3742–3747.

Berruet, P. (1998). *Contribution au recouvrement des systèmes flexibles de production manufacturière : analyse de la tolérance et reconfiguration*. Thèse de doctorat, Université des Sciences et Technologies de Lille, France.

Berruet, P., Pétin, J.-F., Rigaud, F., Toguyeni, A., et Zamaï, E. (2007). Architectures de pilotage de procédés industriels. In *Technique de l'Ingénieur*, number AG 3510, pages 1–19.

Berruet, P., Toguyeni, A. K. A., Elkhattabi, S., et Craye, E. (1998). Toward an implementation of recovery procedures for fms supervision. In *IFAC Information Control Problems in Manufacturing Technology (INCOM'98)*, volume 3, pages 371–376, Nancy, France.

Boel, R. et van Schuppen, J. (2002). Decentralized failure diagnosis for discrete-event systems with costly communication between diagnosers. In *International Workshop on Discrete Event Systems - WODES'02*, Zaragoza, Espagne.

Bohez, E. et Thieravarut, M. (1997). Expert system for diagnosing computer numerically controled machines : a case study. *Computer in Industry*, 32(3) :233–248.

Boufaied, A. (2000). Etude de la fonction pronostic : cas d'ateliers flexibles de production. Mémoire de diplôme d'etudes appronfondie, Université Paul Sabatier, Toulouse, France.

Boufaied, A. (2003). *Contribution à la Surveillance Distribuée des Systèmes à Evénements Discrets Complexes*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Bouyer, P., Chevalier, F., Krichen, M., et Tripakis, S. (2005). Observation partielle des systèmes temporisés. In *Modélisation des Systèmes Réactifs*, Grenoble, France.

Chaillet, A. (1995). *Approche Multi Modèles pour la Commande et la Surveillance en Temps Réel des Systèmes à Evénements Discrets.* Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Chang, S., DiCesare, F., et Goldbogen, G. (1991). Failure propagation trees for diagnosis in manufacturing systems. *IEEE Transactions on Systems Man and Cybernetics*, 21(4) :767–776.

Chuiton, E. (2005). *Mise en œuvre d'un système de commande interprété d'ateliers de fabrication automatisés.* Mémoire cnam, Conservatoire National des Arts et Métier, Grenoble, France.

CIM (1989). A reference model for computer integrated manufacturing from the viewpoint of industrial automation. *International Journal of Computer Integrated Manufacturing*, 2(2) :114–127.

Combacau, M. (1991). *Commande et surveillance des systèmes à événements discrets complexes : application aux ateliers flexibles.* Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Combacau, M., Berruet, P., Zamaï, E., Charbonnaud, P., et Khatab, A. (2000). Supervision and monitoring of production systems. In *IFAC 2nd Conference on Management and Control of Production and Logistics (MCPL'00)*, Grenoble, France.

Combacau, M., Esteban, P., et Nketsa, A. (2005). Commandes basées réseaux de Petri. Mise en oeuvre. *Technique de l'Ingénieur*, (S7 573) :1–15.

Combacau, M., Kouiss, L., et Toguyeni, A. (2002). *Fondements du pilotage des systèmes de production.* Traité ic2 productique edition.

da Silveira, M. R. (2003). *Sur la Distribution avec redondance Partielle de Modèles à Evénements Discrets pour la Supervision de Procédés industriels.* Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

da Silveira, M. R., Combacau, M., et Subias, A. (2002). From centralized to distributed models : A systematic procedure based on petri nets. In *2002 IEEE International Conference on Systems Man and Cybernetics (SMC'02)*, Hammamet, Tunisie.

de Jonge, F., Roos, N., et Witteven, C. (2006). Primary and secondary plan diagnosis. In *The International Workshop on Principles of Diagnosis (DX'06)*, pages 145–152, Penaranda de Duero, Espagne.

de Lamotte, F. F. (2006). *Proposition d'une approche haut niveau pour la conception, l'analyse et l'implantation des systèmes reconfigurables.* Thèse de doctorat, Université de Bretagne-Sud, France.

Debouk, R., Lafortune, S., et Teneketzis, D. (2000). Coordinated decentralized protocols for failure diagnosis of discrete-event systems. *Discrete Event Dynamic Systems : Theory and Applications*, 10(1-2) :33–86.

Deschamps, E., Henry, S., et Zamaï, E. (2007a). Automatic design of control laws based on petri net formalism for complex discrete event systems. *Studies in Informatics and Control*, 16(1).

Deschamps, E., Henry, S., Zamaï, E., et Jacomino, M. (2004). Controlled system model with petri net formalism for reconfiguration. In *IFAC Conference on Manufacturing, Modelling, Management and Control (MIM'04)*, Athènes, Grèce.

Deschamps, E., Henry, S., et Zamaï, E. (2006a). Models of knowledge on manufacturing systems for control law synthesis. In *IFAC Symposium on Information Control Problems in Manufacturing (INCOM'06)*, Saint Etienne, France.

Deschamps, E., Henry, S., et Zamaï, E. (2006b). Petri nets modelling for control of discrete events systems. In *IEEE International Conference on Computational Engineering in Systems Applications (CESA'06)*, Beijing, Chine.

Deschamps, E., Henry, S., et Zamaï, E. (2006c). Synchronization of operating part model in failure context'. international conference on computational intelligence for modelling. In *International Conference on Computational Intelligence for Modelling, Control and Automation (CIMCA'06)*, Sydney, Australie.

Deschamps, E., Henry, S., et Zamaï, E. (2007b). Operating part model for on-line diagnosis. In *International Workshop on Principles of Diagnosis (DX'07)*, Nashville, Etats-Unis.

Deschamps, E. et Zamaï, E. (2007). Diagnosis for control system reconfiguration. In *IFAC Conference on Management and Control of Production and Logistics (MCPL'07)*, Sibiu, Roumanie.

Dousson, C. (1994). *Suivi d'´evolutions et reconnaissance de chroniques*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Dousson, C. et Du'o'ng, T. V. (1999). Discovering chronicles with numerical time constraints from alarm logs for monitoring dynamic systems. In *16th International Joint Conference on Artificial Intelligence IJCAI'99*, Stockholm, Suède.

Dubuisson, B. (2001). *Diagnostic, intelligence artificielle et reconnaissance des formes*. Hèrmès, Traité IC2.

Garay, M. (1979). *Computers and Intactability : a Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, New York, États-Unis.

Genc, S. et Lafortune, S. (2003). Distributed diagnosis of discrete-event systems using petri nets. In *International Conference on Application and Theory of Petri Nets*, pages 316–336, Eindhoven, Pays-Bas.

Genc, S. et Lafortune, S. (2006). A distributed algorithm for on-line diagnosis of place-bordered petri nets. In *16th IFAC World Congress*.

Ghallab, M., Nau, D., et Traverso, P. (2004). *Automated Planning, Theory and Practice*. Elsevier.

Gouyon, D., Petin, J. F., et Morel, G. (2004). Control synthesis for product-driven automation. In *7th IFAC Workshop on Discrete Event Systems (WODES'04)*, Reims, France.

Granier, P. (2003). Développement d'un outil de supervision de procédés industriels sous l'environnement tornado/vxworks. application au procédé pilote saphir.

Hamidi, K. (2005). *Contribution à un modèle d'évaluation quantitative des performances fiabilistes de fonctions électroniques et programmables dédiés à la sécutité.* Thèse de doctorat, Institut National Polytechnique de Lorraine, France.

Hamscher, W., Consol, L., et Kleer, J. D. (1992). *Readings in Model-Based Diagnosis.* Morgan Kaufmann, San Mateo, CA, Etats-Unis.

Henry, S. (2005). *Synthèse de Lois de Commande pour la Configuration et la Reconfiguration des Systèmes Industriels Complexes.* Thèse de doctorat, Institut National Polytechnique de Grenoble, France.

Henry, S., Deschamps, E., Zamaï, E., et Jacomino, M. (2004a). Control law synthesis algorithm for discrete-event systems. In *IFAC Conference on Management and Control of Production and Logistics (MCPL'04)*, Santiago, Chili.

Henry, S., Zamaï, E., et Jacomino, M. (2003). Decisional requirements for supervision, monitoring and control structures. In *IEEE International Conference on Computational Engineering in Systems Applications (CESA'03)*, Lille, France.

Henry, S., Zamaï, E., et Jacomino, M. (2004b). Real time reconfiguration of manufacturing system. In *IEEE International Conference on System Man and Cybernetic (SMC'04)*, La Haye, Pays Bas.

Henry, S., Zamaï, E., et Jacomino, M. (2005). Controlled system model adapted for control law synthesis. In *16th IFAC World Congress (IFAC'05)*, Prague, Républic Tchèque.

Hu, W., Starr, A., et Leung, A. (1999). Two diagnostic models for plc controlled flexible manufacturing systems. *International Journal of Machine Tools & Manufacture*, 39 :1979–1991.

Huvenoit, B., Craye, E., et Gentina, J. (1992). Elaboration de la commande de cellules de production flexibles dans l'industrie manufacturière. In *Conférence Automatisation Industrielle*, volume 1, pages 6.21–6.25, Montréal, Canada.

Jacquet, L., Sallez, Y., et Soenen, R. (1995). Toward a specification procedure of operational functions for an automated system. In *IEEE international Conference on Systems, Man and Cybernetics*, volume 5, pages 4480–4485, Vancouver, Canada.

Jones, A. et Saleh, A. (1989). A multi-layer/multi-level control architecture for computer integrated manufacturing system. *IECON 89*, pages 519–525.

Kahn, P. (2004). Normalisation en matière de sûreté de fonctionnement des logiciels. *Techniques de l'ingénieur*, (SE2510).

Kempowsky, T., Subias, A., et Aguillar-Martin, J. (2004). Supervision of complex processes : Strategy for fault detection and diagnosis. In *IFAC Conference on Management and Control of Production and Logistics (MCPL'04)*, Santiago, Chili.

Kleer, J. D. et Williams, B. (1989). Diagnosing with behavioral modes. In *the 11th International Joint Conference on Artificial Intelligence, IJCAI-89*, pages 1324–1330, Détroit, Etats-Unis.

Lafortune, S., Sampath, D. T. M., Sengupta, R., et Sinnamohideen, K. (2001). Failure diagnosis of dynamic systems : An approach based on discrete event systems. In *American Control Conference (ACC'01)*, pages 2058–2071.

Mauser, N. (2006). Prise en compte des opérations de surveillance pour la supervision et la commande des procédés industriels complexes. Mémoire de diplôme d'etudes appronfondie, INP-Grenoble, France.

Mehrabi, M. G., Ulsoy, A. G., et Koren, Y. (2000). Reconfigurable manufacturing systems : Key to future manufacturing. *Journal of Intelligent Manufacturing*, 11 :403–419.

Mendez, H. (2002). *Synthèse de lois de surveillance pour les procédés industriels complexes.* Thèse de doctorat, Institut National Polythechnique de Grenoble, France.

Mendez, H., Zamaï, E., et Descotes-Genon, B. (2003). Quality, productivity, security and ecological constraints for synthesis of monitoring laws. In *5eme Congrès International Pluridisciplinaire Qualité et Sûreté de Fonctionnement (QUALITA 2003)*, Nancy, France.

Murand, F. (à paraître en 2007). *Mise en œuvre des mécanismes de suivi temps réel et de diagnostic de services.* Mémoire cnam, Conservatoire National des Arts et Métier, Grenoble, France.

Noureddine, M. (2005). Système d'aide au pilotage des flux de production par les systèmes multi-agents. In *4th International Conference Integrated Design and Production*, Casablanca, Maroc.

O'Grady, P., Kim, Y., et Young, R. (1994). Conception d'assemblages pour la customisation de masse (design of assembly for mass customization). *International Journal of Computer-Integrated Manufacturing*, 7(3) :152–1662.

Pencolé, Y. (2002). *Diagnostic décentralisé de systèmes à événements discrets : application aux réseaux de télécommunications.* Thèse de doctorat, Université de Rennes 1, France.

Perrin, J., Binet, F., Dumery, J., Merlaud, C., et Trichard, J. (2004). *Autmatique et informatique industrielle : Bases théoriques, méthodologiques et techniques.* Nathan Technique.

Philippot, A. (2006). *Contribution au diagnostic décentralié des systèmes à évènements discrets : application aux systèmes manufacturiers.* Thèse de doctorat, Université de Reins Champagne Ardenne, France.

Pickard, K., Muller, P., et Bertsche, B. (2005). Multiple failure mode and effects analysis - an approach to risk assessment of multiple failures with fmea. In *Annual Reliability and Maintainability Symposium*, pages 457–462, Alexandria, États-Unis.

Qiu, W. et Kumar, R. (2006). Decentralized failure diagnosis of discrete event systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 36(2) :384–395.

Reiter, R. (1987). A theory of diagnosis from first principle. *Artificial Intelligence*, 32(1) :57–96.

Sampath, M., Sengupta, R., et S.Lafortune (1996). Failure diagnosis using discrete event models. *IEEE Transactions on Control Systems Technology*, 4(2) :105–124.

Sampath, M., Sengupta, R., S.Lafortune, et Teneketzis, D. (1995). Diagnosability of discrete event systems. *IEEE Transaction on Automatic Control*, 40(9) :1555–1575.

Soldani, S., Combacau, M., Thomas, J., et Subias, A. (2006). Intermittent fault detection through message exchanges : a coherence based approach. In *The International Workshop on Principles of Diagnosis (DX'06)*, pages 251–256, Penaranda de Duero, Espagne.

Sourisse, C. et Boudillon, L. (1997). *La sécurité des machines automatisées. Tome 2 : Techniques et moyens de prévention opératifs -Systèmes de commande - Utilisation des machines*. Groupe Schneider, collection technique edition.

Toguyeni, A., Berruet, P., et Craye, E. (2003). Models and algorithms for failure diagnosis and recovery in fmss. *International Journal of Flexible Manufacturing Systems*, 15 :57–85.

Toguyeni, A., Craye, E., et Gentina, J. (1996). A framework to design a distributed diagnosis in fms. In *IEEE international Conference on Systems, Man and Cybernetics*, volume 4, pages 2774–2779, Beijing, Chine.

Trentesaux, D. et Sénéchal, O. (2002). Conduite des systèmes de production manucfacturière. In *Techniques de l'ingénieur, traité Informatique Industrielle*, number S7598.

Tromp, L. (2000). *Surveillance et diagnostic de systèmes industriels complexes : une approche hybride numérique/symbolique*. Thèse de doctorat, Université de Rennes 1, France.

Valette, R. et Künzle, L. (1994). Réseaux de petri pour la détection et le diagnostic. Journées nationales : Sûreté, surveillance, supervision.

Vallée, F. (2003). Sécurité informatique pour la gestion des risques. *Technique de l'Ingénieur*, (SE2500).

Witteveen, C., Roos, N., van der Krogt, R., et de Weerdt, M. (2005). Diagnosis of single and multi-agent plans. In *the 4th international conference on Autonomous Agents and Multi-Agent Systems*, pages 805–812, Utrecht, Nouvelle-Zélande.

Zamai, E. (1997). *Architecture de surveillance-commande pour les systèmes à événements discrets complexes*. Thèse de doctorat, Université Paul Sabatier, Toulouse, France.

Zamai, E., Subias, A., et Combacau, M. (1998). An architecture for control and monitoring of discrete events systems. *Computers in Industry*, 36(1, 2).

Zaytoon, J. (1996). Specification and design of logic controllers for automated manufacturing systems. *Robotics and Computer Integrated Manufacturing*, 12(4) :353–366.

Zaytoon, J., Ndjab, C., et Carré-Ménétrier, V. (1999). Grafcet et graphe d'états : Synthèse hors ligne de la commande. *APII-JESA Journal Européen des Systèmes Automatisés*, 33(7) :783–814.

Zwingelstein, G. (1995). *Diagnostic des défaillances : Théorie et pratique pour les systèmes industriels*. Hermes, traité des nouvelles technologies edition.

Zwingelstein, G. (1999). Sûreté de fonctionnement des systèmes industriels complexes. *Techniques de l'ingénieur, traité Informatique Industrielle*, (S8250).

Annexes

Annexe A

Formalisation du comportement des opérations

Cette annexe présente la formalisation des différents types d'opération, à l'exception des opérations d'action déjà présentées dans le manuscrit.

1 Formalisation du comportement des opérations d'induite

Nous adopterons pour les informations décrites dans le chapitre 4 les notations présentées par la figure A.1. Une opération $Oind_i$ est composée de :

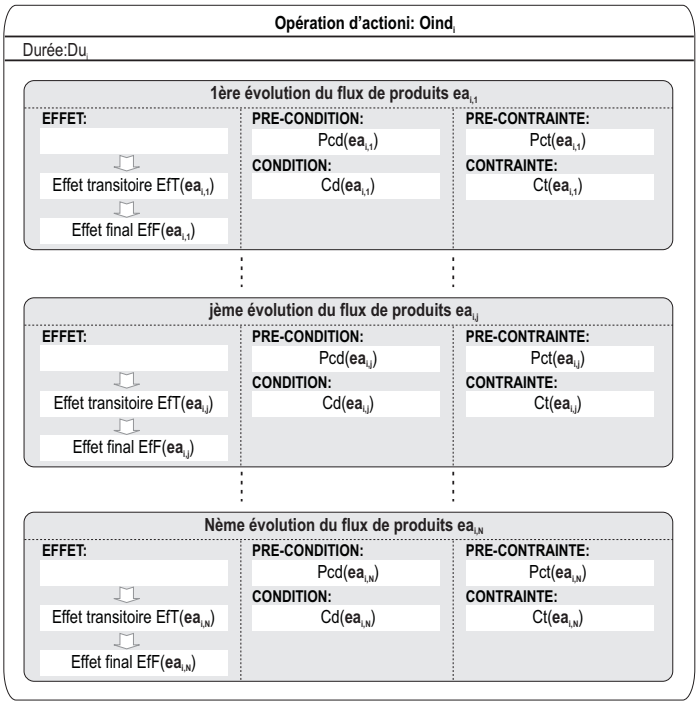


FIG. A.1 – Notation opération induite

1. Du_i, la durée de l'opération $Oind_i$.

2. $ea_{i,j}$ pour $i \in [1, N_i]$, les évolutions associées du flux de produits, pour une opération $Oind_i$ avec N_i évolutions possibles du flux de produits. Chaque évolution $ea_{i,j}$ est elle même composée de :
   - $EfT(ea_{i,j})$, l'effet transitoire sur le flux de produits.
   - $EfF(ea_{i,j})$, l'effet final sur le flux de produits.
   - $PCd(ea_{i,j})$, la pré-condition à respecter avant le lancement de l'opération pour que l'effet transitoire $EfT(ea_{i,j})$ soit réalisé.
   - $Cd(ea_{i,j})$, la condition à respecter pendant l'exécution de l'opération pour que l'effet final $EfF(ea_{i,j})$ soit réalisé.
   - $PCt(ea_{i,j})$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération si la pré-condition $PCd(ea_{i,j})$ est vraie.
   - $Ct(ea_{i,j})$, la contrainte, à respecter durant toute l'évolution $ea_{i,j}$.

1.1 Formalisation des évolutions du flux de produits

L'effet transitoire sur l'état du flux de produits $EfF(ea_{i,j})$ pour $j \in [1, Ni]$ correspondant à l'évolution associée $ea_{i,j}$ de l'opération $Oind_i$ est réalisé si :

- la pré-contrainte $PCt(ea_{i,j})$ et la pré-condition $PCd(ea_{i,j})$ de $ea_{i,j}$ sont respectées,
- les pré-contraintes des autres évolutions associées du flux de produits qui doivent être réalisées sont respectées, autrement dit les pré-contraintes $PCt(ea_{i,l})$ des évolutions $ea_{i,l}$ dont les pré-conditions $PCd(ea_{i,l})$ sont vraies pour $l \in [1, N_i] \setminus j$.

De surcroît nous considérerons que si l'effet $EfT(ea_{i,l})$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition A.1 *Effet transitoire d'une évolution du flux de produits*

$$PCd(ea_{i,j}) \wedge PCt(ea_{i,j}) \wedge PCt(ec_i) \bigwedge_{\forall l \in [1,N_i] \setminus j} \Big[PCd(ea_{i,l}) \Rightarrow PCt(ea_{i,l}) \Big] \Leftrightarrow EfT(ea_{i,j})$$

L'effet final sur l'état du flux de produits $EfF(ea_{i,j}) \forall j \in [1, Ni]$ correspondant à l'évolution $ea_{i,j}$ de l'opération Oa_i est réalisé si :

- l'effet transitoire $EfT(ea_{i,j})$ est réalisé,
- la contrainte $Ct(ea_{i,j})$ et la condition $Cd(ea_{i,j})$ sont respectées,
- si les contraintes des autres évolutions associées du flux de produits qui sont réalisées sont respectées.

De plus nous considérerons que si l'effet $EfF(ea_{i,l})$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition A.2 *Effet transitoire d'une évolution du flux de produits*

$$EfT(ea_{i,j}) \wedge Cd(ea_{i,j}) \wedge Ct(ea_{i,j}) \bigwedge_{\forall lj \in [1,N_i] \setminus j} \Big[PCd(ea_{i,l}) \Rightarrow Ct(ea_{i,l}) \Big] \Leftrightarrow EfF(ea_{i,j})$$

2 Formalisation du comportement des opérations d'acquisition

Nous adopterons pour les informations décrites dans le chapitre 4 les notations présentées par la figure A.2. Une opération est composée de :

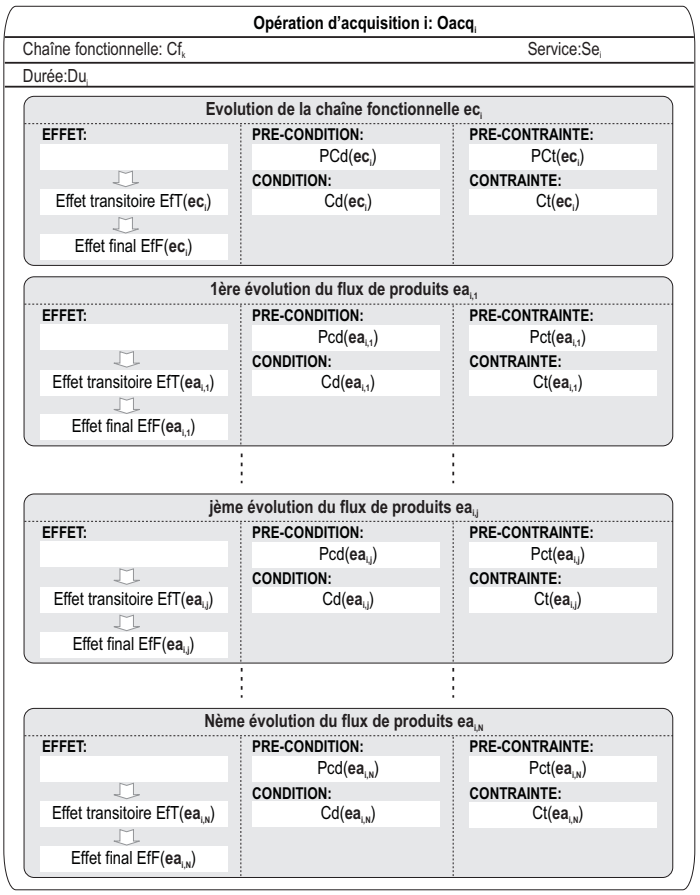


FIG. A.2 – Notation opération d'acquisition

La structure d'une opération d'acquisition étant identique à celle d'une opération d'action, elle bénéficiera de la même formalisation du comportement.

3 Formalisation du comportement des opérations requises

Nous adopterons pour les informations décrites dans le chapitre 4 les notations présentées par la figure A.3. Une opération $Oreq_i$ est composée de :

1. Du_i, la durée de l'opération $Oreq_i$, si elle est connue.
2. ee_i, l'évolution de l'environnement si sa connaissance est nécessaire au niveau coordination considéré, elle-même composée de :
   - $EfT(ee_i)$, l'effet transitoire sur l'environnement.
   - $EfF(ee_i)$, l'effet final sur l'environnement.
   - $PCd(ee_i)$, la pré-condition à respecter avant le lancement de l'opération pour que l'effet transitoire sur l'environnement soit réalisé.

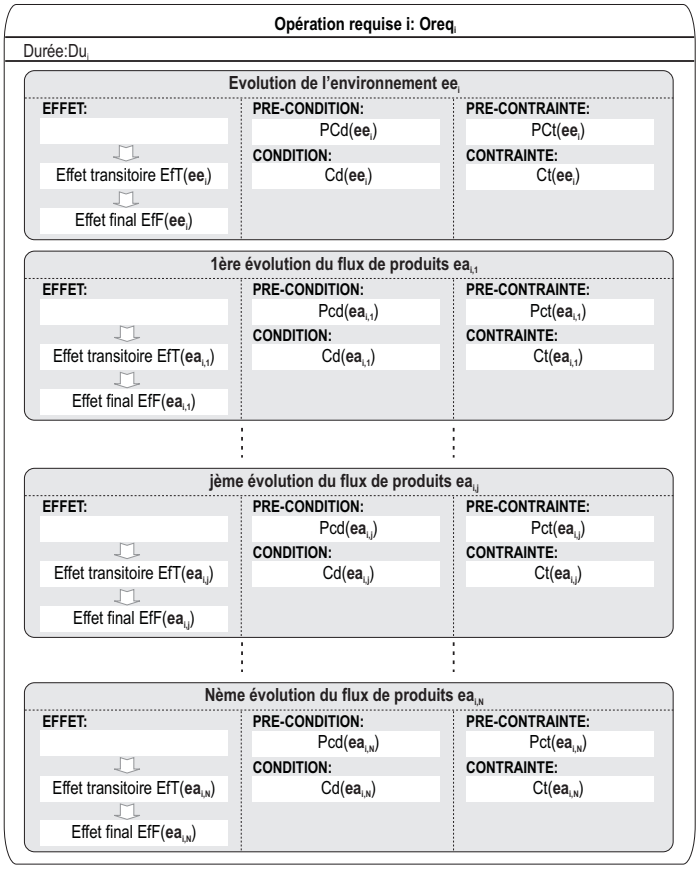


FIG. A.3 – Notation opération requise

- $Cd(ee_i)$, la condition à respecter pendant l'exécution de l'opération pour que l'effet final sur l'environnement soit réalisé.
- $PCt(ee_i)$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération si la pré-condition $PCd(ee_i)$ est vraie.
- $Ct(ee_i)$, la contrainte, à respecter durant l'exécution de l'opération.

3. $ea_{i,j}$ pour $i \in [1, N_i]$, les évolutions associées du flux de produits, pour une opération $Oreq_i$ avec N_i évolutions possibles du flux de produits. Chaque évolution $ea_{i,j}$ est elle même composée de :

   - $EfT(ea_{i,j})$, l'effet transitoire sur le flux de produits.
   - $EfF(ea_{i,j})$, l'effet final sur le flux de produits.
   - $PCd(ea_{i,j})$, la pré-condition à respecter avant le lancement de l'opération pour que l'effet transitoire $EfT(ea_{i,j})$ soit réalisé.
   - $Cd(ea_{i,j})$, la condition à respecter pendant l'exécution de l'opération pour que l'effet final $EfF(ea_{i,j})$ soit réalisé.
   - $PCt(ea_{i,j})$, la pré-contrainte à respecter sur l'état du flux de produits et des chaînes fonctionnelles avant le début de l'opération si la pré-condition $PCd(ea_{i,j})$ est vraie.
   - $Ct(ea_{i,j})$, la contrainte, à respecter durant toute l'évolution $ea_{i,j}$.

3.1 formalisation de l'évolution de l'environnement

L'effet transitoire sur l'état de l'environnement $EfT(ee_i)$ correspondant à l'évolution ee_i de l'opération $Oreq_i$ est réalisé si :

- l'opération $Oreq_i$ est disponible à son lancement. Cette disponibilité est notée $\neg AN(Oreq_i(2x-1))$, où x est la x^{ime} fois que l'opération induite est exécutée,
- la pré-contrainte $PCt(ee_i)$ et la pré-condition $PCd(ee_i)$ sont respectées,
- les pré-contraintes des évolutions associées du flux de produits qui doivent être réalisées sont respectées, c'est à dire les pré-contraintes $PCt(ea_{i,j})$ des évolutions $ea_{i,j}$ dont les pré-conditions $PCd(ea_{i,j})$ sont vraies. En effet le non respect d'une de ces contraintes entraînera forcément une mauvaise exécution de l'opération dans sa globalité.

De surcroît nous considérerons que si l'effet $EfT(ee_i)$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition A.3 *Effet transitoire de l'évolution de l'environnement*

$$\neg AN(Oreq_i(2x-1)) \wedge PCd(ee_i) \wedge PCt(ee_i) \bigwedge_{\forall j \in [1,N_i]} \Big[PCd(ea_{i,j}) \Rightarrow PCt(ea_{i,j})\Big] \Leftrightarrow EfT(ee_i)$$

L'effet final sur l'état de l'environnement $EfF(ee_i)$ correspondant à l'évolution ee_i de l'opération $Oreq_i$ est réalisé si :

- l'opération $Oreq_i$ est disponible durant l'opération. Cette disponibilité est notée $\neg AN(Oreq_i(2x))$,
- l'effet transitoire $EfT(ee_i)$ est réalisé,
- la contrainte $Ct(ee_i)$ et la condition $Cd(ee_i)$ sont respectées,
- si les contraintes des évolutions associées du flux de produits qui sont réalisées sont respectées, c'est à dire les conditions $Cd(ea_{i,j})$ et les contraintes $Ct(ea_{i,j})$ des évolutions $ea_{i,j}$ dont les pré-conditions $PCd(ea_{i,j})$ sont vraies.

De surcroît nous considérerons que si l'effet $EfF(ee_i)$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition A.4 *Effet final de l'évolution de l'environnement*

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

3.2 Formalisation des évolutions associées du flux de produits

L'effet transitoire sur l'état du flux de produits $EfF(ea_{i,j})$ pour $j \in [1, Ni]$ correspondant à l'évolution associée $ea_{i,j}$ de l'opération $Oreq_i$ est réalisé si :

- l'opération $Oreq_i$ est disponible,
- la pré-contrainte $PCt(ea_{i,j})$ et la pré-condition $PCd(ea_{i,j})$ de $ea_{i,j}$ sont respectées,
- la pré-contrainte $PCt(ee_i)$ et la pré-condition $PCd(ee_i)$ de ee_i sont respectées, car sans effet sur l'environnement il ne peut y avoir un effet sur le flux de produits,

– les pré-contraintes des autres évolutions associées du flux de produits qui doivent être réalisées sont respectées, c'est à dire les pré-contraintes $PCt(ea_{i,l})$ des évolutions $ea_{i,l}$ dont les pré-conditions $PCd(ea_{i,l})$ sont vraies pour $l \in [1, N_i] \setminus j$.

De surcroît nous considérerons que si l'effet $EfT(ea_{i,l})$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition A.5 *Effet transitoire d'une évolution du flux de produits*

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

L'effet final sur l'état du flux de produits $EfF(ea_{i,j}) \forall j \in [1, Ni]$ correspondant à l'évolution $ea_{i,j}$ de l'opération $Oreq_i$ est réalisé si :

– l'opération $Oreq_i$ est disponible durant l'opération,
– l'effet transitoire $EfT(ea_{i,j})$ est réalisé,
– la contrainte $Ct(ea_{i,j})$ et la condition $Cd(ea_{i,j})$ sont respectées,
– la contrainte $Ct(ee_i)$ et la condition $Cd(ee_i)$ sont respectées,
– si les contraintes des autres évolutions associées du flux de produits qui sont réalisées sont respectées.

De surcroît nous considérerons que si l'effet $EfF(ea_{i,l})$ est réalisé l'ensemble des conditions ci-dessus sont vraies.

Ceci nous donne donc :

Définition A.6 *Effet transitoire d'une évolution du flux de produits*

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

3.3 Impact du non respect des pré-contraintes et contraintes

Pour le diagnostic, une des extensions proposées dans le cadre de la modélisation est la prise en compte de l'impact des pré-contraintes et contraintes d'une opération sur le fonctionnement de l'opération requise (disponibilité).

La non satisfaction d'une pré-contrainte de l'évolution de l'environnement ($PCt(ee_i)$), ou des évolutions associées du flux de produits qui doivent être réalisées ($PCt(ea_{i,j})$ pour j tel que $PCd(ea_{i,j})$ est vraie) ont pour conséquence potentielle d'entraîner un fonctionnement anormal de l'environnement (partie du système complet exécutant l'opération requise) dès le lancement de l'opération :

Définition A.7 *Impact du non respect des pré-contraintes*

$$\neg PCt(ee_i) \bigvee_{\exists j \in [1,Ni]} \neg \Big[PCd(ea_{i,j}) \Rightarrow (PCt(ea_{i,j}) \Big] \Rightarrow AN(Oreq_i(2x-1))$$

De la même manière, la non satisfaction d'une contrainte de l'évolution de l'environnement ($Ct(ee_i)$), ou des évolutions associées du flux de produits qui doivent être réalisées ($Ct(ea_{i,j})$ pour j tel que $PCd(ea_{i,j})$ est vraie) ont pour conséquence potentielle d'entraîner une fonctionnement anormal de l'environnement (partie du système complet exécutant l'opération requise) correspondante durant l'exécution de l'opération :

Définition A.8 *Impact du non respect des contraintes*

$$\neg Ct(ee_i) \bigvee_{\exists j \in [1,Ni]} \neg \Big[PCd(ea_{i,j}) \Rightarrow Ct(ea_{i,j}) \Big] \Rightarrow AN(Oreq_i(2x))$$

Annexe B

Intégration des opérations de surveillance dans la génération de la lois de commande

1 Lancement d'une opération de surveillance

En général les valeurs de la variable d'état à surveiller définissent dans quel état de la partie opérative une opération de surveillance est utilisable. En revanche, l'instant de lancement de l'opération de surveillance est différent selon le type de surveillance.
Pour une surveillance incluant celle du comportement temporel de l'effet modifiant la variable d'état à surveiller, l'opération de surveillance doit être lancée simultanément avec l'opération associée. Car non seulement une variable d'état doit être vérifiée mais aussi le temps au bout duquel la variable doit changer de valeur. L'exemple de la figure B.1 présente deux opérations de surveillance qui sont lancées différemment.

L'opération de surveillance correspondant à la présence d'une pièce en B est lancée à l'instant ou l'opération induite *transfert de la position A à B* est lancée. La durée du transfert est indiquée dans la requête de l'opération de surveillance pour le module de pilotage du capteur de surveillance. Ainsi le module connaît l'instant prévu pour l'arrivée de la pièce. Si la pièce arrive en B trop tôt ou trop tard, ce dysfonctionnement est détecté et le module de pilotage génère un compte-rendu de fonctionnement anormal au niveau coordination. L'instant du lancement de l'opération de surveillance est déduit à partir des (pré-)conditions et (pré-)contraintes. L'opération de surveillance correspondant à la présence d'une pièce en B n'a pas de (pré-)conditions ou (pré-)contraintes, (cf. figure B.1) ; elle doit donc être lancée en même temps que l'opération qui amène la pièce de A vers B.
L'opération de surveillance pour la vérification d'un état du produit (une caractéristique physique) peut être lancée si toutes les conditions, et (pré-)contraintes sont respectées. Dans l'exemple, la surveillance du perçage de la pièce en B est seulement possible si la pièce est devant le capteur, donc la (pré-)condition est *PièceEnB == non vide*. Le lancement de l'opération de surveillance est en conséquence seulement possible après la fin de l'opération induite.

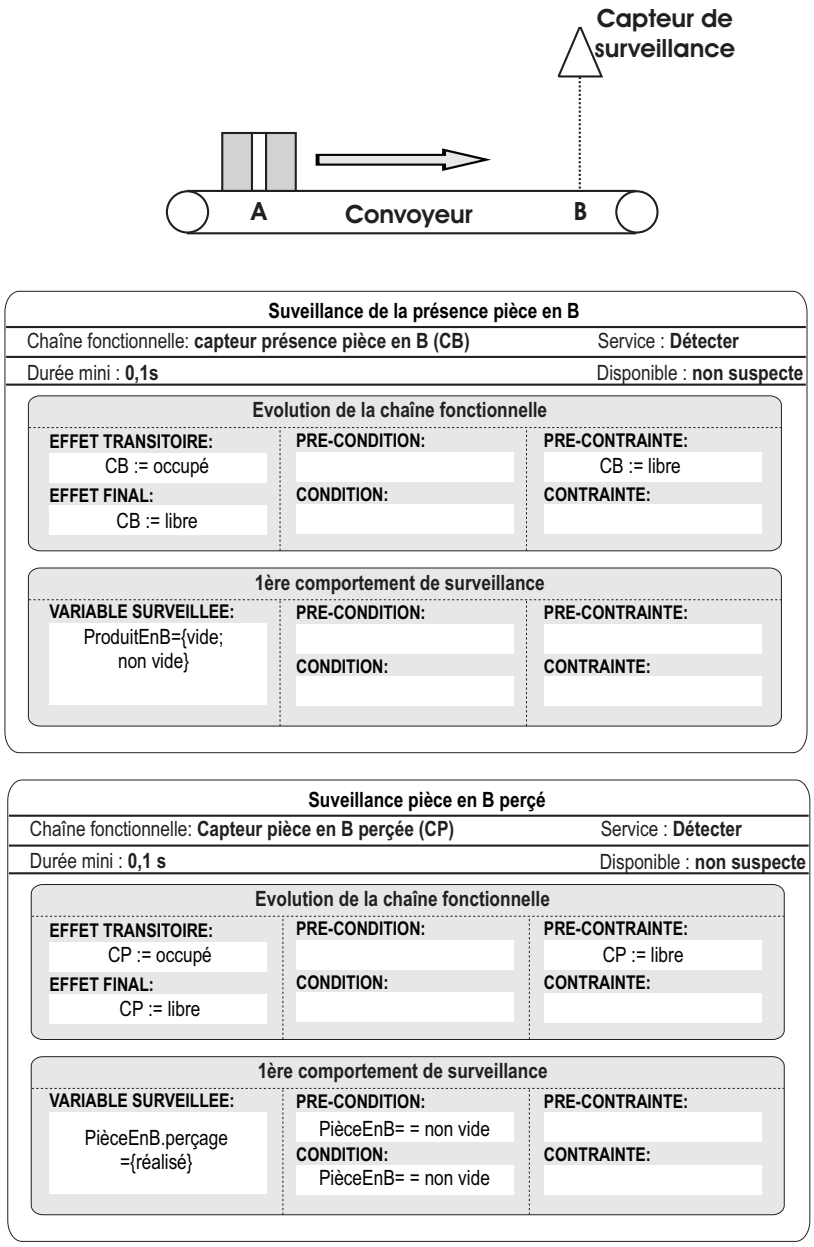


FIG. B.1 – Un exemple sur le lancement des opérations de surveillance

2 Utilisation des opérations de surveillance

L'utilisation des opérations de surveillance dégrade en général la performance temporelle de la loi de commande (l'arrêt du flux de produits ou la déviation du flux au lieu de surveillance) et donc impacte a priori la productivité. Cependant, les opérations de surveillance permettent de détecter des déviations de comportement du processus de production au plus tôt, autorisant ainsi des reconfigurations et permettent le maintient de la productivité. Toutefois ce principe ne s'applique pas à toutes les opérations de surveillance. En effet, si nous prenons les opérations de surveillance qui vérifient la présence d'un produit à une position, ces dernières peuvent être exécutées en temps masqué. Elles sont en effet parallélisées avec les autres évolutions.

Les surveillances des caractéristiques produits influençant le temps de cycle global sont à demander par le niveau supérieur du niveau coordination, car il impose les spécifications du produit (et aussi la vérification de leur qualité) et le temps admis pour transformer le produit selon ces spécifications. C'est donc le niveau supérieur qui informe le niveau coordination via les objectifs de production de la nécessité de vérifier certaines caractéristiques physiques.

3 Étude de l'intégration des opérations de surveillance

Les opérations de surveillance qui vérifient des caractéristiques physiques des produits peuvent nécessiter comme les autres opérations du système (transformation ou transitique) des opérations de préparation. Une mesure valable par exemple pourrait être seulement possible sur un produit indexé et bridé. Il peut s'avérer nécessaire de lancer des opérations de préparation dans la loi de commande avant de commencer les opérations de surveillance ou d'attendre le lancement d'une autre opération jusqu'à ce que l'opération de surveillance soit terminée. Les opérations de surveillance sont donc à synchroniser avec les opérations gérées nativement dans la loi de commande. Ceci justifie la prise en compte de ces opérations de surveillance pendant la synthèse de la loi de commande. Cela entraîne donc une modification de l'algorithme de synthèse. Avec les opérations de surveillance un troisième type d'opération est désormais à prendre en compte. Il faut l'intégrer dans l'algorithme de synthèse. L'algorithme de synthèse proposé dans (Henry, 2005) propose un découpage de la synthèse de loi de commande en trois étapes. Trouver tout d'abord la séquence des opérations de transitique, puis dans un second temps les opérations de transformations des caractéristiques produit qui doivent être insérées dans la séquence. Enfin dans un troisième temps, les opérations de préparation sont à ajouter à cette dernière séquence. Il est donc nécessaire de déterminer dans quelles étapes de l'algorithme de synthèse de lois de commande, les opérations de surveillance sont à intégrer.

La construction d'un graphe d'état commun à ces deux étapes, sur lequel se base la synthèse de lois de commandes, peut devenir rapidement complexe. Lorsque la fabrication du produit n'impose pas beaucoup de contraintes de précédence pour les transformations, presque toutes les combinaisons des valeurs des variables d'état concernant les caractéristiques physiques du produit sont possibles et donc la taille de ce graphe augmente considérablement. Afin de vérifier que la fusion est réaliste, une étude de complexité a été réalisée.

La comparaison de la complexité du graphe commun avec le graphe qui prend en compte seulement les opérations de transformation montre que le nombre des sommets est augmenté par un facteur multiplicatif de *nombre de positions du système*. Le nombre d'arcs augmente avec un facteur additif de *nombre de comportements de transitique* * $2^{(\textit{nombre de variables d'état correspondants aux caractéristiques physiques du produit})}$. La complexité ajoutée en fusionnant le graphe des opérations de transformations avec le graphe des opérations de transitique reste donc raisonnable par rapport aux graphes séparés. L'utilisation d'un graphe d'état commun est donc envisageable. La suite de la synthèse de loi de commande correspond à l'algorithme conçu dans (Henry, 2005).

Annexe C

Description des capacités opératoires du système de vernissage

Le lecteur trouvera dans cette annexe la description complète du cas d'étude utilisé dans la partie 3 de ce manuscrit.

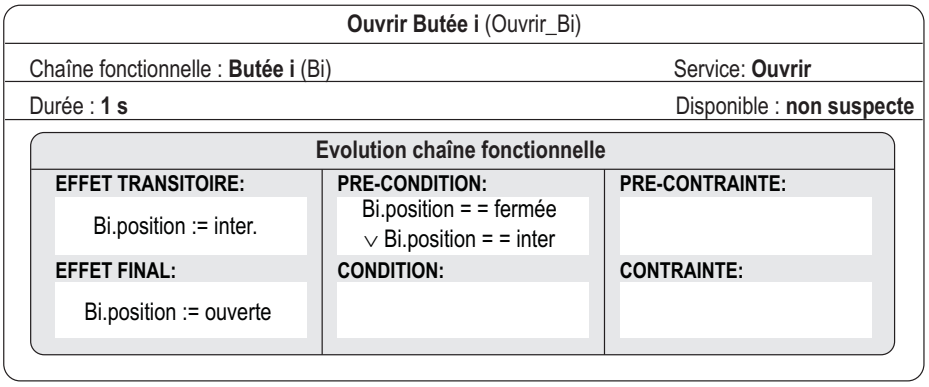


FIG. C.1 – Opération d'action *Ouvrir Butée i*

Fermer Butée i (Fermer_Bi)

Chaîne fonctionnelle : **Butée i** (Bi) — Service: **Fermer**

Durée : **1 s** — Disponible : **non suspecte**

Evolution chaîne fonctionnelle

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
Bi.position := inter.	Bi.position = = ouverte ∨ Bi.position = = inter	(*)
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
Bi.position := fermée		(*)

(*) si i=1 : CarteEnA-B = = vide
si i=2 : CarteEnB-C = = vide
si i=3 : CarteEnC-D = = vide
si i=4 : CarteEnD-E = = vide
si i=5 : CarteEnF-G = = vide
si i=6 : CarteEnG-H = = vide

FIG. C.2 – Opération d'action *Fermer Butée i*

Sortir Vérin de tranfert de carte (Sortir_Vt)

Chaîne fonctionnelle : **Vérin de tranfert** (Vt) — Service: **Sortir**

Durée : **5 s** — Disponible : **non suspecte**

Evolution chaîne fonctionnelle

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
Vt.position := inter.	Vt.position = = rentrée ∨ Vt.position = = inter	CarteEnB-C = = vide ∧ CarteEnC-D = = vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
V1.position := sortie		CarteEnB-C = = vide ∧ CarteEnC-D = = vide

1ere Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
CarteEnC → CarteEnC-F	CarteEnC = = non vide	CarteEnC-F = = vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnC-F → CarteEnF		CarteEnF = = vide

2ème Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
	CarteEnC-F = = non vide	
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnC-F → CarteEnF		CarteEnF = = vide

FIG. C.3 – Opération d'action *Sortir vérin de transfert de carte*

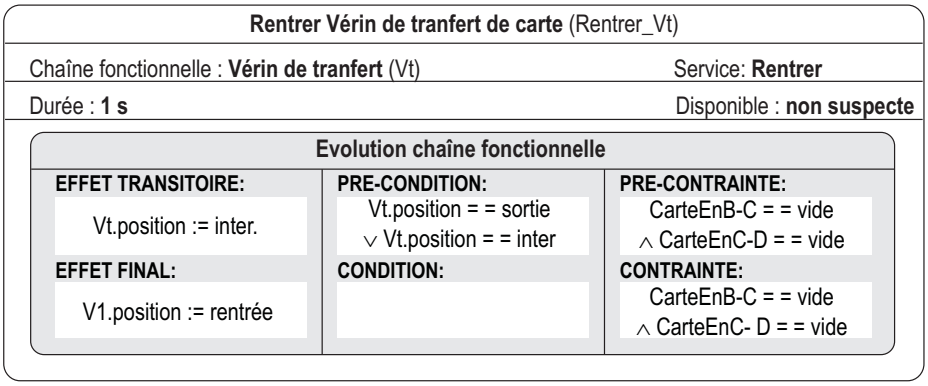
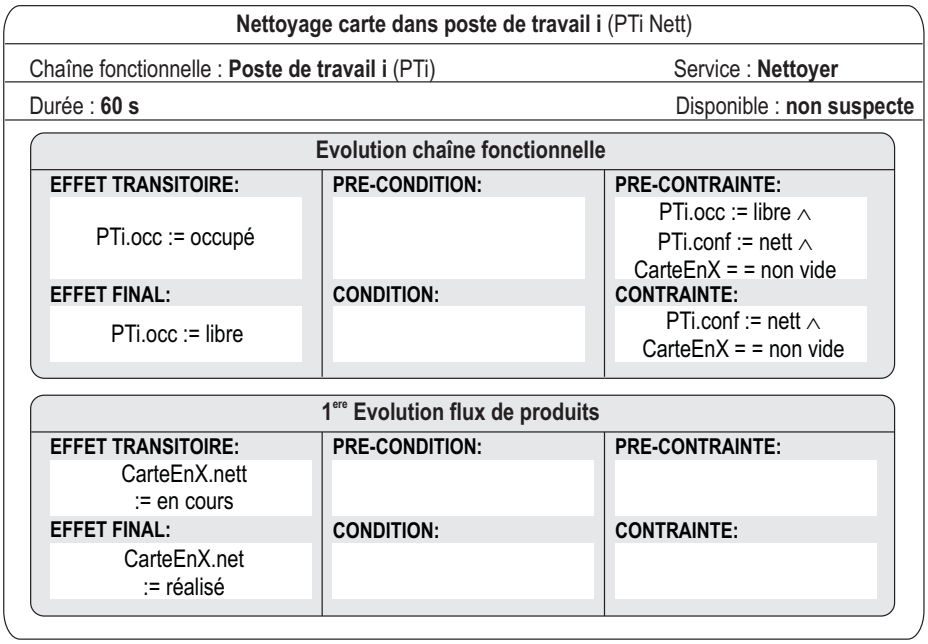


FIG. C.4 – Opération d'action *Rentrer vérin de transfert de carte*



si i=1 : X = B
si i=2 : X = D
si i=3 : X = G

FIG. C.5 – Opération d'action *Nettoyage carte dans poste de travail i*

Vernissage carte dans poste de travail i (PTi Vernis)

Chaîne fonctionnelle : **Poste de travail i** (PTi) — Service : **Vernir**

Durée : **120 s** — Disponible : **non suspecte**

Evolution chaîne fonctionnelle

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
PTi.occ := occupé		PTi.occ := libre ∧ PTi.conf := vernis ∧ CarteEnX = = non vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
PTi.occ := libre		PTi.conf := vernis ∧ CarteEnX = = non vide

1ere Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
CarteEnX.vernis := en cours		CarteEnX.nett = = réalisé ∧ CarteEnX.vernis = = non r.
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnX.vernis := réalisé		

si i=1 : X = B
si i=2 : X = D
si i=3 : X = G

FIG. C.6 – Opération d'action *Vernissage carte dans poste de travail i*

Configuration du poste de travail i pour nettoyage (PTi Conf Nett)

Chaîne fonctionnelle : **Poste de travail i** (PTi) — Service : **Pré Nett**

Durée : **30 s** — Disponible : **non suspecte**

Evolution chaîne fonctionnelle

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
PTi.occ := occupé PTi.conf := en cours		PTi.occ = = libre
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
PTi.occ := libre PTi.conf := nett		

FIG. C.7 – Opération d'action *Configuration du poste de travail i pour nettoyage carte*

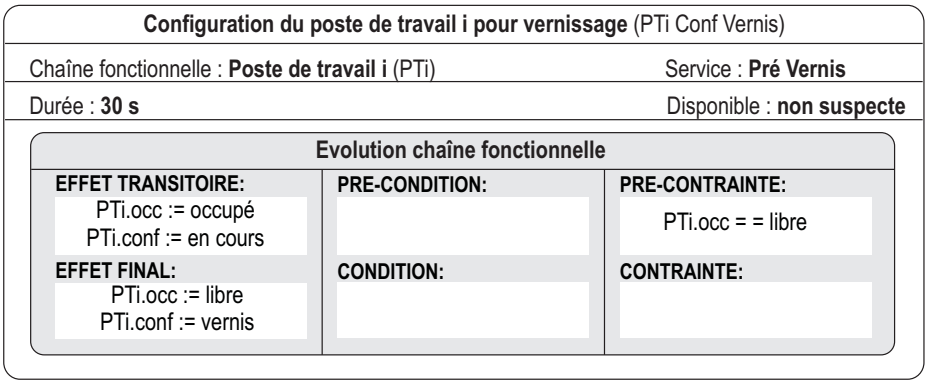


FIG. C.8 – Opération d'action *Configuration du poste de travail i pour vernissage carte*

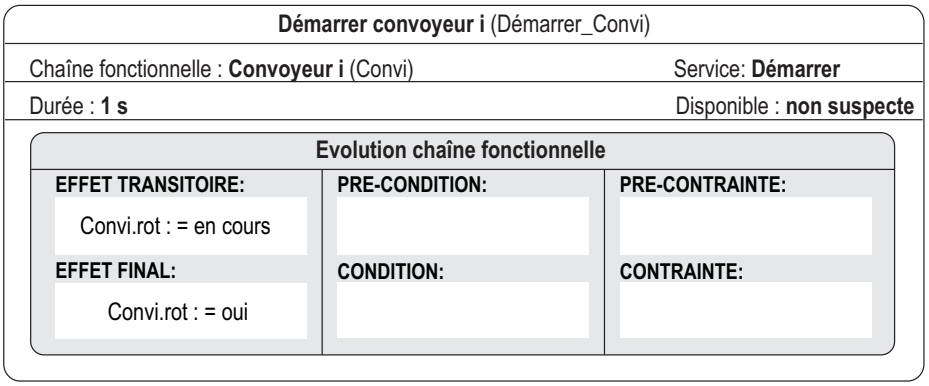


FIG. C.9 – Opération d'action *Démarrer Convoyeur i*

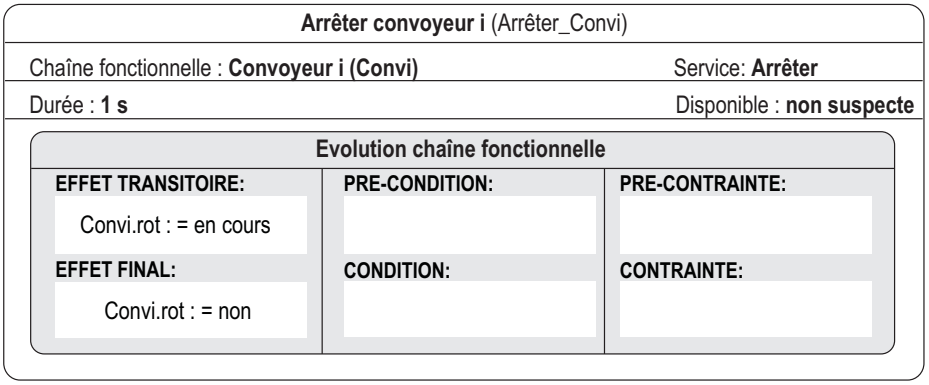


FIG. C.10 – Opération d'action *Arrêter Convoyeur i*

Transférer carte de la position A à la position B (Transf_A-B)

Durée : **5 s**

1ere Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
CarteEnA → CarteEnA-B	CarteEnA = = non vide ∧ B1.position = = ouverte ∧ Conv1.rot = = oui	CarteEnA-B = = vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnA-B → CarteEnB	Conv1.rot = = oui	CarteEnB = = vide ∧ B1.position = = ouverte

2ème Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
	CarteEnA-B = = non vide ∧ Conv1.rot = = oui	CarteEnB = = vide ∧ B1.position = = ouverte
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnA-B → CarteEnB	Conv1.rot = = oui	CarteEnB = = vide ∧ B1.position = = ouverte

FIG. C.11 – Opération induite *Transférer carte de la position A à la position B*

Transférer carte de la position B à la position C (Transf_B-C)

Durée : **5 s**

1ere Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
CarteEnB → CarteEnB-C	CarteEnB = = non vide ∧ B2.position = = ouverte ∧ Conv1.rot = = oui	CarteEnB-C = = vide ∧ Vt.position = = rentrée
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnB-C → CarteEnC	Conv1.rot = = oui	CarteEnC = = vide ∧ B2.position = = ouverte ∧ Vt.position = = rentrée

2ème Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
	CarteEnB-C = = non vide ∧ Conv1.rot = = oui	CarteEnC = = vide ∧ B2.position = = ouverte ∧ Vt.position = = rentrée
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnB-C → CarteEnC	Conv1.rot = = oui	CarteEnC = = vide ∧ B2.position = = ouverte ∧ Vt.position = = rentrée

FIG. C.12 – Opération induite *Transférer carte de la position B à la position C*

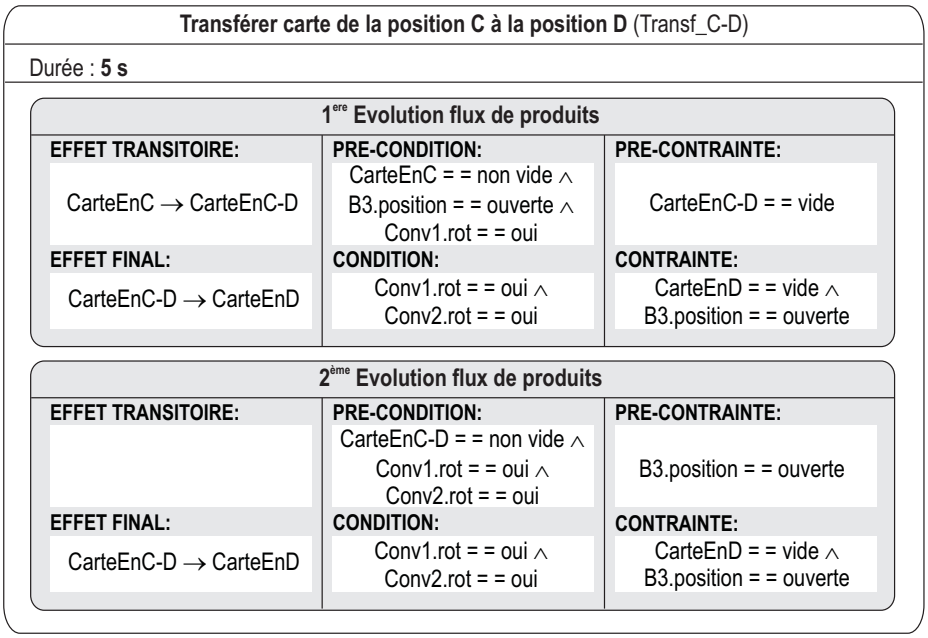


FIG. C.13 – Opération induite *Transférer carte de la position C à la position D*

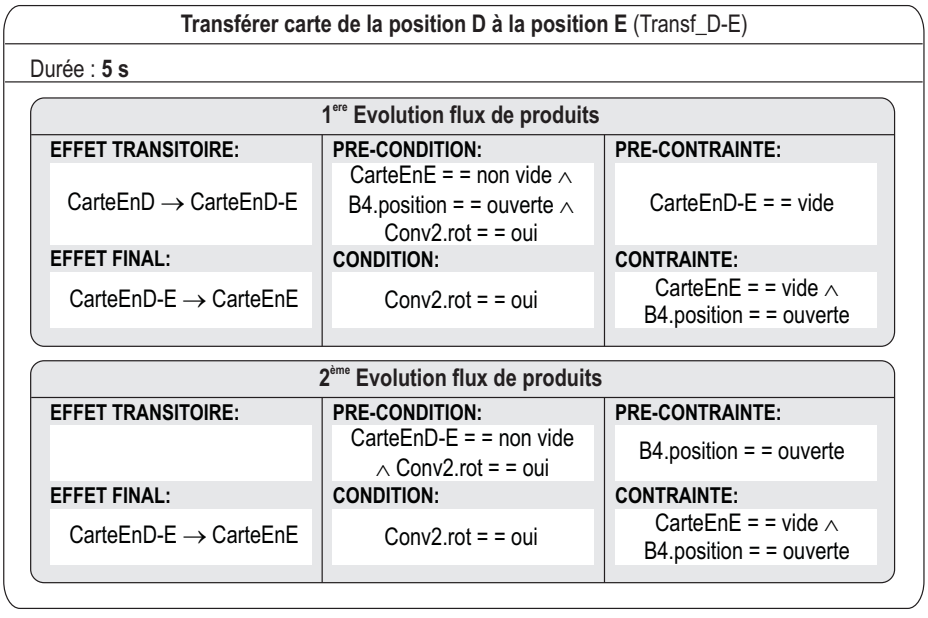


FIG. C.14 – Opération induite *Transférer carte de la position D à la position E*

Transférer carte de la position F à la position G (Transf_F-G)

Durée : **5 s**

1ere Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
CarteEnF → CarteEnF-G	CarteEnF = = non vide ∧ B5.position = = ouverte ∧ Conv3.rot = = oui	CarteEnF-G = = vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnF-G → CarteEnG	Conv3.rot = = oui	CarteEnG = = vide ∧ B5.position = = ouverte

2ème Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
	CarteEnF-G = = non vide ∧ Conv3.rot = = oui	B5.position = = ouverte
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnF-G → CarteEnG	Conv3.rot = = oui	CarteEnG = = vide ∧ B5.position = = ouverte

FIG. C.15 – Opération induite *Transférer carte de la position F à la position G*

Transférer carte de la position G à la position H (Transf_G-H)

Durée : **5 s**

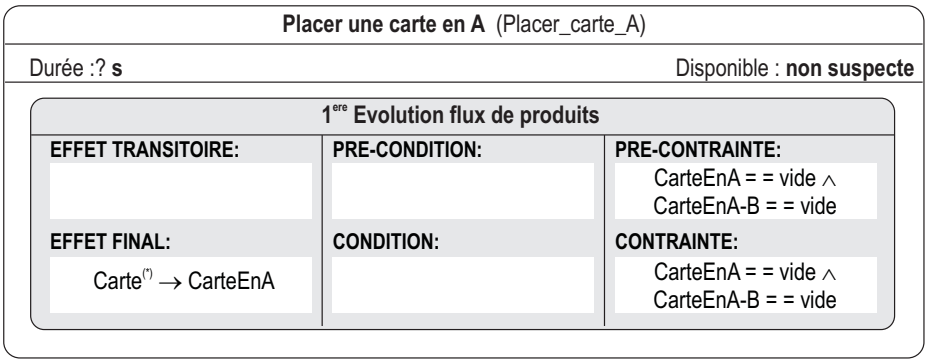
1ere Evolution flux de produits

EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
CarteEnG → CarteEnG-H	CarteEnG = = non vide ∧ B6.position = = ouverte ∧ Conv3.rot = = oui	CarteEnG-H = = vide
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnG-H → CarteEnH	Conv3.rot = = oui	CarteEnH = = vide ∧ B6.position = = ouverte

2ème Evolution flux de produits

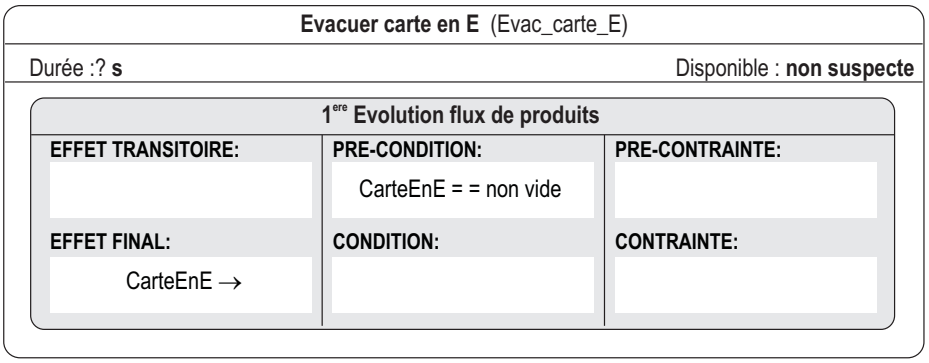
EFFET TRANSITOIRE:	PRE-CONDITION:	PRE-CONTRAINTE:
	CarteEnG-H = = non vide ∧ Conv3.rot = = oui	B6.position = = ouverte
EFFET FINAL:	**CONDITION:**	**CONTRAINTE:**
CarteEnG-H → CarteEnH	Conv3.rot = = oui	CarteEnH = = vide ∧ B6.position = = ouverte

FIG. C.16 – Opération induite *Transférer carte de la position G à la position H*



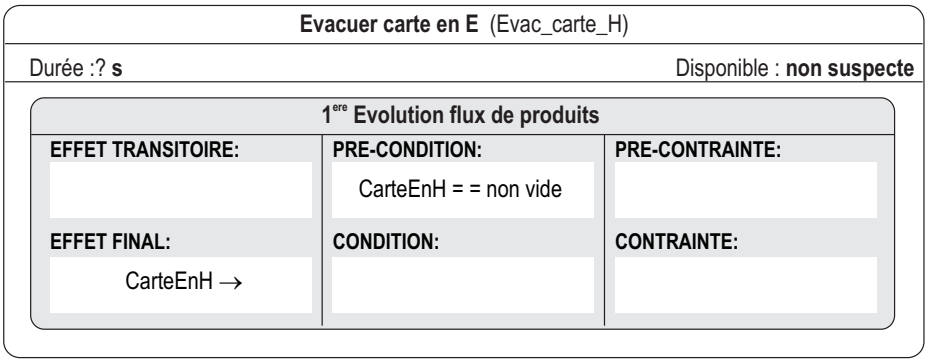
(*) Carte.nett : = non réalisé
Carte.vernis : = non réalisé

FIG. C.17 – Opération requise *Placer une carte en A*



(*) Carte.nett : = non réalisé
Carte.vernis : = non réalisé

FIG. C.18 – Opération requise *Évacuer carte en E*



(*) Carte.nett : = non réalisé
Carte.vernis : = non réalisé

FIG. C.19 – Opération requise *Évacuer carte en H*

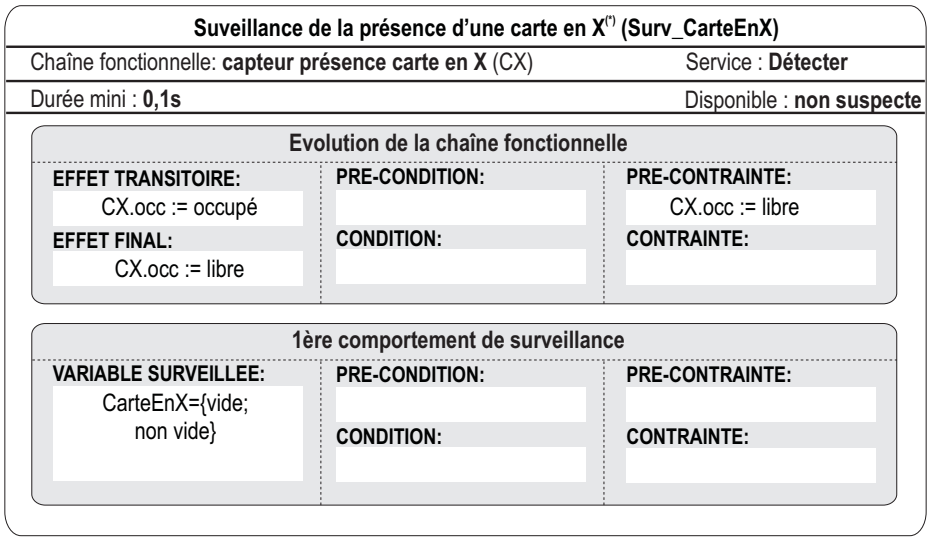


FIG. C.20 – Opération de *surveillance de la présence d'une carte à une position*

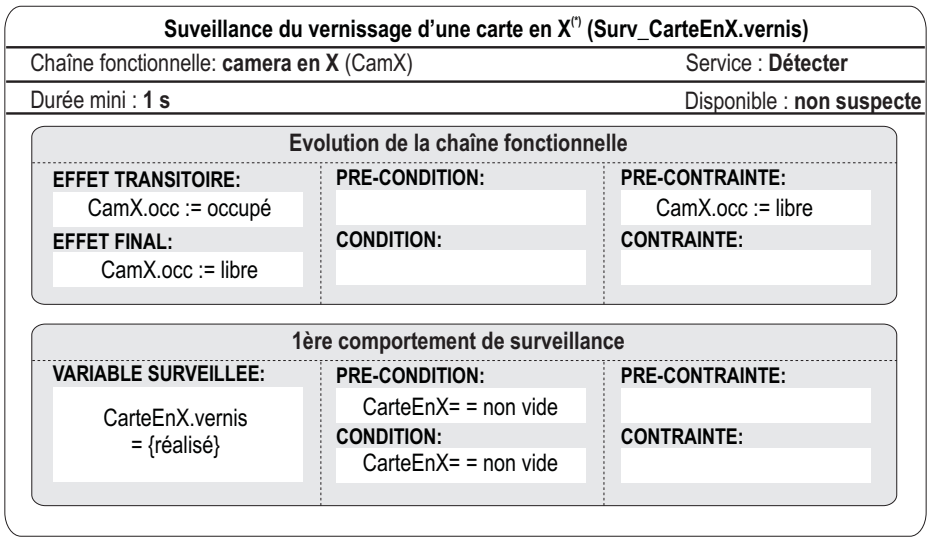


FIG. C.21 – Opération de *surveillance du vernissage d'une carte*

Résumé : Cette thèse s'inscrit dans le domaine de la reconfiguration dynamique des systèmes de production. Elle apporte sa contribution au diagnostic de services en présence de défaillances de la partie opérative. L'objectif visé est de mettre à jour, en ligne, un modèle représentant les capacités offertes par une partie opérative. Aussi, sur la base d'un tel modèle reprenant les principes issus de la planification automatique, nous avons tout d'abord proposé un mécanisme de suivi permettant de construire et de gérer un modèle d'historique. Ensuite, un algorithme de recherche avant/arrière basé sur un système de règles a été développé afin, sous l'occurrence de défaillances, de localiser non seulement les services étant à l'origine possible du dysfonctionnement constaté mais aussi ceux potentiellement affectés. Après quoi, une mise à jour du modèle de partie opérative est opérée afin de présenter au système de reconfiguration une image "honnête" des capacités opératoires encore disponibles.

Mots clès : Diagnostic à base de modèle, réactivité aux aléas de fonctionnement, reconfiguration, systèmes à événements discrets, systèmes manufacturiers.

Abstract: This Phd thesis takes place in the context of dynamic reconfiguration of production systems. It contributes to the diagnosis of services in the presence of failures in the operative part. The aim is to update on line the model, which uses the principles issued from automatic planning, representing the capacities offered by the operative part. Firstly, a tracking mechanism allowing the construction and the management of an history is proposed. Secondly, an algorithm of forward/backward search based on a system of rules has been developed to locate not only the services being at the possible origin of the observed dysfunction but also services potentially affected. Finally, the operating part model is updated in order to provide the reconfiguration system with an "honest" model of the operational capacities which are still available.

Keywords: Model-based diagnosis, reactivity to falures, reconfiguration, discrete-event systems, manufacturing systems.