Méthode de recherche des scénarios redoutés pour l'évaluation de la sûreté de fonctionnement des systèmes mécatroniques du monde automobile

Sarhane Khalfaoui

N° d'ordre : Année 2003

THÈSE

Présentée au

Laboratoire d'Analyse et d'Architecture des Systèmes
du CNRS

en vue de l'obtention du titre de

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE TOULOUSE

Ecole doctorale : Systèmes
Spécialité : Systèmes Automatiques

par

M. ***Sarhane KHALFAOUI***

Soutenue le 26 septembre 2003

MÉTHODE DE RECHERCHE DES SCÉNARIOS REDOUTÉS POUR L'ÉVALUATION DE LA SÛRETÉ DE FONCTIONNEMENT DES SYSTÈMES MÉCATRONIQUES DU MONDE AUTOMOBILE

MEMBRES DU JURY :

MM.	Etienne Craye	Rapporteur
	Yves Dutuit	Rapporteur
	Hamid Demmou	Directeur de thèse
	Robert Valette	Directeur de thèse
	Dominique Charny	Examinateur
	Gérard Fontan	Examinateur

A la mémoire de ma grand-mère,

Avant-Propos

Le travail présenté dans ce mémoire a été effectué dans le cadre d'une convention CIFRE entre PSA Peugeot Citroën et le Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS) du CNRS.

Ce travail a été dirigé par Hamid Demmou et Robert Valette à qui j'exprime toute ma gratitude d'avoir su instaurer entre nous un climat d'amitié, ce qui a rendu mes séjours au LAAS très agréables, très fructueux et plus nombreux. Merci Hamid de m'avoir écouté et conseillé dans les moments difficiles. Merci Robert pour les nombreux instants que tu as consacrés à m'écouter, à deviner mes idées parfois désordonnées, et pour ta recette magique quant au syndrome de la page blanche. Que de bons moments passées à lire ces quelques chapitres au LAAS entre deux sollicitations ou à la montagne en guise de livre de chevet. Je souhaite de tout mon cœur à tout doctorant d'avoir des directeurs aussi humains, compétents et complémentaires que vous deux.

Au sein de PSA, mon encadrement a été assuré par Edwige Guilhem puis par Thierry Cambois du service Electronique Informatique Embarqués Véhicule (EIEV) de la Direction des Systèmes d'Informations (DSIN). Merci Edwige de m'avoir écouté, de m'avoir fait confiance et surtout merci pour tes remarques très pertinentes par rapport à des travaux théoriques très éloignés de ton domaine. Thierry, même si tu n'a pris le relais que vers la fin de la thèse, tu as su comprendre et saisir l'essentiel de mes travaux très vite. Merci de m'avoir assuré un climat de sérénité pendant la phase de rédaction.

Je tiens à remercier Messieurs Etienne Craye et Yves Dutuit, Professeurs des universités respectivement à l'Ecole Centrale de Lille et à l'Université de Bordeaux I, pour avoir accepté d'étudier mes travaux avec beaucoup d'intérêt et d'en être les rapporteurs.

Je tiens également à exprimer ma reconnaissance à Monsieur Gérard Fontan, Professeur à l'Institut National Polytechnique de Toulouse, de m'avoir fait l'honneur de présider mon jury de soutenance et d'avoir suivi, depuis le début, l'avancement de mes travaux avec beaucoup d'intérêt.

J'exprime ma profonde gratitude à Monsieur Dominique Charny, alors chef du service Electronique Informatique Embarqués Véhicule (EIEV), de m'avoir encouragé, de m'avoir fait confiance et d'avoir accepté de faire partie de mon jury de soutenance.

Enfin, je tiens à remercier tous mes collègues et amis de l'équipe EIEV de la bonne ambiance et pour les nombreuses discussions intéressantes que nous avons eues dans la salle café. Que mes amis du LAAS soient remerciés pour leur soutien et leur sympathie.

Que ceux qui ont participé de près ou de loin à la réalisation de ce travail trouvent ici le témoignage de ma reconnaissance.

Je dédie enfin ces quelques lignes à tous ceux qui ont cru en moi et qui m'ont aidé et soutenu pendant ces trois années. Je pense à Sandrine qui a été à mes côtés et qui a su me comprendre et me soutenir dans les moments difficiles. Je pense à mes parents qui ont tout donné pour que je puisse réaliser mes rêves. Je pense à ma sœur, à mes frères, à mes amis et à toute ma famille.

Table des matières

Introduction générale ***9***

Objectif de la thèse et contribution 10

Plan du manuscrit 10

Chapitre 1. Sûreté de Fonctionnement des systèmes mécatroniques : concepts, méthodes et limites ***13***

I Introduction 13

II La Sûreté de Fonctionnement 13

II.A Quelques notions 13

II.B Quelques méthodes 15

III Les systèmes mécatroniques 16

III.A Définition 16

III.B Exemples de systèmes mécatroniques 17

III.C L'aspect dynamique hybride des systèmes mécatroniques 19

IV La Sûreté de Fonctionnement des systèmes mécatroniques 20

IV.A Limites des méthodes classiques 20

IV.B La fiabilité des systèmes dynamiques hybrides 21

IV.C Les méthodes de modélisation 23

IV.D Les méthodes d'analyse 27

V Un tour d'horizon 30

V.A Travaux de J. L. Chabot 31

V.B Méthode des graphes de flux dynamiques 32

V.C Travaux de G. Moncelet 38

VI Synthèse 43

Chapitre 2. Modélisation hybride et logique ***45***

I Introduction 45

II Aspect hybride 45

II.A Approches de modélisation de l'aspect hybride avec les RdP 45

II.B Les RdP associés à des équations différentielles 48

II.C Les réseaux de Petri Predicats-Transitions Différentiels et Stochastiques (RdP PTDS) 52

II.D Discussion sur le modèle hybride 54

III Aspect logique et accessibilité 55

III.A Introduction 55

III.B Logique Linéaire 55

III.C Traduction des réseaux de Petri en logique Linéaire 58

III.D Construction de l'arbre de preuve canonique 61

III.E Graphe de précédence 65

III.F Séquent caractéristique d'un seul ordre partiel 68

III.G Exemple avec conflit 69

III.H Apport de la logique Linéaire 70

Chapitre 3. Extraction des scénarios redoutés à partir d'un modèle RdP ***73***

I Introduction 73

II Scénarios redoutés 73

II.A Définition 73

II.B Formalisation en RdP et en logique Linéaire 74

III Accessibilité entre deux marquages : deux approches duales 74

III.A Accessibilité avant 75

III.B Accessibilité arrière 75

IV Raisonnement dans un contexte inconnu 78

IV.A Raisonnement avant 78

IV.B Raisonnement arrière 81

V Méthode de recherche de scénarios redoutés 82

V.A Principe 83

V.B Les différentes étapes 86

V.C Exemple d'application de la méthode 87
V.D La recherche des scénarios est un processus itératif 88
VI Algorithme pour la recherche de scénarios 88
VI.A Enrichissement du marquage 88
VI.B Structures de données 90
VI.C Quelques procédures 92
VI.D Algorithme 93
VI.E Application sur un cas d'étude 95
VII Conclusion 100

Chapitre 4 : Application 101
I Introduction 101
II Le conjoncteur disjoncteur électromécanique 101
II.A Présentation du système 101
II.B Modélisation du conjoncteur-disjoncteur 102
II.C Application de la méthode de recherche de scénarios 105
III Le système de régulation des réservoirs 109
III.A Présentation 109
III.B Modélisation 110
III.C Application de la méthode de recherche de scénarios 112
IV Conclusion 116

Conclusion générale 119

Bibliographie 123

Annexe A1. Conflit de jetons et de transitions : duplication de l'arbre de preuve 133

Annexe A2. Organigramme ½ 135

Annexe A2. Organigramme 2/2 136

Introduction générale

L'intégration de l'électronique dans les systèmes automobiles a apporté des améliorations notables et indéniables. Ces améliorations concernent l'augmentation du confort et des performances (comportement routier et freinage), la diminution de la pollution et de la consommation ainsi que l'amélioration de la sécurité des véhicules automobiles. Ces innovations n'auraient sûrement pas existé sans l'aide de l'électronique. Il est reconnu que désormais 90% des nouveautés apportées dans les nouveaux véhicules ne pourraient l'être sans l'électronique.

Son utilisation a permis d'implémenter des lois de pilotage avancées et sophistiquées pour la commande des systèmes mécaniques ou hydrauliques. L'association de ces derniers avec le système de pilotage a donné naissance aux systèmes mécatroniques. Ces derniers cumulent les avantages de chacune des technologies mécanique, hydraulique et électronique pour réaliser une fonction donnée en gérant au mieux les ressources d'un véhicule. Afin d'améliorer les performances en freinage du véhicule, le système ABS (exemple de systèmes mécatroniques) intervient sur le système de freinage classique pour éviter le blocage des roues en analysant en permanence, au travers d'un calculateur, la vitesse du véhicule, sa variation et celles des quatre roues.

La facilité d'implémentation des lois de contrôle commande est un atout majeur des systèmes mécatroniques. Quand nous avons besoin d'une nouvelle fonction et quand nous disposons des ressources nécessaires, il suffit de la coder. Par exemple, disposant des capteurs de vitesse du système ABS, le système ASR (anti-patinage) permet d'éviter aux roues de patiner sous l'effet d'un couple excessif après l'ajout d'un simple programme informatique. Cette flexibilité logicielle a certainement amélioré les performances et la sécurité des véhicules mais a considérablement augmenté la complexité des systèmes mécatroniques, ce qui rend la maîtrise de leur fiabilité très difficile.

Afin de garantir un niveau de fiabilité et de sécurité convenable, des études de sûreté de fonctionnement doivent être menées tout au long du cycle de développement d'un véhicule : de la spécification jusqu'à l'intégration. Traditionnellement, la satisfaction des exigences de sûreté de fonctionnement est validée après des tests sur des prototypes réels. C'est à ce stade que nous détectons les erreurs de conception et/ou la non satisfaction des exigences. Afin de diminuer le nombre de prototypes nécessaires à la validation des systèmes conçus, les études de sûreté de fonctionnement doivent être menées au plus tôt dans le cycle de développement avant que les choix de conception ne soient figés. Au plus tôt nous détectons une erreur de conception, au plus tôt elle sera corrigée et moins coûteuse sera cette correction. Par conséquent, ces études doivent être effectuées à partir de l'analyse des modèles des systèmes en cours de conception.

La flexibilité logicielle des systèmes mécatroniques a été également très utilisée afin d'implémenter des stratégies de reconfiguration permettant d'accroître leur sûreté. Ces reconfigurations consistent à assurer un mode de fonctionnement dégradé de la fonction réalisée en présence de défaillances. Le nombre de ces reconfigurations a rapidement augmenté dans l'objectif d'assurer un bon niveau de sécurité. Toutefois, cela a contribué à

accroître fortement la complexité des systèmes mécatroniques et par conséquent à en diminuer la maîtrise.

Cette complexité provient du nombre important de fonctions à réaliser ainsi que de la forte interaction entre ces fonctions. Ceci rend inefficaces les approches classiques de sûreté de fonctionnement basées sur des études séparées des sous-systèmes relativement indépendants. Outre ces interactions entre fonctions, les systèmes mécatroniques, en tant que systèmes dynamiques hybrides, sont le siège d'une forte interaction entre les grandeurs énergétiques et les paramètres de commande et de reconfiguration. Toutes ces interactions font que certains comportements inattendus peuvent surgir et altérer la sécurité du système en donnant naissance à des scénarios redoutés. Il ne suffit donc plus de composer les scénarios redoutés des différents sous-systèmes pour avoir l'ensemble des scénarios redoutés du système global mais il faut chercher ces scénarios en se basant sur une modélisation du système complet incluant les interactions entre les sous-systèmes.

Pour évaluer quantitativement la sécurité des systèmes mécatroniques pendant la phase de conception, il faut tout d'abord connaître les scénarios redoutés. Pour cela, nous sommes amenés à faire une analyse qualitative à partir d'un modèle qui capture les caractéristiques essentielles des systèmes mécatroniques du point de vue de la sûreté de fonctionnement.

Objectif de la thèse et contribution

Les travaux développés au cours de cette thèse porte sur l'aide à la conception de systèmes mécatroniques sûrs de fonctionnement. Nous avons focalisé nos efforts sur l'analyse qualitative de ces systèmes en vue de l'obtention des scénarios redoutés. La connaissance de ces scénarios permet de les évaluer et de valider les lois de reconfiguration pour orienter le choix des concepteurs quant aux différents types d'architectures possibles proposées pour le système à concevoir.

Nous avons développé une méthode de recherche des scénarios basée sur la modélisation préalable d'un système mécatronique sous la forme d'un Réseau de Petri et d'un ensemble d'équations différentielles. Cette modélisation hybride présente l'avantage de séparer clairement les aspects discrets et continus. Ceci nous permet une analyse logique (fondée sur la logique Linéaire) des causalités résultant des changements d'états. Grâce à cette analyse, il est possible à partir d'un état redouté de remonter les chaînes de causalité et de mettre ainsi en évidence tous les scénarios possibles conduisant à une situation critique. Chaque scénario est donné sous la forme d'un ordre partiel entre les événements nécessaires à l'apparition de l'état redouté. L'originalité de notre approche est qu'elle n'implique pas une énumération brutale et globale de tous les états accessibles du système. Au contraire, elle permet de se focaliser sur le voisinage de l'état redouté en faisant une énumération locale d'états partiels. Autrement dit, nous ne considérons que les états des composants directement impliqués dans l'apparition de l'état redouté.

Plan du manuscrit

Ce manuscrit est composé de quatre chapitres. Nous introduirons dans le premier chapitre quelques notions relatives à la sûreté de fonctionnement, les caractéristiques des systèmes mécatroniques et les limites des méthodes classiques de sûreté de fonctionnement. Nous y présenterons également les principales méthodes de modélisation et d'analyse des systèmes mécatroniques avant de terminer par une présentation des principaux travaux se rapprochant

de notre approche. A l'issue de ce chapitre, nous choisirons le modèle « états-transitions » adéquat pour notre problématique : les réseaux de Petri.

Le deuxième chapitre détaillera la modélisation des systèmes mécatroniques avec le formalisme des réseaux de Petri, sous deux angles : une modélisation hybride et une modélisation logique. La modélisation hybride respecte l'interaction et la bonne séparation entre les aspects discrets et continus des systèmes mécatroniques. La modélisation logique, quant à elle, est basée sur la logique Linéaire et permet d'exprimer les relations de cause à effet, dites également relations de causalité, entre les événements menant à des états redoutés. A la fin de la première partie sur la modélisation hybride, nous introduirons un nouveau formalisme adapté à la modélisation des systèmes mécatroniques dans le cadre des études de sûreté de fonctionnement : le formalisme des réseaux de Petri Prédicats Transitions Différentiels Stochastiques (RdP PTDS). La recherche des relations de causalité opère sur le modèle réseau de Petri ordinaire sous-jacent à ce nouveau formalisme.

Au cours du troisième chapitre, nous présenterons la méthode de recherche de scénarios redoutés que nous avons développée. Elle est basée sur la recherche de cause à effet à partir du modèle réseau de Petri du système et permet de caractériser les scénarios redoutés sous la forme d'un ensemble de relations d'ordre entre les événements menant d'un état de bon fonctionnement à un état redouté. Nous présenterons également un algorithme permettant de rendre automatique l'application de cette méthode.

Le dernier chapitre portera sur l'application de la méthode à deux exemples de systèmes mécatroniques : le conjoncteur disjoncteur électromécanique et un système de régulation de deux réservoirs dont le fonctionnement est inspiré de celui de la suspension hydraulique.

Chapitre 1. Sûreté de Fonctionnement des systèmes mécatroniques : concepts, méthodes et limites

I Introduction

Dans ce chapitre, nous présenterons un état de l'art sur les méthodes de sûreté de fonctionnement adaptées à l'étude des systèmes industriels complexes. Nous rappellerons dans un premier temps quelques concepts de la sûreté de fonctionnement. Nous présenterons ensuite les systèmes mécatroniques en tant que classe à part entière des systèmes dynamiques hybrides avant de placer nos travaux dans le cadre de la fiabilité dynamique. Nous exposerons, à ce stade les limites des méthodes classiques de sûreté de fonctionnement, les méthodes de modélisation et d'analyse développées pour l'étude de risque ou de sécurité des systèmes mécatroniques. Nous terminerons par un tour d'horizon des principaux travaux dans ce domaine qui se rapprochent de notre problématique.

II La Sûreté de Fonctionnement

Ce paragraphe présente une synthèse des définitions données dans [Laprie 96] et [Villemeur 88], références auxquelles le lecteur peut se rapporter pour des aspects plus détaillés ou complémentaires.

II.A Quelques notions

La **Sûreté de Fonctionnement** (notée SdF) peut être définie, au sens large, comme la science des défaillances [Villemeur 88]. Elle inclut leur connaissance, leur évaluation, leur prévision, leur mesure et leur maîtrise. Elle représente l'aptitude d'une entité à satisfaire à une ou plusieurs fonctions requises dans des conditions données.

Selon [Laprie 96], la SdF est la propriété d'un système permettant à ses utilisateurs de placer une confiance justifiée dans le service délivré.

Au sens strict, la Sûreté de Fonctionnement est l'aptitude d'une entité à satisfaire une ou plusieurs fonctions requises dans des conditions données. Elle peut être caractérisée par les attributs suivants :

- La ***fiabilité*** : c'est l'aptitude d'une entité à accomplir une fonction requise, dans des conditions données, pendant une durée donnée. Elle est généralement mesurée

par la probabilité qu'une entité accomplisse une fonction requise, dans les conditions données, pendant l'intervalle de temps [0, t].

- La ***disponibilité*** : c'est l'aptitude d'une entité à être en état d'accomplir une fonction requise dans des conditions données et à un instant donné. La disponibilité est généralement mesurée par la probabilité qu'une entité soit en état d'accomplir une fonction requise dans des conditions données et à un instant t donné.
- La ***maintenabilité*** : c'est l'aptitude d'une entité à être maintenue ou rétablie dans un état dans lequel elle peut accomplir une fonction requise, lorsque la maintenance est effectuée dans des conditions données avec des procédures et des moyens prescrits. Elle est généralement mesurée par la probabilité que la maintenance d'une entité accomplie dans des conditions données, avec des procédures et des moyens prescrits, soit achevée au temps t, sachant que l'entité est défaillante à l'instant t = 0.
- La ***sécurité*** : c'est l'aptitude d'une entité à éviter de faire apparaître, dans des conditions données, des événements critiques ou catastrophiques. La sécurité est généralement mesurée par la probabilité qu'une entité évite de faire apparaître, dans des conditions données, des événements critiques ou catastrophiques.

La Sûreté de Fonctionnement a pour objectif de spécifier, concevoir, réaliser et exploiter des systèmes où la faute est naturelle, prévue et tolérable [Laprie 96].

Une **défaillance** est la cessation de l'aptitude d'une entité à accomplir une fonction requise. La défaillance d'une entité résulte de causes qui peuvent dépendre des circonstances liées à la conception, la fabrication ou l'emploi et qui ont entraîné la défaillance. Enfin, le mode de défaillance est l'effet par lequel une défaillance est observée (définition de la Commission Electrotechnique Internationale).

Une **reconfiguration** est l'action de modifier la structure d'un système qui a défailli, de telle sorte que les composants non-défaillants permettent de délivrer un service acceptable, bien que dégradé.

Les principales méthodes d'analyse de la Sûreté de Fonctionnement sont les suivantes :

- L'Analyse Fonctionnelle
- L'Analyse Préliminaires des Risques (APR)
- L'Analyse des Modes de défaillances, de leurs Effets et de leurs Criticités (AMDEC)
- L'Arbre de Défaillance (AdD)
- Le Diagramme de Fiabilité (DdF)

Nous détaillerons par la suite les deux méthodes qui concernent notre travail, à savoir l'analyse préliminaire des risques et la méthode des arbres de défaillances. Par ailleurs, il existe des ateliers logiciels destinés à analyser quantitativement et qualitativement l'effet d'une défaillance sur le comportement du système. Ils permettent de générer les modèles utilisés en Sûreté de Fonctionnement comme les AMDEC, AdD, DdF.

II.B Quelques méthodes

II.B.1 L'Analyse Préliminaire des Risques

L'analyse préliminaire des risques a été utilisée pour la première fois aux Etats-Unis au début des années soixante. Depuis, cette approche a conquis nombre de secteurs industriels tels que l'industrie aéronautique, chimique, nucléaire ou automobile.

Cette méthode a pour objectifs :

- d'identifier les dangers d'un système et de définir ses causes,
- d'évaluer la gravité et les conséquences liées aux situations dangereuses et aux accidents potentiels.

A l'issue de cette étude, des actions correctives sont mises en œuvre afin de permettre la maîtrise ou la suppression des situations dangereuses et des accidents potentiels décelés. Il est recommandé de réaliser l'analyse préliminaire des risques dès les premières phases de conception. Cette étude sera ensuite complétée et enrichie à mesure de l'avancement dans le cycle de vie et ce, jusqu'à la fin de vie du système.

L'APR est en général une étude préliminaire nécessitant la réalisation d'études complémentaires de sûreté de fonctionnement telle que la méthode des arbres des défaillances utile à la détermination des causes des événements indésirables décelés lors de l'analyse préliminaire.

II.B.2 La méthode des Arbres de Défaillances

L'analyse par Arbre de Défaillance est une analyse déductive qui permet de représenter graphiquement les combinaisons d'événements élémentaires qui conduisent à la réalisation d'un événement redouté. L'Arbre de Défaillance, dont la racine correspond à l'événement redouté pour lequel on cherche à évaluer la probabilité d'occurrence, est formé de niveaux successifs tels que chaque événement soit généré à partir des événements du niveau inférieur par l'intermédiaire d'opérateurs logiques (ET, OU, …). La décomposition s'arrête au niveau des événements élémentaires, caractérisés par le fait qu'ils sont indépendants entre eux ou que leurs probabilités peuvent être estimées ou qu'on ne désire pas les décomposer en éléments plus simples.

Un Arbre de Défaillance caractérise de façon claire les liens de dépendance, du point de vue du dysfonctionnement, entre les composants d'un système.

L'analyse par Arbre de Défaillance peut être uniquement qualitative, par recherche systématique des combinaisons minimales de défaillances entraînant l'apparition de l'événement redouté (*coupes minimales*), afin d'identifier les chemins les plus critiques, et donc d'identifier les points faibles du système. Elle peut aussi être d'ordre quantitative ; dans ce cas, on assigne à chaque événement de base une probabilité d'occurrence pour effectuer le calcul de celle de l'événement redouté.

L'analyse par Arbre de Défaillance est largement répandue et utilisée dans les études de sûreté de fonctionnement car elle caractérise de façon claire les liens de dépendance, du point de vue du dysfonctionnement, entre les composants d'un système. En dépit de la simplicité d'utilisation de cette technique, elle souffre néanmoins de l'existence d'hypothèses implicites dont la vérification a posteriori est rarement effectuée par les praticiens. Par exemple, il est

supposé que toute modification de l'ordre dans lequel les événements sont considérés n'a pas d'impact sur le scénario redouté (y compris sa probabilité d'occurrence).

Or, comme nous allons le voir par la suite, cette notion d'ordre dans les événements de défaillance joue un rôle primordial dans les systèmes mécatroniques. Avant de développer ce point, nous présenterons tout d'abord ces systèmes et leurs spécificités.

III Les systèmes mécatroniques

III.A Définition

Un **système mécatronique** est un système combinant des technologies qui relèvent des domaines de la mécanique, de l'hydraulique, de la thermique, de l'électronique et des technologies de l'information [Moncelet 98]. Il peut être décomposé en quatre entités en interaction (voir figure 1.1) : les capteurs, la partie opérative, le système de commande et de reconfiguration et les actionneurs.

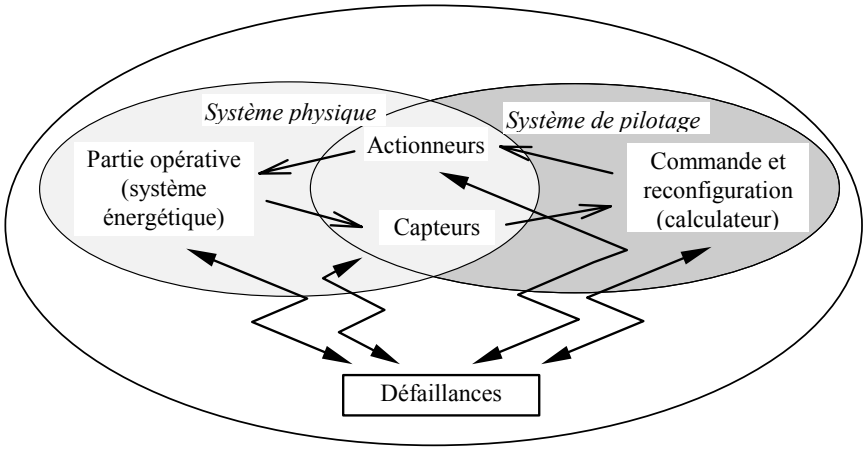


Figure 1.1. **Eléments constitutifs d'un système mécatronique**

Les capteurs mesurent des grandeurs physiques continues caractéristiques de la partie opérative. Le système de commande et de reconfiguration établit en fonction de ces mesures les actions à réaliser. Les actionneurs agissent sur la partie opérative.

Le système de commande a aussi pour objectif d'assurer que certaines grandeurs de la partie opérative soient maintenues dans un intervalle de sécurité. Lorsque certains événements relatifs à la sécurité du système se produisent, comme le franchissement d'un seuil de sécurité par une variable caractéristique de la partie opérative, des actions sont mises en œuvre de façon à reconfigurer la partie opérative et à ramener les grandeurs caractéristiques de celle-ci dans les limites permises.

Les systèmes mécatroniques sont, par leur nature même, des systèmes hybrides[1] dont la partie continue est constituée par la mécanique et l'hydraulique, et la partie discrète est représentée en partie par la commande numérique et les défaillances.

Les spécificités des systèmes mécatroniques sont liées au fait que :

[1] Il s'agit de systèmes dont le comportement ne peut pas être décrit de façon purement continue ou purement discrète.

- ils sont reconfigurables et la décision de reconfiguration est prise par le système lui-même,

- la réussite des reconfigurations dépend, dans le cas général, de la dynamique continue de la partie opérative, mais aussi du temps de réaction du système de pilotage.

Nous supposerons que le système de commande et de reconfiguration (calculateur) est exempt de fautes. Nous étudierons uniquement l'effet des défaillances des capteurs, des actionneurs et de la partie opérative.

III.B Exemples de systèmes mécatroniques

III.B.1 Système d'antiblocage des roues (ABS)

Le freinage a connu ces dernières années d'importantes évolutions. La plus spectaculaire concerne sans aucun doute l'antiblocage des roues : l'ABS (*Anti-lock Braking System*), un système qui a fait son apparition en 1952 sur les avions puis, dès 1978, sur les voitures. L'ABS est désormais devenu incontournable sur la quasi-totalité des véhicules produits, au grand bénéfice de la sécurité active[2] [Kassaagi 01].

Le fonctionnement de l'ABS repose sur un calculateur électronique qui analyse en permanence la vitesse du véhicule et sa variation, ainsi que celle des quatre roues (capteurs embarqués). Lorsqu'il détecte un blocage d'une ou plusieurs roues (glissement[3] de 100%), le système réagit en ordonnant au système de freinage de diminuer son action sur la ou les roues désignées. Cette action est rendue possible par l'utilisation d'électrovalves (ou électrovannes du bloc hydraulique). Une pompe électrique remet de la pression dès que la roue a repris sa vitesse (cette opération peut s'effectuer jusqu'à 12 fois par seconde). Grâce à un temps de réaction très court, le système peut maintenir chaque roue à la limite du blocage, permettant ainsi une décélération optimale (glissement autour de 15%) et empêchant le dérapage du véhicule. En général, le déclenchement de l'ABS est accompagné de vibrations dans la pédale de frein dues au relâchement et à la mise en pression successifs dans le circuit de freinage.

Lors d'un freinage d'urgence, ce système évite donc aux roues du véhicule de se bloquer, de sorte que celui ci « roule » et « ne glisse pas ». Il reste ainsi dirigeable par le conducteur, lui permettant de réaliser un évitement par déport latéral tout en bénéficiant d'une décélération maximale. Lors d'un freinage sur une surface sèche ou mouillée, la distance d'arrêt avec un ABS est légèrement inférieure à celle qu'on aurait obtenu avec les freins conventionnels. En revanche, le freinage avec ABS sur une surface meuble (gravier, neige, …) a pour effet de rallonger relativement les distances d'arrêt : les pneus tournent, demeurent sur le dessus de la surface et, par conséquent, « flottent » sur celle-ci (avec un train conventionnel, le pneu s'enfonce dans le sol, créant un effet « chasse-neige » qui accentue le ralentissement).

Afin d'assurer une grande fiabilité pour ce système vue sa criticité, et ce, sans compromettre ses performances et ses temps de réponse, les constructeurs et/ou

[2] La sécurité active s'intéresse à l'ensemble des fonctions permettant d'éviter un accident (aide au freinage d'urgence, anti-blocage de roue, etc.). Elle est complémentaire à la sécurité passive qui intervient lors de l'accident (airbags, déformation de l'habitacle, etc.).

[3] On définit le taux de glissement par la relation suivante : $\boxed{\lambda = \frac{R \cdot \omega - v_x}{v_x}}$, avec ω, v_x et R, respectivement, vitesse de rotation, vitesse longitudinale et rayon de la roue.

équipementiers automobiles l'ont doté de plusieurs stratégies de reconfigurations en cas de défaillances. Nous en citons quelques unes. Comme nous venons de le voir, le système ABS est composé d'un calculateur ABS, de cinq capteurs de vitesses, du système de freinage en parallèle avec quatre électrovannes. Les stratégies de reconfiguration sont nombreuses et très complexes. Parmi les plus simples, celle qui consiste à estimer la vitesse d'une roue lorsque le capteur associé est diagnostiqué défaillant (détection par le calculateur). La vitesse de la roue est donc estimée au travers des valeurs fournies par les autres capteurs. Selon le mode de défaillance du capteur (dérive, absence de signal, …), d'autres modes de reconfiguration sont prévus. Par exemple, en cas de dérive, le capteur est remis à zéro puis réactivé (une à plusieurs fois selon les versions). La concordance de ses valeurs avec celles des trois autres permet de confirmer ou infirmer sa défaillance définitive et donc de poursuivre ou non l'estimation de la vitesse de la roue concernée. Le conducteur est averti à ce moment de la perte du capteur. Si deux capteurs sont défaillants, le voyant STOP est allumé.

Une autre fonction de plus haut niveau assure la cohérence des actions du calculateur (ou de la fonction) ABS. Si ce dernier est détecté défaillant, cette nouvelle fonction, dite ESP, prend le relais.

III.B.2 Système de contrôle de trajectoire (ESP)

Le **système de contrôle de stabilité** (ou de trajectoire) ESP (*Electronic Stability Program*) est une application relativement récente de l'ABS. Ce système aide le véhicule à maintenir la trajectoire voulue par le conducteur, notamment suite aux erreurs de conduite de celui-ci : il détecte une tendance au dérapage (ou dérive excessive) et corrige en agissant sur une ou plusieurs roues par l'intermédiaire des freins et/ou du moteur (couple) afin de remettre le véhicule sur sa trajectoire. Cette situation est généralement rencontrée suite à une perte d'adhérence du véhicule, le plus souvent dans les virages ou en cas de manœuvre brutale (comme un coup de volant excessif suite à une sortie de chaussée ou une tentative d'évitement d'obstacle). De manière générale, en plus des vitesses des roues (mesurées par le biais des capteurs ABS), le calculateur d'ESP contrôle aussi l'accélération transversale, la vitesse de lacet et l'angle volant du véhicule [Dietsche 01].

Dans le cas d'un sous-virage, le véhicule a tendance à « tirer tout droit » : les roues avant perdent de l'adhérence et le véhicule commence à se diriger vers l'extérieur du virage. Dans un virage à droite (respectivement à gauche), l'ESP freine la roue arrière droite (respectivement gauche) pour créer un couple de rotation qui le fera pivoter légèrement pour reprendre sa trajectoire. En revanche, dans le cas d'un sur-virage, le véhicule a tendance à « glisser du train arrière » : les roues arrières perdent de l'adhérence et le véhicule commence à se diriger vers l'intérieur du virage. Dans un virage à droite (respectivement à gauche), l'ESP freine la roue avant gauche (respectivement droite) pour créer un couple de rotation qui le fera pivoter légèrement pour reprendre sa trajectoire.

De plus en plus de systèmes (ABS, EBD, AFU, Antipatinage, …) sont aujourd'hui intégrés à l'ESP avec éventuellement des capteurs supplémentaires (direction et accélération) pour corriger les amorces de dérapage du véhicule. L'ESP est l'un des nombreux systèmes embarqués qui illustre la forte coopération et interaction entre les sous-systèmes d'une automobile afin de réaliser une ou plusieurs fonctions. Comme nous venons de le voir, le calculateur ESP contrôle le système ABS, le système de freinage et le couple moteur afin de maintenir l'adhérence du véhicule. Sur les véhicules futurs, il est prévu de les équiper avec des directions électroniques (ou « steering-by-wire ») affranchies de la colonne mécanique. Ceci offrirait à l'ESP une possibilité d'agir sur la direction des roues afin d'optimiser

l'adhérence au sol, ce qui contribuerait à accroître ses performances mais également sa complexité. En effet, l'interaction croissante entre les composants et les interconnexions des fonctions pourrait être à l'origine de l'apparition de séquences redoutées non prévues du fait de la complexité croissante des systèmes automobiles. Ceci met à mal les méthodes de travail des concepteurs qui sont basées sur des études et des validations séparées des différents sous-systèmes, auparavant indépendants mais fortement interconnectés aujourd'hui. Sur les véhicules les plus évolués, on dénombre jusqu'à cinquante calculateurs coopérant pour réaliser toutes les fonctions du véhicule [Hartley 97].

L'augmentation du nombre de composants et de fonctions assurées par ce système rend plus complexe l'élaboration des stratégies de reconfigurations. Une des plus simple (et qui est partagée avec le système ABS) est celle qui consiste à compenser une défaillance des électrovannes alimentant en pression hydraulique le circuit de freinage. Si une des électrovannes est bloquée en ouverture ou en fermeture, le système envoie des trains d'impulsions de forte intensité pendant un certain temps afin de secouer la bobine de commande de l'électrovanne et de la débloquer. Pendant ce temps, l'ESP continue de remplir sa mission dans un mode relativement dégradé en s'appuyant sur les autres électrovannes pour compenser l'excès ou le manque de pression dans le circuit hydraulique.

La réussite de certaines configurations du système ESP est dépendante des temps de réponse des grandeurs énergétiques telles que la pression. Cette dépendance entre les aspects continus et discrets (la commande et les défaillances) place les systèmes mécatroniques dans le cadre des systèmes dynamiques hybrides que nous allons présenter dans la section qui suit.

III.C L'aspect dynamique hybride des systèmes mécatroniques

Un **système dynamique hybride** est un système dont la description nécessite l'utilisation de variables continues et de variables discrètes, ainsi que la prise en compte d'une dynamique continue (variables continues apparaissant sous forme dérivée) et d'une dynamique discrète (changements d'états dus à l'occurrence d'événements), ce qui rend la modélisation hybride indispensable.

Un ouvrage [Zaytoon 01], très intéressant, recense un grand nombre de travaux sur les systèmes dynamiques hybrides et en donne quelques applications sur des systèmes industriels. Plusieurs modèles dits « hybrides » y sont présentés. Ces modèles peuvent être classés selon deux approches : celle qui intègre, au sein d'un même formalisme, les aspects continus et discrets, dite **approche intégrée** ; et celle qui sépare ces deux aspects en faisant coopérer deux modèles différents. Cette dernière est appelée **l'approche séparée**. L'approche intégrée englobe tous les modèles issus de l'extension de modèles existants. Nous distinguons ceux issus de l'extension de modèles continus comme les Bond Graph à commutations [Buisson 93] et ceux issus de l'extension de modèles à événements discrets comme les réseaux de Petri hybrides [David 89]. Quant à l'approche séparée, elle regroupe les modèles à base d'Automates hybrides, de Statecharts hybrides, de réseaux de Petri Mixtes ou de réseaux de Petri Prédicats-Transitions-Différentielles (RdP PTD). Dans le chapitre suivant, nous reprendrons une présentation détaillée des principales approches de modélisation de l'aspect hybride basées sur les réseaux de Petri, à savoir les RdP hybrides, les RdP Mixtes et les RdP PTD.

Les systèmes dynamiques hybrides sont caractérisés par une interaction entre des processus continus et des processus discrets [Zaytoon 01], ce qui rend la modélisation hybride indispensable. Les systèmes mécatroniques sont un exemple de systèmes dynamiques

hybrides (cf figure 1.2). En effet, la partie opérative est un processus continu puisque c'est un système énergétique**.** Le système de commande et de reconfiguration, quant à lui, est un processus discret. Les défaillances peuvent également être vues comme un processus discret si on ne s'attache qu'à leur date d'apparition.

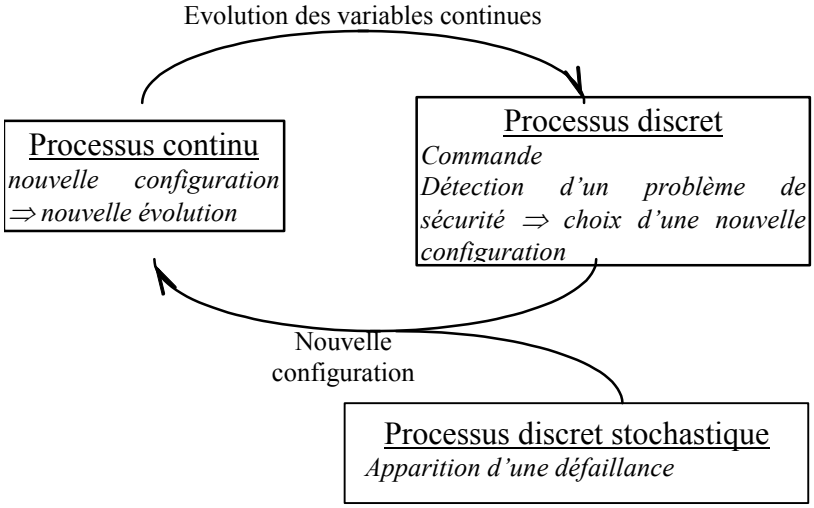


Figure 1.2. **Interaction entre processus continus et discrets dans un système mécatronique**

Ainsi nous intègrons la connaissance des conditions d'apparition des défaillances et la réponse du système à ces défaillances dans notre modèle des systèmes mécatroniques. Ces défaillances agissent sur le processus continu et ce dernier joue un rôle important dans la réussite des reconfigurations et dans l'évaluation de sûreté de fonctionnement de ces systèmes. Dans la section suivante, nous évoquerons les limites des méthodes classiques de sûreté de fonctionnement face à cette nouvelle catégorie de systèmes, et présenterons une panoplie de méthodes de modélisation et d'analyse qui sont mieux adaptées pour les études SdF de ce type de systèmes.

IV La Sûreté de Fonctionnement des systèmes mécatroniques

IV.A Limites des méthodes classiques

Les méthodes classiques de sûreté de fonctionnement, comme celles que nous avons vues en section I, sont basées sur une représentation logique du système étudié. Basées sur la logique booléenne, ces approches sont dites *statiques.* En effet, ces méthodes sont bien adaptées à l'étude des systèmes dits à configuration statique, c'est-à-dire des systèmes dont les relations fonctionnelles entre leurs composants restent figées tout au long de leur mission.

Dans le cadre de nos travaux, la prise en compte des mécanismes de reconfiguration dans les systèmes mécatroniques est essentielle. Les méthodes classiques de sûreté de fonctionnement ne prennent pas en compte cet aspect, comme nous l'expliquerons dans le paragraphe suivant.

Considérons par exemple la méthode des Arbres de Défaillance. Elle ne permet pas de différencier les scénarios comportant des ordres d'apparition d'événements différents. En effet, une séquence d'événements peut conduire à un événement redouté alors que les mêmes événements se produisant dans un ordre différent ou à des dates différentes n'y conduisent

pas. De plus, le temps séparant deux événements n'est pas explicitement pris en compte dans la construction du scénario, nous ne peut donc pas représenter les reconfigurations. Enfin, il n'est pas possible de prendre en compte les défaillances temporaires.

Un exemple simple et connu [Moncelet 98], permettant d'illustrer ces limites, est celui du système à deux composants en redondance passive. Ces deux composants (**a** et **b**) peuvent tomber en panne en service (défaillances notées respectivement def_a et def_b) et subir des réparations (notées respectivement rep_a et rep_b). La commutation d'un composant à un autre est supposée immédiate et fiable. L'événement redouté du système est l'absence de service. Le composant **a** étant le composant principal, les séquences suivantes ont des conséquences différentes : $s_1 = [def_a, def_b, rep_a]$ et $s_2 = [def_a, rep_a, def_b]$. En effet, la première séquence conduit à l'événement redouté puisque les deux composants sont simultanément défaillants pendant un certain temps, tandis que le système continue à délivrer un service pendant et après la séquence s_2.

Cet exemple correspond exactement au cas des systèmes avec reconfiguration. Si une deuxième défaillance arrive avant la fin de la reconfiguration il y a occurrence d'un événement redouté, dans le cas contraire tout se passe bien (comme cela a été prévu).

Plusieurs extensions des méthodes classiques ont été proposées afin d'élargir leur champs d'application pour prendre en compte le problème que nous venons de mentionner. Citons à titre d'exemple les Arbres de Défaillance avec les portes « A avant B ». Ces méthodes restent tout de même fortement combinatoires et incapables de prendre en compte les changements d'états et les reconfigurations dans les scénarios redoutés. Dans ce but, d'autres méthodes ont été introduites, comme les Diagrammes de Séquence d'Evènements (Event Sequence Diagrams, noté ESD) [Labeau 02a], afin de permettre une meilleure représentation visuelle des scénarios dynamiques. Ils apportent une aide à l'identification et à la construction d'événements ordonnés dans le temps. Des phénomènes tels que les compétitions, les conditions, les synchronisations peuvent être représentés grâce à ce formalisme. Bien que les ESD représentent de manière claire les scénarios en compétition, ils ne peuvent pas être générés automatiquement et nécessitent de la part de l'analyste une définition des états et des transitions. Le concepteur doit donc énumérer tous les états de commande et de reconfiguration. Or, dans le cas des systèmes mécatroniques qui sont des systèmes dynamiques hybrides, cela veut dire que le nombre d'états du système est infini si on prend en compte la partie énergétique. Il en est de même pour les méthodes fondées sur les graphes de Markov qui seront détaillés en section IV.C.1.1.

IV.B La fiabilité des systèmes dynamiques hybrides

Cette section a pour but de présenter la problématique de la fiabilité des systèmes mécatroniques vus comme une classe particulière des systèmes dynamiques hybrides. Quelques exemples illustreront les notions importantes relatives à ce type de système, à savoir l'intime dépendance entre les processus de défaillance et de reconfiguration et le processus continu.

La fiabilité des systèmes dynamiques hybrides est une discipline relativement récente (une dizaine d'années) de la sûreté de fonctionnement. Elle est également connue sous le nom de « fiabilité dynamique » ou encore « Probabilistic Dynamics » [Devooght 92a et b] [Labeau 02a].

[Labeau 02a] définit cette discipline comme « la partie de la sûreté de fonctionnement qui étudie de manière intégrée le comportement des systèmes industriels complexes affectés par

une évolution dynamique continue sous-jacente ». Comme pour les systèmes mécatroniques, le fonctionnement de tels systèmes est régi par deux phénomènes : la variation continue et déterministe des paramètres énergétiques**,** mais aussi par les sollicitations et défaillances des composants du système, de nature discrète et/ou stochastique. Ces deux phénomènes sont en interaction et leur interaction influence les paramètres de sûreté de fonctionnement tels que la fiabilité ou la sécurité.

Les méthodes classiques de sûreté de fonctionnement, y compris les méthodes fondées sur les graphes de Markov sont incapables de prendre en compte de manière satisfaisante la dynamique des variables continues correspondant aux paramètres énergétiques [Dufour 02].

Illustrons au travers de quelques exemples les interactions entre les phénomènes continus et les phénomènes discrets et/ou stochastiques et leur influence sur la fiabilité ou la sécurité. Considérons un composant électronique dont le taux de défaillance peut être affecté par sa température. Le fonctionnement de ce composant est caractérisé par trois plages de température. En dessous du seuil $T_1 = 180°$, le taux de défaillance de ce composant est λ_1. Quand sa température dépasse ce seuil mais reste inférieure à $T_2 = 250°$, son taux de défaillance devient égal à λ_2 ($\lambda_2 > \lambda_1$). Au delà du seuil T_2, le composant cesse de fonctionner. On voudrait calculer la fiabilité de ce composant dans un environnement où la température est variable. Il est indispensable de relier cette évolution de la température avec les différents états du composant pour évaluer sa fiabilité. Cet exemple illustre l'influence des paramètres continus sur les paramètres stochastiques des défaillances. Pour souligner l'influence des défaillances sur les phénomènes continus, imaginons que ce composant serve à commander un ventilateur qui a pour mission de ralentir la croissance de la température ambiante. La défaillance de ce composant empêcherait le ventilateur de fonctionner, ce qui contribuerait à accélérer la croissance de la température ambiante.

Considérons l'exemple de la figure 1.3. Celui-ci a été traité dans [Dufour 02] et dans [Plabeau 02a]. Il s'agit d'un système (réservoir), décrit par une variable continue x (la pression), au sein duquel a lieu un transitoire caractérisé par la croissance rapide de x dans une configuration i = 1. Un endommagement de l'installation se produit si x excède la valeur **L**. Si le seuil **l** correspondant à une valeur d'alerte est atteint, un dispositif de protection est sollicité, avec des probabilités p_0 de succès (entrée du système dans l'état i = 2) et de p_1 de fonctionnement partiel ($p_0 + p_1 = 1$), correspondant à une atténuation moins forte du transitoire (i = 3). Supposons en outre que le transitoire puisse devenir plus sévère, avant d'atteindre le seuil **l**, par la défaillance supplémentaire d'un autre composant. Après cette défaillance, qu'on suppose pouvoir se produire uniquement pour des valeurs de x croissantes, le système se trouve dans une configuration i = 4. Dans ce cas, le dispositif de protection peut encore couper la montée de x s'il fonctionne parfaitement (configuration i = 5), mais son mode de fonctionnement partiel (i = 6) n'est plus suffisant pour permettre d'empêcher la panne du système. Cependant, celle-ci se produit plus tard que dans les configurations i = 1 et i = 4, ce délai laissant une possibilité supplémentaire d'intervention sur le système.

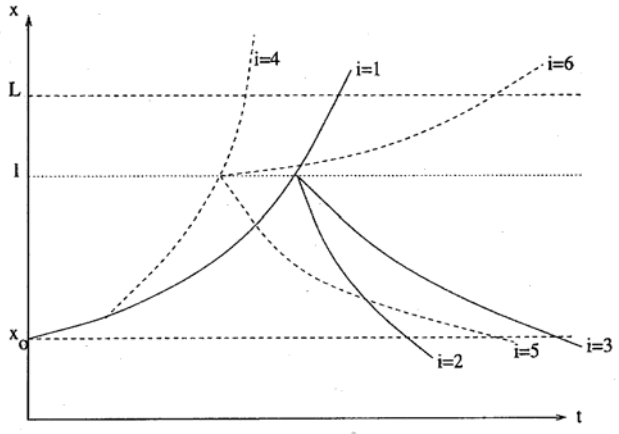


Figure 1.3. **Evolutions transitoires dans un exemple de systèmes dynamiques hybrides**

Cet exemple met en lumière que les lois d'évolution de la variable continue x dépendent de la configuration i, et que certains changements de configuration sont associés à un franchissement de seuil de x. Il y a donc interaction entre les aspects continus et discrets du système. Ceci nous conduit à la conclusion que toute méthode de fiabilité des systèmes dynamiques hybrides doit prendre en compte ces interactions, comme celles présentées dans la section suivante.

IV.C Les méthodes de modélisation

Nous avons soulevé, dans le paragraphe précédent, les limites des méthodes classiques de sûreté de fonctionnement. Nous avons également souligné la nécessité d'utiliser des méthodes plus adaptées à la modélisation et à l'analyse des systèmes dynamiques hybrides tels les modèles états transitions. Ces modèles englobent les graphes d'états (graphes de Markov et automates) et les approches basées sur le formalisme des réseaux de Petri.

Nous présentons ici un ensemble de méthodes de modélisation que nous avons classées selon deux points de vue : les méthodes qui permettent de décrire les aspects dysfonctionnels (les défaillances et les réparations) et le comportement du système en présence de dysfonctionnements, et les méthodes qui permettent la description comportementale des systèmes dynamiques hybrides. La séparation entre ces deux aspects (le comportemental et le dysfonctionnel) est, selon nous, le plus grand handicap qui se dresse contre l'efficacité des méthodes SdF classiques concernant les problèmes relevant de la fiabilité dynamique. En effet, ces deux aspects doivent être intégrés au sein d'un même modèle de fiabilité respectant leur interaction mutuelle.

IV.C.1 Modélisation de l'aspect dysfonctionnel

IV.C.1.1 Graphes de Markov

Cette méthode est destinée à analyser et évaluer la sûreté de fonctionnement des systèmes réparables. La première étape de construction d'un graphe de Markov consiste à identifier les différents états (défaillants ou non défaillants) que le système peut occuper. L'étape suivante consiste à chercher comment le système passe d'un état à un autre. A chaque transition, de l'état E_i vers l'état E_j, est associé un taux de transition L_{ij} défini de telle sorte que $L_{ij}.dt$ est égal à la probabilité de passer de E_i vers E_j entre deux instants très proches t et t + dt sachant que l'on est en E_i à l'instant t.

La modélisation avec les graphes de Markov permet de prendre en compte les dépendances temporelles et fonctionnelles beaucoup plus largement que les méthodes classiques. En effet, l'existence d'une défaillance de mode commun entre deux composants en redondance s'exprimera simplement par l'ajout d'arcs supplémentaires entre états du système correspondant au fonctionnement simultané de ces deux composants et les états où ils sont tous les deux défaillants.

En dépit de leur simplicité conceptuelle et de leur aptitude à pallier certaines carences des méthodes classiques de sûreté de fonctionnement, les graphes de Markov souffrent de l'explosion du nombre d'états, car le processus de modélisation implique l'énumération de tous les états possibles et de toutes les transitions entre ces états. Par exemple, le graphe de Markov associé à un système avec N composants redondants (chacun ayant deux états possibles : marche et panne) peut contenir jusqu'à 2^N états.

Pour éviter ce problème de l'explosion combinatoire du nombre d'états dans la modélisation par graphe de Markov, il est possible sous certaines hypothèses (hypothèse markovienne) de modéliser avec des réseaux de Petri (RdP stochastiques généralisés par exemple) et de générer par la suite le graphe de Markov, ce qui facilite beaucoup la modélisation car elle est plus structurée et plus compacte. Comme l'information associée aux changements d'états est stochastique (taux de transition), cette approche est bien adaptée pour décrire les défaillances. Par contre, elle pose un problème si l'on veut décrire un comportement complexe en dehors de la présence de défaillances (aspect comportemental).

IV.C.1.2 Réseaux de Petri stochastiques

L'utilisation des **réseaux de Petri stochastiques** permet donc de prendre en compte, de manière plus structurée que dans les graphes de Markov, l'occurrence des défaillances et leur influence sur le comportement du système. Ils sont obtenus à partir des réseaux de Petri classiques [Valette 92] en associant des durées de franchissement aléatoires aux transitions [Ziegler 96].

L'hypothèse markovienne est telle que les probabilités de transition entre états dépendent uniquement de l'état courant et non pas de la manière suivant laquelle le système est arrivé dans l'état courant (chemin emprunté dans le graphe des états). Sous ces hypothèses, il est possible de modéliser un processus markovien avec des réseaux de Petri stochastiques mais il ne faut utiliser que des lois de transition exponentielles décroissantes. Une extension des réseaux de Petri stochastiques permet de prendre en compte, en plus des transitions avec des lois exponentielles, d'autres transitions dites « immédiates » tirées sans délai et qui sont prioritaires par rapport aux transitions à délai aléatoire. Ce type de réseaux de Petri est appelé réseau de Petri stochastiques généralisés (RdPSG) [Marsan 84]. A part les RdPSG, d'autres classes ont été proposées. Elles sont classées en fonction du type de la durée de franchissement qu'elles peuvent manipuler. On peut citer les réseaux de Petri Stochastiques Etendus (RdPSE) [Dugan 84] et les réseaux de Petri Stochatiques et Déterministes (RdPSD) [Marsan 86]. La première classe permet de prendre en compte des lois de distribution quelconques et la seconde combine des délais exponentiellement distribués et des délais constants (pouvant être nul).

Une fois le modèle réseau de Petri établi, on génère le graphe des marquages accessibles et on en déduit le graphe de Markov. Deux méthodes de résolution sont alors possibles : soit analytiquement par résolution numérique exacte, soit par simulation de Monte Carlo dans les cas complexes et/ou non Markoviens.

Après avoir présenté les formalismes de modélisation de l'aspect dysfonctionnel, nous présentons ci-dessous ceux utilisés pour la modélisation de l'aspect fonctionnel : décrire le fonctionnement normal du système.

IV.C.2 Modélisation de l'aspect fonctionnel

Comme nous l'avons vu précédemment, les systèmes mécatroniques sont de nature hybride et celle-ci doit être prise en compte pour évaluer leur sûreté.

La modélisation de l'aspect hybride a été abordée selon deux approches : étendre un modèle continu ou étendre un modèle discret [Andreu 96]. La première approche consiste à prendre en compte l'aspect discret par une extension d'un formalisme continu. L'introduction de variables booléennes ou entières dans un système d'équations, celle des « éléments de commutation » dans le formalisme des Bond Graph [Buisson 93] en sont des exemples. Ce type d'approche s'adapte bien aux systèmes intégrant quelques aspects discrets mais qui sont fondamentalement continus. Quand l'aspect discret est conséquent, ce qui est le cas dans les systèmes mécatroniques, cette approche devient inutilisable car très lourde à mettre en œuvre [Khalfaoui 00].

L'autre approche de l'aspect hybride consiste à intégrer l'aspect continu au sein d'un formalisme à événements discrets comme les réseaux de Petri. De nouvelles classes de RdP ont été introduites afin de prendre en compte l'aspect hybride. Nous reviendrons dans le chapitre suivant sur les principaux modèles développés selon cette approche.

IV.C.2.1 Automates

L'un des formalismes états transitions les plus utilisés dans la description des systèmes à événements discrets est celui des automates. Afin de modéliser correctement les systèmes dynamiques hybrides, ce formalisme a été étendu par les automates hybrides. Ils constituent une extension des automates temporisés[4] [Alur 95] dans laquelle les variables continues ne sont plus contraintes à être positives et les évolutions de ces variables ne suivent plus toutes les mêmes lois [Zaytoon 01]. D'un point de vue informel et général, un **automate hybride** est l'association d'un automate d'état fini et un ensemble d'équations dynamiques continues pilotées par ce dernier. Les équations modélisant le comportement continu à un instant donné dépendent de l'état de l'automate**,** ce dernier pouvant évoluer en fonction des valeurs des grandeurs continues.

Toutefois, l'utilisation de ces formalismes est aussi limitée par la taille des modèles qu'ils engendrent (explosion combinatoire du nombre d'états du graphe). Dans le cas de la modélisation de systèmes complexes pouvant être découpés en sous-systèmes ou en sous-modules, il est possible de construire un modèle d'automate pour chacun d'eux et de les composer ensuite pour élaborer l'automate correspondant au système global. Ceci permet de s'affranchir du problème de l'explosion du nombre d'états.

Ces mécanismes de composition se font par synchronisation entre les automates des différents sous-systèmes, soit par messages, soit par variables partagées. Toutefois, cette composition entre automates pose une certaine difficulté quant à la conservation de leurs propriétés. D'où le besoin de disposer de mécanismes de structurations plus puissants offerts par des modèles de plus haut niveau comme les statecharts [Harel 87].

[4] Les automates temporisés sont des automates à états finis étendus par un ensemble d'horloges dont les valeurs croissent uniformément avec le passage du temps et qui peuvent être remises à zéro.

Dans le cadre de mes travaux, nous cherchons à caractériser, à partir d'un modèle du système, les relations de cause à effet susceptibles de mener à un état redouté. Dans le formalisme des automates, ces relations ne sont pas représentées de façon claire et homogène. En effet, au sein d'un automate (élément d'un produit d'automates), les relations de cause à effet sont représentées par les transitions définies par des triplets de la forme (sommet source, événement, sommet destination). Par contre, entre deux automates, ces relations sont représentées par des synchronisations par messages ou par variables. Ceci relève une non unification de la représentation des relations de cause à effet inter et intra automates. Nous verrons dans le paragraphe suivant que ce n'est pas le cas dans le formalisme des réseaux de Petri.

IV.C.2.2 Les réseaux de Petri et l'approche hybride

Les **réseaux de Petri** sont largement utilisés dans la modélisation des systèmes à événements discrets et dans les études de sûreté de fonctionnement des systèmes dynamiques. Ils se caractérisent par une évolution asynchrone dans laquelle les transitions des composantes parallèles sont franchies les unes après les autres, et par une représentation explicite des synchronisations et des mécanismes d'allocation de ressources. Ces caractéristiques sont très intéressantes pour modéliser les aspects événementiels des systèmes hybrides. A partir du modèle d'origine, maintes extensions ont été proposées afin d'étendre leur pouvoir d'abstraction (pour répondre à la modélisation de problèmes spécifiques) et de structuration (pour maîtriser la taille et la lisibilité des modèles). L'un des atouts indéniables des réseaux de Petri, par rapport aux autres formalismes comme les statecharts, est qu'ils reposent sur des fondements théoriques permettant de vérifier les propriétés générales d'un modèle (vérifier que le modèle est réinitialisable, vivant, sans blocage, borné, etc.) ainsi que l'accessibilité de certains marquages. Les méthodes de recherche de propriétés dans les réseaux de Petri sont basées non seulement sur l'élaboration du graphe des marquages accessibles comme c'est le cas pour les automates, mais aussi sur l'algèbre linéaire (calcul des invariants de places et de transitions), la réduction des réseaux ainsi que sur la logique Linéaire. Cette dernière permet de donner une condition nécessaire et suffisante de l'accessibilité entre deux marquages et de caractériser de manière élégante et efficace les relations d'ordre partiel.

Revenons sur les avantages des réseaux de Petri par rapport aux automates. Un réseau de Petri peut être considéré comme une représentation abstraite permettant de générer des automates finis de grande taille, ce qui permet de faciliter la modélisation des systèmes conséquents. Toutefois, la modélisation des systèmes industriels de grande taille avec les réseaux de Petri nécessite tout de même d'intégrer, pendant l'élaboration du modèle, des méthodes de hiérarchisation et de modularité ainsi que des approches à objets pour faciliter l'élaboration et la réutilisabilité du modèle tout en préservant la lisibilité.

Parmi les diverses extensions des réseaux de Petri pour prendre en compte l'aspect hybride, on peut citer les réseaux de Petri de haut niveau, les réseaux de Petri hybrides et les réseaux de Petri couplés avec des équations algébro-différentielles. Nous détaillerons ce point dans le chapitre suivant en présentant les différentes approches de modélisation de l'aspect hybride avec les réseaux de Petri.

Maintenant que nous avons présenté les principales méthodes de modélisation des systèmes mécatroniques, que ce soit pour les études SdF ou pour la description comportementale, nous allons aborder les méthodes d'analyse qualitative et quantitative pour les systèmes mécatroniques.

IV.D Les méthodes d'analyse

Comme nous l'avons vu dans la section III.B, la complexité des systèmes mécatroniques, du fait de la forte interaction entre leurs différents composants, peut faire apparaître des scénarios de défaillance inattendus. Les études de sécurité de ces systèmes sont basées tout d'abord sur une analyse qualitative ayant pour objectif de déterminer ces scénarios. Vient, ensuite, l'analyse quantitative pour estimer leurs probabilités d'occurrence. L'analyse qualitative permet d'améliorer l'efficacité de l'analyse quantitative en l'orientant vers des scénarios précis. En effet, si l'on simule le modèle du système sans aucune connaissance à priori des scénarios pertinents, les défaillances étant très rares, on va passer l'essentiel du temps de simulation à parcourir des comportements correspondant au fonctionnement normal et les informations obtenues sur les reconfigurations et les scénarios critiques seront très pauvres. C'est pourquoi, si l'on part d'un modèle comportemental fonctionnel, les études de sûreté doivent d'abord débuter par une analyse qualitative permettant la mise en évidence des comportements en présence de défaillance pour obtenir, d'une façon ou d'une autre, un modèle dysfonctionnel. Commençons par présenter l'analyse qualitative.

IV.D.1 Analyse qualitative

Cette analyse a pour but de caractériser les scénarios redoutés par des changements d'états et des enchaînements d'événements qui conduisent le système vers un état dit redouté. Selon le formalisme choisi pour la modélisation du système, nous pouvons procéder soit par exploration du graphe des marquages accessibles dans le cas d'un modèle réseaux de Petri, soit par model-cheking s'il s'agit d'automates. Ces deux approches ont un point commun : on génère les trajectoires possibles du système (les exécutions de l'automate ou les séquences de tirs de transitions du réseau de Petri).

IV.D.1.1 Génération du graphe des marquages accessibles

Cette approche commence par la génération du graphe des marquages accessibles à partir d'un modèle en réseau de Petri. Nous déterminons ensuite toutes les séquences permettant l'accessibilité d'un état donné (un marquage partiel) par exploration de ce graphe.

La modélisation des systèmes mécatroniques nécessite la prise en compte de la partie continue. Une façon de faire est de discrétiser les variables continues en un ensemble d'intervalles pertinents du point de vue sûreté de fonctionnement. Parmi les travaux les plus significatifs dans ce cadre, nous pouvons citer ceux de [Moncelet 98]. Nous y reviendrons par la suite de manière plus détaillée.

Cette approche souffre de deux inconvénients majeurs. Tout d'abord, elle est limitée par le problème classique de l'explosion combinatoire du nombre d'états. Cette explosion du nombre d'états est due d'une part à la taille des systèmes traités et d'autre part à l'augmentation du nombre d'états suite à la discrétisation des variables continues. Le second inconvénient est dû au traitement du parallélisme dans le graphe des marquages accessibles par l'entrelacement. Ceci compromet la minimalité des séquences qui mènent vers l'état redouté car elles contiennent des informations qui ne concernent pas uniquement le scénario redouté mais aussi le comportement de sous-systèmes parallèles non impliqués. En effet, ce n'est pas parce qu'un franchissement de transition t_2 suit un franchissement de t_1 dans une séquence que t_1 est une cause nécessaire de t_2. Ce n'est le cas que si un jeton produit par t_1 est consommé par t_2. Cette information n'est pas disponible dans le graphe des marquages, c'est pourquoi le traitement a posteriori des ces séquences pour en extraire l'essentiel (les scénarios minimaux) est difficilement automatisable et l'extraction des ordres partiels n'est pas aisée.

Toutefois, cette approche a l'avantage d'être exhaustive (elle donne toutes les séquences menant vers l'état redouté).

IV.D.1.2 Vérification par model-checking

Le model-checking [Schnoebelen 99] est une technique de vérification qui s'applique à une large classe de systèmes : ceux qui sont modélisables par un automate fini (ou une variante de cette représentation générale). Elle comporte trois étapes : la représentation d'un système par automate, la représentation d'une propriété par une formule logique (en logique temporelle) et finalement l'algorithme de model-checking qui vérifie que l'ensemble des états de l'automate est bien un modèle, au sens logique du terme, de la formule. C'est en 1977 qu'A. Pnuelli a proposé pour la première fois l'utilisation de la logique temporelle pour la spécification des propriétés comportementales des systèmes [Pnuelli 84]. Il existe plusieurs variétés de logique temporelle comme CTL, PLTL, TCTL, …

Une utilisation particulière du model-checking permet de trouver des scénarios menant vers un état redouté [Kehren 03]. Pour cela, il suffit de considérer la propriété de sûreté comme la négation d'une propriété d'atteignabilité : « *on ne peut pas atteindre un état tel que…* ». Dans ce cas, le model-checker (l'outil permettant de faire du model-checking) donne une réponse négative et un contre-exemple qui est une exécution possible permettant d'atteindre l'état redouté. Pour obtenir un autre scénario, il faut formuler une nouvelle propriété qui exclut le premier contre-exemple et recommencer. Or, cette formulation est généralement très compliquée.

En pratique, la taille des systèmes est bien le principal obstacle qui reste à franchir, même en utilisant des structures de données très compactes (les Diagrammes de Décision Binaires). Les utilisateurs de model-checkers sont couramment amenés à simplifier le modèle qu'ils analysent, cela jusqu'à pouvoir le maîtriser. Ce faisant, ils obtiennent toujours plus de garanties au sujet d'un modèle qui s'éloigne toujours plus du système réel. C'est un compromis difficile à gérer. D'autres inconvénients sont à signaler quant à cette utilisation particulière du model-checking. La minimalité des scénarios n'est pas assurée. Les ordres partiels ne peuvent pas être obtenus directement et il est très fastidieux d'assurer l'exhaustivité des scénarios.

En plus, avec ce type de formalisme, il n'est pas possible d'effectuer des évaluations quantitatives de sûreté de fonctionnement. Or, il est important d'utiliser le même formalisme pour les analyses qualitatives puis quantitatives afin d'assurer la cohérence des résultats quantitatifs avec le modèle de départ et de ne pas alourdir la phase de modélisation. La section suivante présente les différentes méthodes d'analyse quantitative de la sûreté de fonctionnement, à savoir les méthodes analytiques, les méthodes fondées sur la discrétisation des variables continues et celles basées sur la simulation de Monte Carlo.

IV.D.2 Analyse quantitative

IV.D.2.1 Méthodes analytiques

Ces méthodes consistent à résoudre les équations de Chapman-Kolmogorov associées au graphe de Markov. Celles-ci donnent l'évolution temporelle de la probabilité d'être dans un état. Sous l'hypothèse que les processus de défaillance et de réparation des composants du système suivent une loi exponentielle à taux constant (l'hypothèse markovienne est vérifiée et nous avons affaire à un processus de Markov homogène), ces équations peuvent être formulées ainsi : $\left[\frac{dP_1(t)}{dt}, \frac{dP_2(t)}{dt}, \cdots, \frac{dP_n(t)}{dt}\right] = \left[P_1(t), P_2(t), \cdots, P_n(t)\right] * \Lambda \quad \Rightarrow \quad P(t) = P_0(t) * e^{\Lambda t}$ où $P_i(t)$

est la probabilité d'être dans l'état i à l'instant t, n est le nombre d'états du graphe et Λ est une matrice carrée de dimension n (matrice des taux de transitions entre états). Il s'agit d'un système linéaire d'équations différentielles du premier ordre dont le traitement analytique est relativement aisé (par transformation de Laplace ou par calcul matriciel) lorsque la taille est raisonnable et que les éléments de Λ ont des ordres de grandeurs comparables.

L'hypothèse markovienne reste restrictive même si elle s'applique assez bien aux composants électroniques dont le taux de défaillance est constant (hors des périodes de défaillances précoces ou d'usures). En effet, le comportement d'un opérateur est fortement non markovien ainsi que le vieillissement d'un composant. Les réparations de durées déterministes sont par exemple très difficilement prises en compte dans le cadre markovien. Sous certaines hypothèses (dépendance du temps passé dans un état ou du temps global), ces approches analytiques peuvent être utilisées dans des cas tests (de petite taille) à condition de mémoriser le temps. Quand l'hypothèse markovienne n'est pas vérifiée, et c'est le cas dans une grande partie des systèmes industriels complexes, les évaluations se font par la simulation de Monte Carlo.

Mais comme nous l'avons vu, le problème majeur concernant les systèmes mécatroniques provient du fait qu'il s'agit de systèmes hybrides. Une approche possible consiste à discrétiser les variables continues.

IV.D.2.2 Méthodes de discrétisation

Plusieurs techniques peuvent être classées dans ce cadre selon que l'on discrétise ou non le temps en plus de la discrétisation des variables continues (Cell-to-Cell Mapping Technique notée CCMT ou Continuous CCMT notée CCCMT) [Moncelet 98]. Nous n'en présenterons que la plus connue et la plus utilisée en fiabilité dynamique : les Arbres Dynamiques Discrets (Discrete Dynamic Event Trees ou DDET).

Cette méthode [Devooght 92] a pour but de générer les scénarios redoutés obtenus par propagation des défaillances des composants élémentaires du système. Elle est basée sur la partition de l'espace des variables continues en cellules disjointes. A partir d'un modèle qui prédit la réponse du système aux défaillances, l'évolution des variables physiques est calculée à chaque pas du temps. A cet instant on calcule de même la probabilité de toutes les combinaisons des états des composants et on génère toutes les séquences d'événements constituant les scénarios possibles, et ainsi de suite jusqu'à ce qu'un état absorbant soit atteint. Une présentation détaillée de cette technique est donnée dans [Moncelet 98] et [Labeau 02a].

Le principal inconvénient de cette méthode est le grand nombre de séquences à traiter, du fait de la discrétisation du temps. Augmenter le pas de discrétisation peut diminuer le nombre de séquences mais diminue également la précision de l'analyse. Il existe d'autres techniques pour diminuer le nombre de séquences comme l'introduction de critères de coupure portant sur le nombre de séquences autorisées ou sur un seuil de probabilité minimal en dessous duquel l'évolution du processus n'est plus prise en compte. Le choix de la valeur à attribuer à ces critères dépend du type du système étudié et influence fortement la répartition dans l'espace des cellules.

Les méthodes comme DYLAM (Dynamic Logical Analytical Methodolgy) et DETAM (Dynamic Event Tree Analysis Method) s'inscrivent dans le cadre des DDET et ont été appliquées à des systèmes de taille industrielle [Devooght 94] [Acosta 93].

IV.D.2.3 Simulation de Monte Carlo

La simulation de Monte Carlo est une méthode numérique basée sur le tirage de nombres aléatoires [Labeau 02 a et b]. Elle permet d'estimer l'espérance mathématique d'une variable aléatoire qui est une fonction de plusieurs paramètres (eux mêmes des variables aléatoires). Cette estimation est obtenue en moyennant les résultats issus d'un grand nombre d'histoires. Son utilisation dans les études de sûreté de fonctionnement permet de lever l'hypothèse markovienne et permet ainsi de traiter des systèmes à l'échelle industrielle.

Dans le cadre des études de sécurité des systèmes mécatroniques, l'estimation des probabilités des scénarios redoutés à partir d'un modèle comportemental est confrontée à la problématique des événements rares [Moncelet 98]. En effet, plus le scénario redouté est rare, plus le nombre d'histoires à simuler doit être élevé pour obtenir une bonne estimation. La durée de simulation devient, de ce fait, prohibitive. Différentes techniques ont été développées pour réduire le nombre d'histoires à réaliser. Parmi ces techniques, une nous semble prometteuse : la Méthode du Conditionnement Temporel (MCT) [Garnier 98]. D'autres techniques se sont focalisées sur la réduction de la durée d'une histoire. Dans le cadre de la fiabilité dynamique, la prise en compte de l'évolution de la partie continue alourdit considérablement la simulation d'une histoire. Le système passe, en effet, la majorité de son temps dans un état de bon fonctionnement alors que ce qui nous intéresse du point de vue sûreté de fonctionnement sont les états de dysfonctionnement (suite à des défaillances) susceptibles de provoquer le scénario redouté [Moncelet 98]. Afin d'accélérer la simulation d'une histoire en réduisant le coût des calculs dynamiques, on peut procéder de deux façons : soit en utilisant une technique de simulation « orientée événements » [Champagnat 98] quand la partie continue est simple ou simplifiable, soit en réduisant le temps de résolution des équations différentielles en utilisant des abaques, des courbes d'extrapolation ou les réseaux de Neurones [Marseguerra 94b] pour des systèmes plus compliqués.

La simulation orientée événements consiste à utiliser un simulateur à événements discrets qui détermine les dates d'occurrence des événements et va « faire des sauts » jusqu'au prochain événement, où il calculera le nouvel état courant et les nouvelles dates d'occurrence des événements. Cette technique permet de simuler des réseaux de Petri Prédicats-Transitions Différentiels en remplaçant l'intégration des variables continues par des temporisations calculées à partir des équations à intégrer et en calculant les valeurs des variables continues, lors des franchissements des transitions, en fonction des équations décrivant l'évolution des variables continues dans l'état précédent et de la durée de séjour dans l'état courant. Hormis quelques limitations contournables, cette technique est très prometteuse quant à la simulation de Monte Carlo des réseaux de Petri associés à des équations différentielles.

Cette technique a été appliquée par [Moncelet 98] pour l'évaluation de la probabilité d'occurrence des scénarios redoutés. Nous détaillerons ces travaux dans la section suivante.

V Un tour d'horizon

Ce tour d'horizon regroupe des présentations relativement détaillées de quelques travaux très proches de la problématique de mes travaux de thèse, celle de la recherche de scénarios redoutés. Nous présenterons tout d'abord les travaux de thèse de J.L. Chabot sur la simulation hybride des scénarios d'incendie. Ensuite, nous détaillerons la méthodologie des Graphes de Flux Dynamiques avant de finir par les travaux de thèse de G. Moncelet.

V.A Travaux de J. L. Chabot

V.A.1 Principe

Les travaux de J.L. Chabot [Chabot 98] portent sur une méthode pour le calcul de la probabilité d'extinction d'un scénario d'incendie. La particularité de ces scénarios est qu'ils font intervenir des aspects discrets et continus en interaction, ce qui situe ces travaux dans le cadre de la fiabilité dynamique. Les aspects discrets englobent les défaillances des systèmes de détection et de lutte ainsi que les actions humaines. Les dimensions, le développement et les effets du feu font partie de l'aspect continu. L'interaction mutuelle entre ces deux aspects justifie l'intérêt d'une modélisation hybride. La production de fumées et l'augmentation de la température dans les locaux ont une incidence directe sur la sollicitation des protections et sur les actions humaines à partir du franchissement de certains seuils. La réalisation ou l'échec de ces actions ont une influence sur le développement ultérieur de l'incendie.

V.A.2 Modélisation hybride avec les Réseaux de Petri

Chabot [Chabot 98] a proposé deux méthodes de prise en compte de l'aspect hybride en utilisant des modèles RdP stochastiques temporisés synchronisés (supporté par l'outil MOCA-RP©) et les a mis en oeuvre sur un cas test (Sysdyna). Il s'agit d'un réservoir contenant un liquide dont le niveau h doit être maintenu entre valeurs (h_0-1) et (h_0+1). Ce problème classique de régulation s'opère à l'aide de deux pompes (une pompe principale et une pompe de secours) ainsi que d'une vanne d'évacuation. Chacun des trois composants est commandé par une boucle de contrôle comprenant un détecteur de niveau. Ces composants peuvent subir des défaillances (comportement intempestif et blocage en l'état occupé) et sont non réparables. Ces défaillances sont régies par des lois exponentielles. Le but de cette étude est de déterminer les fréquences de débordement et d'assèchement du réservoir pour différentes durées de fonctionnement. L'évaluation de ces fréquences est faite par une simulation de Monte Carlo.

La première méthode consiste à prendre en compte les variations du niveau du liquide en discrétisant ce niveau et en associant à chaque variation élémentaire (Δh) l'apparition ou la disparition d'un jeton de la place modélisant le niveau. On peut choisir, par exemple, que l'apparition ou la disparition d'un jeton correspond à une variation du niveau Δh de 10 cm. Si le niveau initial du réservoir est fixé égal à 4 mètres, la place modélisant ce niveau contiendra initialement 40 jetons. Une pompe met 4 minutes pour faire passer le niveau de h à h + 10 centimètres lorsque la vanne d'évacuation est fermée. Quant aux états redoutés, on déclare qu'il y a assèchement du réservoir quand il y a moins de 30 jetons dans la place correspondant au niveau. Le débordement a lieu au delà de 70 jetons dans la même place.

La deuxième méthode consiste à utiliser des réseaux de Petri hybrides pour modéliser le comportement du réservoir associant des phénomènes discrets et continus. Pour ce faire, deux nouveaux types de transitions spéciales ont été introduits pour prendre en compte l'évolution du volume. On distingue les transitions dites d'activité continue et les transitions dites de jonction. Ces dernières mettent à jour les paramètres des équations algébriques ou différentielles (variation du débit de la vanne par exemple). Quant aux transitions d'activités continues, lorsqu'elles sont sensibilisées, MOCA-RP exécute un algorithme de calcul qui donne leur délai de temporisation avant franchissement (cette approche est donc similaire à celle de [Champagnat 98]). Ce dernier est calculé à partir d'équations algébriques ou différentielles. Ce calcul s'effectue par des lois « spéciales ». Dans le cas du réservoir, le calcul du niveau ainsi que le temps nécessaire pour atteindre un seuil sont directement

calculables à partir des équations sans passer par une discrétisation permettant de détecter les dépassements de seuils. Quand les équations régissant l'évolution de la partie continue sont très compliquées, elles sont résolues numériquement par des codes de calcul appropriés (Code_S ou Flamme_S dans le cas des scénarios d'incendie).

L'évaluation se fait par simulation de Monte Carlo. Appliquée au cas test du réservoir, la simulation hybride offre un temps de calcul 10 fois plus court que la simulation du modèle utilisant les RdP classiques en associant un certain nombre de jetons aux variations des grandeurs continues. Cette méthode a été appliquée à l'évaluation de scénarios d'incendie et a donné des résultats satisfaisants et prometteurs quant à son extension à des systèmes complexes.

V.A.3 Discussion

Ces travaux ont permis de montrer la bonne adéquation des réseaux de Petri quant à la modélisation des systèmes dynamiques hybrides non réparables, au sein desquels la partie continue est aussi importante que la partie discrète, ce qui justifie le recours aux codes de calcul. Le couplage entre un simulateur discret et un calcul continu permet de conserver le potentiel de modélisation de la partie continue et de la partie discrète, et de faire peu d'hypothèses simplificatrices lors de l'élaboration du modèle.

Le cas des scénarios d'incendie est un excellent exemple de systèmes dynamiques hybrides. Ils mélangent, en effet, interventions humaines, fonctionnement et dysfonctionnement de matériels aisément modélisables avec des RdP, mais également un feu qui a des effets non négligeables sur les interventions humaines et le fonctionnement de certains matériels de détection et de lutte.

La simulation de Monte Carlo donne des résultats satisfaisants pour estimer la probabilité d'extinction du feu. Elle peut être également utile pour déterminer les modifications à apporter à l'organisation de la lutte contre l'incendie, celles-ci ayant un effet sensible sur cette probabilité. La simulation de Monte Carlo opérant à partir d'un couplage entre un simulateur discret et un code continu relativement compliqués est rendue possible et satisfaisante (temps de simulation raisonnables) car on n'est pas confronté à la problématique des événements rares. Si c'était le cas, le nombre d'histoires jouées aurait été insuffisant et le temps de simulation d'une seule histoire aurait été augmenté. Par conséquent, la simulation de Monte Carlo aurait certainement nécessité des temps prohibitifs.

V.B Méthode des graphes de flux dynamiques

V.B.1 Principe

La **méthode des graphes de flux dynamiques** (**D**ynamic **F**lowgraph **M**ethodology **DFM**) [Garret 95] est une technique de modélisation et d'analyse pour étudier les risques des systèmes embarqués (systèmes de contrôle numérique : digital control systems). Elle a été utilisée dans les études de risque des systèmes aérospatiaux et nucléaires. Cette approche a pour but d'identifier les scénarios redoutés et d'élaborer une stratégie de test des systèmes étudiés. Elle permet de prendre en compte l'aspect dynamique des systèmes embarqués et l'interaction entre la partie logicielle et la partie matérielle des systèmes.

Elle est composée de deux étapes : une étape de modélisation suivie d'une étape d'analyse. L'étape de modélisation consiste à construire un modèle exprimant le

comportement logique et dynamique du système. Ce modèle se présente sous la forme d'un réseau de nœuds. Ces nœuds sont reliés entre eux par des connecteurs pour exprimer les relations de causalité (relations logiques) et temporelles (comportement dynamique du système) entre les variables physiques de la partie opérative et les paramètres de commande du système de contrôle. Une fois l'étape de modélisation finie, commence l'étape d'analyse. Elle a pour objectif de déterminer comment le système atteint un état donné (état normal ou indésirable). On procède de la façon suivante : à partir du modèle DFM, on inverse les relations de causalité et on construit des Arbres de Défaillance temporisés. Ces derniers expriment non seulement les combinaisons booléennes d'événements (défaillances ou autres) pouvant amener le système dans un état donné, comme dans un Arbre de Défaillance classique, mais aussi les scénarios redoutés sous la forme de séquences d'occurrence d'événements.

V.B.2 La modélisation

La méthode des graphes de flux dynamiques introduit un ensemble de symboles (des nœuds et des connecteurs), décrits ci-dessous, qui permettent de construire les modèles DFM supports de l'analyse. Un exemple simple de modèle DFM sera présenté afin d'illustrer les différentes notions introduites ci-dessus et de montrer comment générer un Arbre de Défaillance temporisé lors de la phase d'analyse.

V.B.2.1 Les nœuds

V.B.2.1.a Les nœuds associés aux variables physiques (Process Variable Node)

Ils représentent les variables de la partie opérative ou bien les variables continues du système de commande représentant les grandeurs physiques du système commandé. Chaque variable est discrétisée en un nombre fini d'états.

Prenons par exemple une variable V indiquant le volume d'un liquide dans un réservoir (Figure 1.4). Cette variable physique peut être discrétisée en 5 états : 0, 1, 2, 3 ou 4 selon que le niveau du liquide est estimé respectivement très bas, bas, moyen, haut ou très haut.

Etat	Variation (en cm^3)
0	[0, 10[
1	[10, 30[
2	[30, 70[
3	[70, 90[
4	[90, 100]

Figure 1.4. **Un nœud physique et sa discrétisation**

Cette discrétisation a pour but de pouvoir énumérer les relations de causalité entre les différentes variables du système avec un point de vue qualitatif. Ces relations sont exprimées sous forme de tables de décision que nous expliquerons par la suite. Par conséquent, chaque nœud peut être considéré comme un vecteur d'état dont les composantes sont les domaines de valeur de la variable associée.

V.B.2.1.b Les nœuds conditionnels (Causality Nodes)

Ils représentent soit les états de défaillance des composants, soit l'état de la commande. Ces nœuds décrivent les changements de modes de fonctionnement du système après une défaillance ou un ordre de la commande.

Sur la figure 1.5, le nœud conditionnel **C** représente l'état (de bon fonctionnement ou de défaillance) d'un capteur de volume dans un réservoir. La relation de causalité entre le nœud Vr (volume réel) et le nœud Vm (volume mesuré) dépend de la valeur associée au nœud conditionnel **C**.

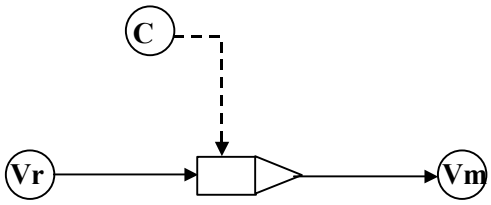


Figure 1.5. **Nœud conditionnel EV**

V.B.2.2 Les connecteurs

V.B.2.2.a Les boîtes de transfert (Transfer Boxes)

Les boîtes de transfert relient les nœuds associés aux variables d'état du système. Elles expriment un lien de causalité instantanée entre les variables d'entrée (la cause, nœud D sur la figure 1.6) et les variables de sortie (l'effet, nœud H sur la même figure).

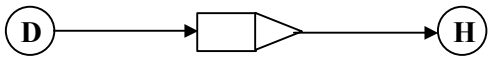


Figure 1.6. **Boîte de transfert**

V.B.2.2.b Les tables de décision (Decision Tables)

Une table de décision fait correspondre à toute combinaison de valeurs des nœuds d'entrée une valeur pour les nœuds de sortie. Une table de décision est associée à chaque boîte de transfert. Dans le cas où un nœud conditionnel est attaché à une boîte de transfert, une table de décision est associée à chaque valeur du nœud conditionnel.

Considérons l'exemple de la figure 1.5. Supposons que lorsque le capteur de volume est en bon état (C=1), le volume mesuré soit égal au volume réel dans le réservoir et qu'en cas de défaillance du capteur (C=0), le volume mesuré devienne nul. Nous obtenons les tables de décision suivantes :

Vr	Vm
0	0
1	1
2	2
3	3
4	4

Table pour C=1

Vr	Vm
0	0
1	0
2	0
3	0
4	0

Table pour C=0

V.B.2.2.c Les boîtes de transition (Transition Boxes)

Les boîtes de transition relient, au même titre que les boîtes de transfert, des nœuds d'entrée à des nœuds de sortie. Toutefois, le changement des valeurs des nœuds de sortie ne se fait plus instantanément mais avec un retard par rapport au changement des valeurs des nœuds d'entrée associés. Un délai est associé à chaque boîte de transition.

Par exemple, considérons une procédure qui calcule la valeur d'une variable **C** à partir des valeurs de deux variables A et B. Si cette procédure a un temps d'exécution égal à 10 ms, nous aurons la représentation donnée par la figure 1.7.

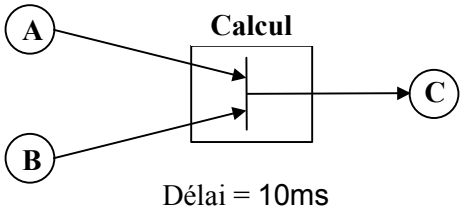
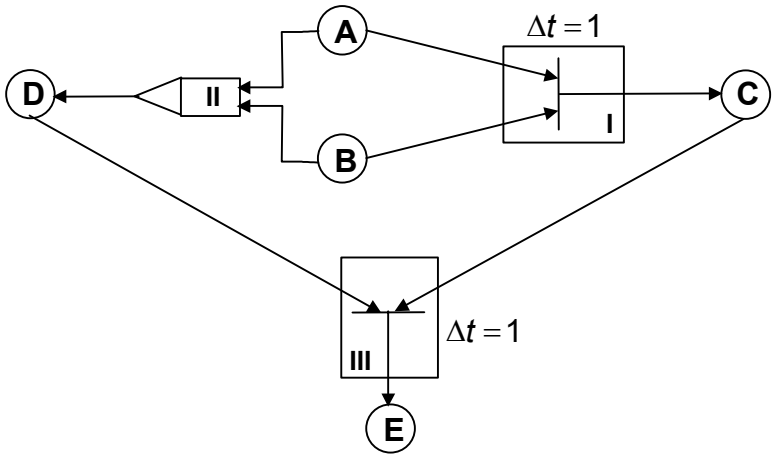


Figure 1.7. **Boîte de transition**

V.B.3 Exemple

Prenons un exemple afin d'illustrer comment nous construisons un arbre de défaillance temporisé à partir d'un modèle DFM. Considérons le modèle de la figure 1.8. Ce modèle est composé de 5 nœuds physiques (A, B, C, D et E) reliés entre eux par 3 connecteurs : 2 boîtes de transition et une boîte de transfert. Chaque nœud physique est discrétisé en 3 états (0, 1, 2). Une table de décision (I, II ou III) est associée à chaque connecteur (respectivement I, II et III).



A	B	C
0	0	1
0	1	0
0	2	1
1	0	1
1	1	1
1	2	2
2	0	2
2	1	1
2	2	1

Table de décision I

A	B	D
0	0	2
0	1	1
0	2	2
1	0	2
1	1	2
1	2	2
2	0	2
2	1	0
2	2	2

Table de décision II

C	D	E
0	0	0
0	1	2
0	2	1
1	0	0
1	1	0
1	2	1
2	0	2
2	1	1
2	2	0

Table de décision III

Figure 1.8. **Exemple de modèle DFM**

V.B.4 Procédure de construction d'un Arbre de Défaillance temporisé

Nous nous proposons d'analyser comment l'état E=2 est atteint dans l'exemple de modèle DFM ci-dessus. Ceci est effectué en construisant un arbre de défaillance ayant comme sommet l'état E=2. La procédure de construction est décrite ci-dessous et l'arbre résultant est représenté sur la figure 1.9. Le principe général est le suivant : à chaque pas, nous construisons toutes les branches de l'Arbre de Défaillance.

A partir de la table de décision III, deux combinaisons des valeurs des nœuds C et D correspondent à l'état E=2 : (C=0 ET D=1) OU (C=2 ET D=0). Ceci est représenté au premier niveau de l'arbre de la figure 2.6 sous l'état E=2. Les nœuds C et D étant reliés à E par une boîte de transition, il est indiqué sur l'arbre de défaillance (par un trait en pointillé) qu'il s'est écoulé une unité de temps (délai associé $\Delta t = 1$) entre le moment où les nœuds C et D ont atteint les valeurs mentionnées et le moment où E atteint la valeur 2. Si on prend comme origine du temps l'instant ***t*** où E est devenu égal à 2. C et D ont atteint les valeurs mentionnées à l'instant ***t-1***.

Comme C est un nœud de sortie d'une boîte de transition et que D ne l'est pas, intéressons nous tout d'abord au nœud D. Cherchons les combinaisons de valeurs des nœuds A et B qui ont engendré instantanément les états D=0 et D=1. Construisons la branche de l'arbre correspondante à D=0. D'après la table de décision II, ceci a lieu uniquement quand (A=2 ET B=1). De même pour le cas où D=1, nous trouvons la condition (A=0 ET B=1). C'est ce qui est représenté dans l'arbre de défaillance en dessous des événements D=0 et D=1. La construction des branches correspondantes se termine puisque A=0, A=2 et B=1 sont tous des événements de base.

Etudions maintenant le nœud C et les causes amenant aux états C=0 et C=2. Selon la table de décision I, C=0 à l'instant ***t-1*** si et seulement si (A=0 ET B=1) à l'instant ***t-2*** car C est en sortie d'une table de décision dont le délai associé est $\Delta t = 1$. De même, on obtient C=2 si et seulement si (A=1 ET B=2) OU (A=2 ET B=0). La construction de l'arbre s'arrête car tous les événements obtenus sont des événements de base.

Si nous considèrons que les valeurs des nœuds A et B peuvent varier dans le temps, nous obtenons finalement un ensemble de séquences caractérisées par des suites d'états amenant à l'état E=2 :

- {[(A=0 ET B=1)/t=t-2] ET [(A=0 ET B=1)/t=t-1]},
- {[(A=1 ET B=2)/t=t-2] ET [(A=2 ET B=1)/t=t-1]},
- {[(A=2 ET B=0)/t=t-2] ET [(A=2 ET B=1)/t=t-1]}.

Si nous supposons que A et B sont constants dans le temps, une seule combinaison mène vers l'état concerné. Il s'agit de la condition (A=0 ET B=1). Les autres affecteraient en effet des valeurs contradictoires aux variables A et B.

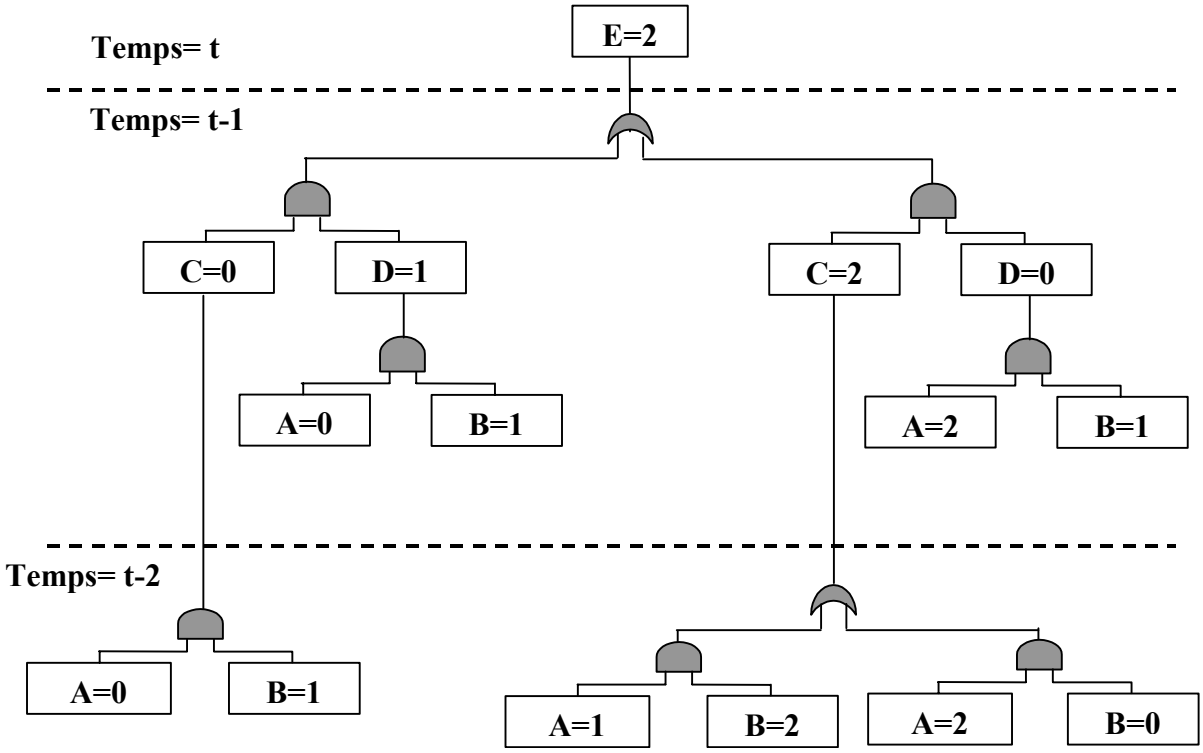


Figure 1.9. **Arbre de défaillance temporisé**

V.B.5 Discussion

L'un des buts de cette approche est de déterminer comment un système peut atteindre un état donné. Cela passe par une modélisation appropriée du système basée sur les aspects logiques et sur les relations de causalité entre les différentes variables du système. Cette approche permet de rompre avec les approches classiques basées sur une analyse séparée des deux parties logicielles et physiques d'un système embarqué. En effet, elle permet une prise en compte explicite des interactions entre la partie opérative et la commande de ces systèmes.

Cette méthode nous paraît bien adaptée pour les études de sûreté de fonctionnement durant la phase de conception des systèmes de commande et de surveillance. En effet, durant cette phase, les concepteurs changent et améliorent fréquemment les modèles développés. Une fois qu'un Arbre de Défaillance est fait pour un modèle par rapport à un état redouté donné, modifier le modèle implique très souvent de refaire un autre Arbre de Défaillance, ce qui est souvent relativement lourd. Or, dans le cas de cette approche l'unicité du modèle et la construction automatique des Arbres de Défaillance amènent une grande flexibilité dans le travail des concepteurs, puisqu'il suffit de faire les modifications sur le modèle et de générer automatiquement les Arbres de Défaillances, et ce pour plusieurs états redoutés. Un autre avantage d'importance pour cette méthode est qu'elle est basée sur la construction d'Arbres de Défaillance (temporisés) bien connus dans la communauté des fiabilistes, ce qui la rendrait plus accessible que d'autres méthodes.

Toutefois, discrétiser systématiquement toutes les variables du système étudié représente un inconvénient majeur de cette méthode quand il s'agit de systèmes de taille industrielle. Cela engendre en effet des modèles de très grande taille. Nous devons faire face à un problème d'explosion combinatoire des états et des séquences qui risque de rendre toute la démarche infaisable. De plus, nous n'avons aucune idée des incertitudes liées à cette

discrétisation. L'impact de la granularité de la discrétisation peut en effet être grand et faire apparaître des scénarios inexistants.

Un deuxième inconvénient est à signaler. Il concerne l'incapacité de cette méthode à prendre en compte les caractéristiques stochastiques des composants des systèmes, par exemple leur taux de défaillance. Ceci nous amène vers un autre point qui est celui de la quantification des événements redoutés. Cette méthode produit des Arbres de Défaillances temporisés montrant comment un système pourrait évoluer vers un état redouté donné. Pour pouvoir déterminer la probabilité d'être dans un tel état, il est indispensable de connaître la dynamique des variables du système c'est-à-dire comment ces variables passent d'un état à un autre et non seulement la probabilité de défaillance des composants du système.

Enfin, un dernier inconvénient de cette méthode est que le modèle sur lequel elle est basée n'est pas sous-tendu par une démarche d'analyse formelle. En effet, comment pourrait-on vérifier que le modèle du système corresponde bien à la réalité ? Comment vérifie-t-on qu'il ne comporte pas d'incohérence, de blocage, … ? Ces arguments jouent en la faveur des modèles formels comme le formalisme des réseaux de Petri, modèle choisi par G. Moncelet pour modéliser les systèmes mécatroniques.

V.C Travaux de G. Moncelet

Soutenue en octobre 1998, la thèse de G. Moncelet [Moncelet 98] traite de l'évaluation qualitative et quantitative de la sûreté de fonctionnement des systèmes mécatroniques automobiles (au sens défini auparavant). Plus particulièrement, ces travaux portent sur la détermination des séquences d'événements redoutés et l'estimation de leurs probabilités d'occurrence par la simulation (simulation de Monte Carlo). A partir d'une modélisation appropriée des systèmes mécatroniques (modèle qualitatif) à l'aide du formalisme des réseaux de Petri colorés [Jensen 92], le graphe d'occurrence (ou graphe des marquages accessibles du réseau de Petri coloré) est généré et tous les chemins menant vers un état redouté donné sont identifiés et caractérisés en termes d'enchaînements d'actions et de changement d'états des composants concernés du système. L'estimation de la probabilité d'occurrence de ces scénarios redoutés se fait par une simulation de Monte Carlo du modèle quantitatif du système.

V.C.1 Formalisme et outil de modélisation

Comme nous venons de le voir, le formalisme choisi pour la modélisation des systèmes mécatroniques proposé par Gilles Moncelet est celui des réseaux de Petri colorés. En associant des couleurs aux jetons et des ensembles de couleurs aux places, ce formalisme permet de construire des modèles plus compacts que ceux obtenus en utilisant les réseaux de Petri ordinaires (moins de places et de transitions). L'utilisation de l'outil DesignCPN© [Jensen 97] lui a permis d'éditer de manière hiérarchique, de vérifier et de simuler des modèles en réseaux de Petri colorés temporisés (des temporisations sont associées aux jetons). Cet outil permet également de construire le graphe d'occurrence et de l'exploiter quand les champs des jetons prennent leurs valeurs sur des ensembles finis.

L'utilisation des réseaux de Petri colorés permet de manipuler des informations portées par des jetons lors du franchissement des transitions impliquant ces jetons. En effet, ce formalisme permet d'associer une action à une transition : un calcul y sera effectué pour instancier une variable sur un arc de sortie. Ceci permet alors de pouvoir mettre à jour les variables continues du système afin de reproduire la dynamique de la partie continue.

Toutefois pour pouvoir construire le graphe d'occurrence, les variables continues doivent être discrétisées.

V.C.2 Principe de la modélisation

Le modèle (hybride) est obtenu en représentant les configurations par des places et les variables d'état continues par des attributs associés aux jetons.

Afin de reproduire la dynamique de la partie continue, la mise à jours des variables continues du système se fait en discrétisant l'axe des temps selon l'un des deux principes suivants :

- Par échantillonnage. On obtient un modèle dit « détaillé » du système.
- En ne s'intéressant qu'aux événements qui correspondent à des changements de configurations. On obtient un modèle dit « abstrait ».

Nous rappelons tout d'abord en quoi consiste le principe de l'échantillonnage selon les inscriptions de DesignCPN avant de présenter les modèles « détaillé » et « abstrait ».

V.C.2.1 Prise en compte de la partie continue selon le principe de l'échantillonnage

Dans cette démarche, la dynamique continue est supposée régie par un ensemble d'équations algébro-différentielles du premier ordre. On utilise le principe de l'échantillonnage afin de prendre en compte l'évolution des variables continues quand le système est dans une configuration donnée.

La mise à jour des variables continues doit être assez fine (voir Figure 2.1) : on remplace l'intégration des équations différentielles par une linéarisation, en prenant un pas de calcul aussi petit que la précision des résultats est grande. La fréquence de mise à jour des variables continues peut être fixe ou variable (selon la dynamique continue).

La figure 1.10 montre comment on peut mettre à jour les variables continues (le volume par exemple) en utilisant les inscriptions associées aux transitions dans DesignCPN. A chaque pas de temps (dt), le jeton dans la place « FREQUENCE » devient disponible et la transition est franchissable. La procédure de calcul associée à cette transition est alors exécutée et effectue la mise à jour de l'information portée par le jeton de la place « VOLUME » qui a pour attribut la valeur du volume.

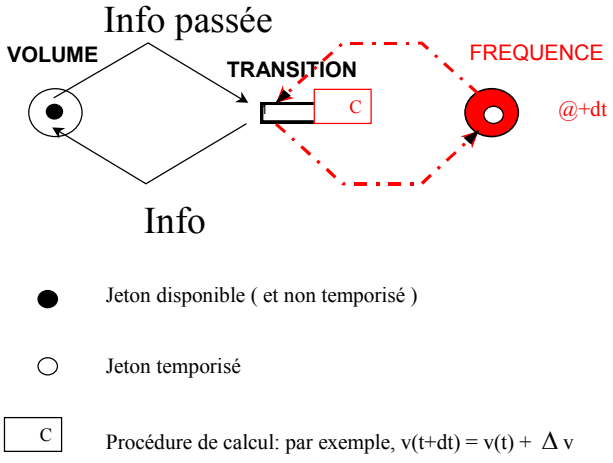


Figure 1.10. **Principe de l'échantillonnage selon les inscriptions de DesignCPN**

V.C.2.2 Modèle détaillé

Ce modèle représente explicitement l'échantillonnage que réaliserait le calculateur pilotant la partie opérative. Le pas d'échantillonnage doit être suffisamment petit pour prendre en compte l'évolution des variables continues quand le système est dans une configuration donnée. Quant aux événements faisant changer le système de configuration (une défaillance par exemple), leur prise en compte n'est pas faite à des dates prédéfinies (comme c'est le cas par échantillonnage) mais dès leur instant d'occurrence. A cet instant, une mise à jour des valeurs des variables d'état du système est effectuée (autant les variables discrètes que les variables continues) et l'instant courant est mémorisé.

V.C.2.3 Modèle abstrait

Dans le modèle détaillé, une mise à jour de toutes les variables d'état du système est effectuée lorsque le système change de configuration, mais également à chaque pas d'échantillonnage. Sachant que le but est de simuler un grand nombre d'histoires comme c'est le cas pour la simulation de Monte Carlo, la fréquence de cette mise à jour alourdit considérablement la simulation. On cherche par conséquent à éviter le calcul des variables continues en dehors des changements de configuration. Cela veut dire que l'on cherche à éviter la simulation de la dynamique des variables continues entre les changements de configuration pour conserver un principe de simulation à événements discrets pilotée par les événements.

Si on est capable de calculer l'instant auquel une variable continue franchit un seuil donné (de contrôle ou de sécurité) et qu'à tout instant, on est capable de calculer toutes les variables du système, alors, l'échantillonnage n'est plus indispensable. En effet, un modèle orienté événement peut reproduire l'interaction entre la partie opérative, la partie commande et le modèle des défaillances, c'est le modèle dit « abstrait ». Il ne reproduit que les événements significatifs (du point de vue de la sûreté de fonctionnement) faisant évoluer la configuration du système, cela sans reproduire explicitement les variations de la partie continue à tout instant. Ces événements peuvent être une défaillance, un dépassement de seuil d'une variable continue (seuil de contrôle ou de sécurité) ou un changement de la valeur d'un paramètre de la partie opérative.

Le principe du modèle abstrait est le suivant : chaque configuration ou mode de fonctionnement du système est représenté par une place. Le jeton associé à cette place porte une temporisation représentant la durée du mode de fonctionnement en cours (c'est-à-dire le délai au bout duquel il y aura un changement de mode de configuration). La valeur de cette temporisation, la valeur de la variable continue associée au jeton et sa date de mise à jour sont calculées à chaque sollicitation correspondant à un événement significatif.

V.C.2.4 Modèle qualitatif

Le modèle qualitatif est déduit directement du modèle quantitatif en supprimant les temporisations associées aux jetons, aussi bien les jetons associés à la mise à jour de la partie continue que ceux indiquant le délai de la prochaine défaillance.

V.C.3 Analyse quantitative par simulation de Monte Carlo

Le but de cette analyse est d'estimer la probabilité d'occurrence des événements redoutés par une simulation de Monte Carlo. On simule le modèle global du système (fonctionnel et dysfonctionnel) pendant sa durée de vie (c'est une histoire). Si un événement redouté apparaît, on arrête la simulation de cette histoire, on comptabilise le nombre d'événements redoutés et on recommence une nouvelle histoire. Si aucun événement redouté ne se produit,

on continue la simulation jusqu'à une date limite et on recommence une nouvelle histoire. La probabilité de l'événement redouté étudié est estimée par le quotient entre le nombre d'histoire ayant abouti à l'occurrence de cet événement et le nombre global d'histoires (un très grand nombre afin d'obtenir une bonne précision, de l'ordre de 100 000 histoires).

Le modèle support de cette analyse pourrait être le modèle détaillé ou le modèle abstrait. En fait, il faut utiliser le modèle abstrait pour éviter des temps de simulation démesurés. Cela est souvent possible car le modèle abstrait contient l'essentiel de l'information utile du point de vue de la sûreté de fonctionnement. Nous verrons par la suite que même en se basant sur ce type de modèle, la simulation de Monte Carlo nécessite un temps de calcul prohibitif.

Il est à noter qu'aucune hypothèse sur les scénarios menant aux événements redoutés n'est nécessaire, a priori, pour l'analyse quantitative.

V.C.4 Analyse qualitative par génération du graphe d'occurrence

Cette analyse a pour objectif d'énumérer tous les scénarios redoutés sous forme de séquences d'événements. Elle permet également la vérification des « bonnes » propriétés que l'on attend d'un modèle en réseau de Petri. Elle est basée sur la construction du graphe des marquages accessibles du réseau de Petri coloré (dit aussi graphe d'occurrence), fonction supportée par l'outil DesignCPN lorsque les variables prennent leurs valeurs dans des ensembles finis.

Rappelons tout d'abord la définition d'une **composante fortement connexe** (notée **cfc**). C'est un ensemble de nœuds dans lequel tout nœud est accessible depuis n'importe quel autre. Cette notion de **cfc** est utile pour la recherche des scénarios redoutés. En effet, tout mode de fonctionnement du système est nécessairement inclus dans une composante fortement connexe non triviale (c'est-à-dire une **cfc** contenant un seul nœud dont aucun arc de sortie n'a pour destination lui même). Les passages d'une **cfc** à une autre correspondent donc à des dégradations irréversibles du système. Elles font passer d'un mode de fonctionnement à un autre et mènent ainsi à un état de défaillance totale (marquage puits).

L'analyse qualitative est réalisée en trois étapes : le graphe d'occurrence et le graphe des **cfc** sont tout d'abord calculés par DesignCPN. On identifie ensuite les modes de fonctionnement aux **cfc** correspondantes et les états redoutés aux états de blocage du graphe. Enfin, pour chaque **cfc**, tous les chemins menant aux plus proches **cfc** sont construits et les scénarios correspondants sont identifiés.

V.C.5 Résultats

Compte tenu de la complexité inhérente aux systèmes mécatroniques, les deux principaux résultats auxquels ont abouti les travaux de Gilles Moncelet sont les suivants :

- La simulation de Monte Carlo du modèle quantitatif nécessite un temps prohibitif,
- L'analyse qualitative est confrontée au problème de l'explosion combinatoire du nombre d'états du graphe d'occurrence.

En ce qui concerne l'analyse quantitative, la rareté des scénarios redoutés rend inefficaces les méthodes basées sur la seule simulation de Monte Carlo du modèle global. En effet, le système passe la majorité de son temps dans un fonctionnement nominal qui n'apporte aucune information sur la sûreté de fonctionnement. Seule nous intéresse la réaction du système face à l'occurrence d'une ou de plusieurs défaillances. On pourrait par conséquent envisager de ne

simuler le comportement du système qu'à la suite de l'occurrence d'une défaillance et d'arrêter cette simulation quand le système retrouve un fonctionnement nominal (après une reconfiguration réussie). Toutefois, se pose le problème de la reconstitution de l'état du système à l'instant de l'occurrence de cette défaillance et de sa probabilité d'occurrence.

Outre la complexité des systèmes mécatroniques (nombre important de composants en interaction et complexité des lois de pilotage), l'explosion combinatoire du nombre d'états du graphe d'occurrence est due d'une part à la discrétisation de la partie continue et d'autre part à l'entrelacement.

La prise en compte de la dynamique continue par discrétisation de l'intervalle de variation des variables continues en un certain nombre de sous intervalles (pertinent du point de vue sûreté de fonctionnement) contribue à accroître le nombre d'états global du graphe d'occurrence. En effet, prenons l'exemple d'un système dont la dynamique continue est régie par l'évolution de deux variables d'états continues. Supposant que l'intervalle de variation de chaque grandeur soit représenté par trois sous intervalles (chacun correspondant à un niveau bas, moyen ou haut). C'est le passage d'un sous intervalle à un autre qui régit la dynamique continue (le cycle {bas, moyen, haut, moyen, bas} dans le cas d'une régulation classique). Le nombre d'états résultant de la prise en compte de la partie continue est de 3^2. Le nombre global d'états du système augmente exponentiellement avec le nombre de sous intervalles nécessaires pour représenter la dynamique des variables continues.

Dans un graphe d'occurrence, le parallélisme est représenté par entrelacement. En effet, pour deux transitions parallèles t et t', leur représentation par entrelacement donne les deux séquences suivantes : {t ; t'} et {t' ; t}. Cet entrelacement dans le traitement du parallélisme engendre une multiplication des séquences dans le graphe qui contribue à alourdir d'une part la construction, et, d'autre part le traitement a posteriori du graphe. Il est pourtant, en général, non significatif. Agréger les composantes fortement connexes sous la forme d'un nœud unique permet l'élimination de certains comportements non significatifs, mais cela peut aussi cacher des relations de causalités significatives.

V.C.6 Discussion

Les travaux de Gilles Moncelet traitent de l'évaluation de la sécurité prévisionnelle des systèmes mécatroniques par une analyse qualitative et quantitative à base de réseaux de Petri colorés. La modélisation adoptée consiste à prendre en compte les aspects discrets (défaillances et commandes) et continus (dynamique des variables énergétiques) et de les regrouper au sein d'un même modèle : un réseau de Petri coloré. Cela revient à discrétiser les aspects continus.

L'analyse quantitative repose sur la simulation de Monte Carlo du modèle du système afin d'estimer la probabilité d'occurrence des événements redoutés. Cette analyse montre qu'il est plus judicieux d'effectuer la simulation à partir du modèle abstrait sans se préoccuper de la représentation explicite de l'échantillonnage. Toutefois, cela suppose des hypothèses et des approximations fortes sur la dynamique de la partie continue du système. Pour les systèmes complexes, ce type de simulation nécessite des temps de calcul prohibitifs. Deux voies ont été abordées pour accélérer la simulation. La première consiste à remplacer le joueur de réseau de Petri, très lent, par un code compilé en langage ML (langage fonctionnel supporté par DesignCPN). Une accélération sensible a été obtenue. Toutefois, cette solution reste très contraignante car elle oblige le concepteur à traduire le réseau de Petri en langage ML.

La deuxième voie pour accélérer la simulation consiste à caractériser, au préalable, les scénarios redoutés de manière qualitative afin de ne simuler que les comportements susceptibles de conduire vers une situation dangereuse. Ceci a donc conduit à effectuer une analyse qualitative de la sécurité des systèmes mécatroniques. Cette analyse qualitative est basée sur la génération du graphe d'occurrence à partir d'un modèle qualitatif. Cette étude a montré que ce graphe explose avec le nombre de défaillances et le nombre de niveaux discrets nécessaires pour prendre en compte la dynamique continue. La recherche de scénarios par exploration de ce graphe est d'autant plus difficile qu'il contient bon nombre d'informations sans intérêt car décrivant des comportements sans défaillances.

VI Synthèse

Après ce tour d'horizon sur les principaux travaux récents ayant attrait à la fiabilité dynamique des systèmes complexes, nous allons résumer leurs principales contributions. Au travers de cette synthèse, nous évoquerons ce que nous avons retenu de ces travaux et développé dans le cadre de mes travaux de thèse.

Les travaux de J.L. Chabot nous ont montré la bonne adéquation des réseaux de Petri à la modélisation de systèmes dynamiques hybrides et les bonnes performances de la simulation de Monte Carlo couplant un joueur de réseau de Petri et un code continu pour l'évaluation des scénarios d'incendie. Toutefois, cette méthode a été appliquée dans un cas particulier : le scénario redouté à évaluer est connu. C'est pourquoi nous avons considéré qu'avant toute analyse quantitative il était nécessaire d'élaborer une méthode systématique de détermination des ces scénarios. S'ils sont inconnus, toute simulation hybride est vouée à des temps de simulation prohibitifs surtout quand il s'agit de scénarios redoutés très rares (problématique des événements rares). C'est effectivement le cas dans les systèmes mécatroniques au travers de toutes les reconfigurations prévues.

Quant à la méthodologie des Graphes de flux dynamiques, elle permet de prendre en compte l'aspect dynamique des systèmes embarqués ainsi que l'interaction entre la partie logicielle et la partie matérielle des systèmes. A partir de la construction d'un seul modèle du système étudié, il est possible de générer automatiquement des arbres de défaillance, et ce, pour chaque événement redouté. Cette approche est très bien adaptée aux études de sécurité pendant la phase de conception où les modèles ne cessent d'évoluer. Ce que nous retenons de cette méthode, c'est la prise en compte explicite des interactions entre la partie logicielle et la partie matérielle sous la forme de relations de causalité et la génération automatique de modèles de scénarios redoutés (arbres de défaillance temporisés).

Nous venons enfin aux travaux de thèse de G. Moncelet. En plus de la bonne prise en compte de l'aspect hybride, la principale contribution de G. Moncelet est d'avoir montré la nécessité de séparer les études de sécurité des systèmes mécatroniques en deux parties : une analyse qualitative suivie d'une analyse quantitative. L'analyse qualitative a été effectuée à partir de la génération du graphe d'accessibilité et permet de déterminer les scénarios redoutés, et l'analyse quantitative a été basée sur la simulation de Monte Carlo et permet d'estimer les probabilités d'occurrence de ces scénarios. L'explosion combinatoire du nombre d'états du graphe était le principal obstacle face à cette technique d'analyse qualitative, tandis que l'analyse quantitative souffre des temps de simulation prohibitifs dus à la prise en compte par discrétisation de la partie continue. L'extraction des scénarios redoutés à partir du graphe d'occurrence n'est pas efficace car, en raison de l'entrelacement, les relations de cause à effet entre les événements sont gommées. Eviter le passage par ce graphe nous semble donc nécessaire.

En conclusion, ce que nous retenons de cet état de l'art par rapport à nos travaux c'est le choix du modèle des réseaux de Petri pour la modélisation de l'aspect discret des systèmes mécatroniques, le couplage de ce modèle avec un modèle de la partie continue (des équations algébro-différentielles) pour l'aspect hybride et la nécessité d'extraire les scénarios redoutés directement à partir d'un modèle du système sans passer par le graphe d'accessibilité. Comme dans le cas des DFM, l'analyse des relations de cause à effet (ou relations de causalité) dans le modèle permettrait une meilleure caractérisation des scénarios redoutés.

Chapitre 2. Modélisation hybride et logique

I Introduction

Comme nous venons de le montrer dans le chapitre précédent, les réseaux de Petri constituent l'outil de modélisation retenu pour l'aspect discret des systèmes mécatroniques. Ce modèle à événements discrets possède certaines propriétés essentielles : possibilité de hiérarchisation, expression du parallélisme, du partage de ressources et validation formelle de certaines propriétés. Dans ce chapitre, nous étudierons différents formalismes, à base de réseaux de Petri, qui ont été étendus afin de prendre en compte un aspect hybride. Plus précisément, nous comparerons les trois principaux types d'approches qui ont été développés dans la communauté académique, puis nous en retiendrons un. Notre critère de choix est de retenir une approche permettant une bonne séparation entre les aspects discret et continu. En effet, par la suite, nous allons nous fonder sur la seule représentation de l'aspect discret pour analyser les relations de cause à effet entre les événements. Il nous faudra toutefois étendre l'approche retenue afin de permettre la prise en compte de l'aspect stochastique, nécessaire pour la représentation des mécanismes de défaillance et de reconfiguration des systèmes mécatroniques.

II Aspect hybride

II.A Approches de modélisation de l'aspect hybride avec les RdP

Les études SdF des systèmes mécatroniques nécessitent la prise en compte de l'aspect discret et de l'aspect continu. Or, quand les deux aspects sont étroitement liés, et que les degrés d'intégration et de complexité sont élevés, la modélisation du système par une seule composante n'est pas possible [Dubois 90]. Les réseaux de Petri constituent l'outil de modélisation retenu pour l'aspect discret des systèmes mécatroniques. La modélisation de l'aspect hybride de ces systèmes avec les réseaux de Petri est alors abordée selon deux tendances : d'une part, la tendance reposant sur l'intégration des aspects continu et discret au sein du même formalisme, et d'autre part, celle associant un formalisme différent pour chacun des deux aspects, les RdP étant le modèle retenu pour représenter la partie discrète.

Dans sa thèse, D. Andreu [Andreu 96] a établi un état de l'art complet sur les différentes approches de modélisation de l'aspect hybride : les approches séparée et intégrée. Dans un premier temps, nous comparerons brièvement ces deux approches et nous présenterons ensuite, plus en détail, les principales méthodes de modélisation de l'aspect hybride, celles basées sur trois formalismes : les réseaux de Petri haut niveau, les RdP hybrides et les RdP couplés avec des équations différentielles. Ce dernier correspond à une approche de modélisation séparée. Quant aux deux premiers, ils s'inscrivent dans le cadre de l'approche intégrée.

La technique de modélisation basée sur un modèle intégré montre une unification des aspects continu et discret au sein d'un même formalisme, et ce au détriment de la délimitation entre le continu et le discret. Cette absence de délimitation est à l'origine de certains

inconvénients. Pour les RdP hybrides, cela fait apparaître des événements qui ne sont pas associés à des tirs de transitions et qui correspondent simplement à des changements de vitesse de franchissement sur les transitions continues. En utilisant les RdP haut niveau, la mise à jour de la partie continue par la technique de l'échantillonnage fait rajouter au modèle du système des transitions artificielles.

La technique de modélisation séparée, basée sur l'utilisation de deux modèles coopérants, un modèle discret et un modèle continu, conserve un caractère plus général. L'utilisation de modèles propres à chaque aspect permet de conserver le potentiel de modélisation dans chacun des domaines [Valentin 93]. Le principe de séparation des deux aspects nous semble par conséquent important. *Les aspects continu et discret correspondent à deux mondes mathématiques différents offrant deux vues différentes du système. La vue discrète n'est pas seulement une abstraction du phénomène continu ; les deux vues sont complémentaires* [Andreu 96].

II.A.1 Modélisation avec les RdP de haut niveau

Les réseaux de Petri de haut niveau regroupent toutes les approches permettant de manipuler des informations. Parmi ces approches, on peut citer les RdP colorés [Jensen 92], les RdP Prédicats Transitions [Champagnat 98a], les RdP à objets [Valette 92], etc. Un n-uplet de variables est associé au jeton, et lors des franchissements des transitions, il est possible de modifier les variables des jetons (uniquement celles associées à des jetons impliqués par le franchissement).

Avec une structure de données complexe attachée aux jetons et des durées associées aux transitions, les RdP de haut niveau peuvent représenter une grande partie des phénomènes hybrides [Genrich 98]. La partie continue est discrétisée et les variables discrétisées sont mises à jour lors du franchissement d'une transition toutes les périodes d'échantillonnage. Les valeurs des variables associées aux jetons sont donc modifiées (par un calcul algébrique) lorsque le jeton franchit une transition, c'est-à-dire à des instants discrets. Comme nous l'avons présenté dans le chapitre précédent, les travaux de G. Moncelet sont fondés sur l'utilisation des réseaux de Petri colorés exemple de réseau de Petri de haut niveau.

Pour avoir une bonne précision des résultats, il faut que la période d'échantillonnage soit la plus petite possible. C'est la solution proposée par [Brettschneider 96] et [Genrich 98]. Cette technique présente cependant deux inconvénients majeurs. Le premier est que l'on alourdit le modèle par l'introduction de transitions qui ne représentent aucun événement. Si on représente un système échantillonné, la présence de telles transitions est justifiée puisqu'elles correspondent à l'échantillonnage, sinon elles sèment la confusion. En effet, on mélange, en quelques sortes, des aspects liés à la représentation du système que l'on modélise avec des aspects liés à la façon d'implémenter le modèle. Le second inconvénient est lié à la durée de simulation. En effet, la simulation de tels modèles devient très lourde car l'intervalle d'échantillonnage doit être très petit pour avoir une bonne précision [Genrich 98].

Pour faire face à ces inconvénients, deux catégories d'approches ont été développées : les RdP hybrides et les RdP associés à des équations différentielles et algébriques. Contrairement au cas des RdP de haut niveau, avec ces deux approches les variables continues sont connues à chaque instant. Dans le premier cas, les variables continues sont représentées par le marquage des places continues du réseau de Petri hybride. Dans le second cas, un système d'équations différentielles et algébriques est intégré entre deux franchissements de transitions.

II.A.2 Modélisation avec les RdP hybrides

Sous un formalisme graphique unifié, les réseaux de Petri hybrides [Allam 98, David 86] permettent la représentation de la dynamique discrète par des places et des transitions ordinaires, et celle de la dynamique continue par des places (dites continues) dont le marquage est un nombre réel (et non plus un nombre entier) positif ou nul, et des transitions (continues) qui correspondent à des écoulements continus. On peut ainsi décrire des variables qui évoluent de façon continue et linéaire en fonction du temps. Une transition continue t_i est franchie avec une vitesse v(t). Cela signifie qu'entre t et t+dt une quantité v(t).dt de marque est enlevée de sa place amont et est ajoutée à la quantité de marque de sa place avale. Une interprétation intuitive du franchissement d'une transition continue peut être imagée par l'écoulement d'un sablier.

L'influence de la partie discrète sur la partie continue se fait par l'intermédiaire de boucles élémentaires reliant des places discrètes à des transitions continues. L'influence de la partie continue sur la partie discrète (franchissement de seuils par des variables continues) se fait par des boucles élémentaires reliant des places continues (représentant les variables testées) à des transitions discrètes.

Les RdP hybrides permettent une représentation homogène des aspects continus (flux continus) et discrets dans le même formalisme, ce qui leur permet de représenter facilement des systèmes de transfert de liquides ou des systèmes de production manufacturiers où le flux des pièces est approximé par un flux continu [Allam 98]. Cependant lorsqu'on introduit des contraintes algébriques, il faut rajouter des équations au modèle et effectuer une mise à jour régulière des variables. Nous perdons ainsi l'intérêt principal des RdP hybrides qui, de ce fait, ne peuvent représenter dans un unique formalisme l'ensemble de la partie discrète et de la partie continue. De plus, il est impossible de prendre en compte des équations différentielles générales.

En conclusion, l'avantage d'une modélisation basée sur le formalisme des RdP hybrides par rapport à celle basée sur les RdP de haut niveau est qu'elle ne nécessite aucune discrétisation des variables continues. Son inconvénient est qu'elle ne permet pas de prendre en compte des équations différentielles générales ni de représenter des équations algébriques en utilisant des places et des transitions d'un RdP hybride, comme c'est le cas avec les RdP couplés avec des équations différentielles. De plus, certains franchissements de seuils significatifs (un niveau de liquide dans une cuve qui atteint la valeur zéro par exemple) ne sont pas représentés par des franchissements de transition. Le réseau de Petri sous-jacent au réseau de Petri hybride ne représente donc pas une bonne vue discrète du système hybride.

II.A.3 Modélisation avec les RdP associés à des équations différentielles

L'idée d'associer à chaque partie du système un formalisme différent permet de séparer clairement la partie discrète de la partie continue. Cela permet aussi de conserver le potentiel de modélisation dans chacun des domaines. En effet, la complexité de la partie continue du modèle hybride est totalement découplée de celle de la partie discrète. Il n'est alors pas nécessaire de se restreindre à des classes prédéfinies d'équations différentielles comme c'est le cas pour les RdP hybrides.

Les réseaux de Petri associés à des équations différentielles regroupent deux types de réseaux de Petri : les réseaux de Petri mixtes [Zaytoon 01] et les réseaux de Petri Prédicats-Transitions Différentiels [Champagnat 98]. Nous détaillerons, ci-dessous, ces deux formalismes et nous les comparerons. Nous présenterons ensuite une extension du formalisme

choisi afin de prendre en compte les aspects stochastiques propres aux mécanismes de défaillances et de réparations des systèmes mécatroniques.

II.B Les RdP associés à des équations différentielles

Dans cette section, nous présenterons les deux modèles hybrides permettant de coupler un réseau de Petri et un ensemble d'équations différentielles et algébriques : les réseaux de Petri Prédicats-Transitions Différentiels (notés RdP PTD) et les réseaux de Petri Mixtes (notés RdP M). Nous comparerons ensuite ces deux formalismes afin de choisir le mieux adapté à la modélisation des systèmes hybrides complexes.

II.B.1 Les RdP Prédicats-Tansitions Différentiels

II.B.1.1 Principe général

Le comportement hybride est pris en compte en associant les variables représentant l'état continu à des jetons et en associant à chaque place un ensemble d'équations algébro-différentielles (algébriques et/ou différentielles), notées EAD. Un jeton mis dans une place déclenche l'intégration des équations correspondantes. Parallèlement à l'intégration, plusieurs seuils sont surveillés. Chaque seuil surveillé est associé à une transition avale d'une place marquée. Quand le seuil est franchi, cela signifie que l'événement attendu correspondant est apparu, et la transition associée à ce seuil est franchie. Un nouveau marquage est calculé et l'intégration du nouveau système démarre. Si deux jetons sont dans la même place, les variables de chacun d'eux évoluent, indépendamment les unes des autres, selon les mêmes équations associées à la place (cf. figure 2.1.).

Le modèle RdP représente les différentes configurations du système. Pour chaque configuration (état discret) à laquelle est rattaché un phénomène continu, un ensemble d'équations est intégré. Ces équations peuvent être de complexité variable (linéaire/non-linéaire, implicite/explicite), ce qui rendra la résolution plus ou moins complexe.

II.B.1.2 Définition

Ce modèle est défini, dans [Hochon 98], par le triplet <PI, T, M> où PI est l'ensemble des places du réseau (*card* (PI) = *l*, *l* étant le nombre de places), T l'ensemble des transitions (*card* (T) = *m*, *m* étant le nombre de transitions) et M le marquage. Les places de ce modèle sont décrites par la paire <PI, F> où F est l'ensemble des fonctions associées aux places. F est défini comme suit :

$$F = \begin{bmatrix} \\ F_1(\dot{X}_1, X_1, t) \\ \vdots \\ \\ F_l(\dot{X}_l, X_l, t) \end{bmatrix}$$

X_i étant l'ensemble des variables manipulées par les fonctions F_i associées à la place Pl_i. Nous avons :

$$F_i(\dot{X}_i, X_i, t) = \begin{bmatrix} \\ f_{i1}(\dot{X}_i, X_i, t) \\ \vdots \\ \\ f_{ik_i}(\dot{X}_i, X_i, t) \end{bmatrix}$$

Où $f_{i1}, ..., f_{ik_i}$ sont des équations représentant l'évolution des variables continues. Ainsi, chaque place du réseau (qui représente un état discret) peut décrire l'évolution de l'ensemble des variables continues X_i (grâce au système d'équations F_i).

Les transitions de ce réseau sont, quant à elles, décrites par le triplet <T, E, J>, où E est l'ensemble des fonctions de **sensibilisation** et J l'ensemble des fonctions de **jonction** associées aux transitions. Nous avons alors E qui est défini comme suit :

$$E = \begin{bmatrix} E_1 \\ \vdots \\ E_m \end{bmatrix}$$

Le seuil E_j associé à la transition T_j est défini comme étant la première solution de :

$$E_j(\dot{X}_j, X_j, t) = 0$$

X_j étant l'ensemble des variables continues utilisées par E_j (X_j est l'union des variables associées aux jetons nécessaires au franchissement de T_j). Cette fonction permet de détecter les événements d'état (une variable atteint une valeur prédéterminée), aussi bien qu'un événement de temps (t atteint une valeur prédéfinie). Les fonctions de sensibilisation sont activées lorsque leurs transitions correspondantes sont sensibilisées. Lors du franchissement d'une transition, les fonctions de jonction associées sont activées. Elles sont définies par :

$$J = \begin{bmatrix} J_1 \\ \vdots \\ J_m \end{bmatrix}$$

La fonction J_j associée à la transition T_j, lors du franchissement de T_j à la date t calcule les valeurs des variables ainsi que leurs dérivées.

$$J_j : \begin{cases} \dot{X}_j(t^+) = J_{j\dot{X}}(\dot{X}_j, X_j, t^-) \\ X_j(t^+) = J_{jX}(\dot{X}_j, X_j, t^-) \end{cases}$$

Ce qui signifie que les valeurs des variables, et de leurs dérivées, juste après t (à t^+) sont calculées à partir de valeurs des variables, et de leurs dérivées, juste avant t (à t^-). Ceci permet de redéfinir les attributs des jetons en cas de discontinuité.

Le dernier élément de ce modèle est M qui représente le marquage du réseau. Les variables du système se trouvent réparties dans l'ensemble des jetons du réseau. Lorsqu'une place comporte plusieurs jetons, chaque jeton instancie le système d'équations correspondant avec l'ensemble des variables qu'il transporte. Prenons par exemple le cas d'une place Pl_i contenant deux jetons A et B. Les équations suivantes seront actives, soit 2 m équations :

$$\begin{bmatrix} \dot{} \\ f_{i1}(\dot{X}_{iA}, X_{iA}, t) \\ \vdots \\ f_{im}(\dot{X}_{iA}, X_{iA}, t) \\ \dot{} \\ f_{i1}(\dot{X}_{iB}, X_{iB}, t) \\ \vdots \\ f_{im}(\dot{X}_{iB}, X_{iB}, t) \end{bmatrix}$$

II.B.1.3 Exemple

L'exemple, présenté à la figure 2.1, permet d'illustrer le principe des RdP PTD.

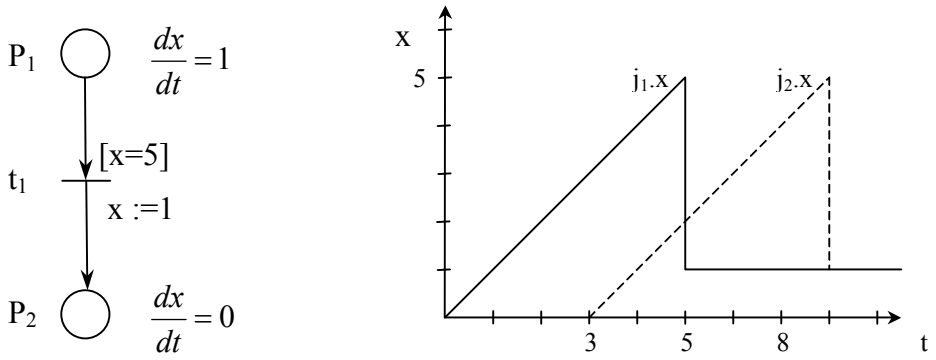


Figure 2.1. **Principe des réseaux de Petri Prédicats-Transitions Différentiels**

La partie droite de la figure montre l'évolution des variables associées à deux jetons A et B dans le réseau de Petri PTD de la partie gauche. Ce réseau est constitué de deux places P_1 et P_2 et d'une transition t_1. A chacune des places P_1 et P_2 est associée une équation différentielle. A la transition t_1 sont associées une fonction de sensibilisation (le seuil [x = 5]) et une fonction de jonction (x :=1). Le premier jeton est introduit dans la place P_1 à l'instant t= 0 et le second à l'instant t = 3. A l'instant t = 5, la transition t_1 devient franchissable pour le premier jeton qui la franchit en subissant le saut (x :=1) alors que le second jeton franchit cette même transition à l'instant t = 8.

Cet exemple montre qu'il est possible de définir autant d'instances d'un même comportement que nécessaire à partir d'un modèle générique, en l'occurrence ici l'évolution des variables associées aux deux jetons. Ce mécanisme d'instanciation est très important quant à la modularité de la modélisation des systèmes complexes [Champagnat 98a].

II.B.2 Les RdP Mixtes

Un réseau de Petri mixte [Zaytoon 01] initialement marqué est un quintuplet {R, PC, M_0, X_0, IH} où :

- R est un ensemble de réseaux de Petri interprétés [David 94] qui peuvent se synchroniser par des événements internes : chaque transition pourra être associée à

un couple d'événements, l'un déclenchant le franchissement et l'autre étant généré par celui-ci;
- PC est la partie continue du modèle. Elle comprend l'ensemble des variables continues et l'ensemble des équations algébro-différentielles qui modélisent les différentes dynamiques continues du système. Les variables sont globales et leur valeur est accessible partout dans le modèle à tout instant ;
- M_0 est le marquage initial ;
- X_0 est l'état initial des variables continues ;
- IH est l'interface hybride entre la partie continue du modèle et les réseaux de Petri.

Cette interface hybride IH est composée de quatre éléments EP, EM, Jump et Guard :

- EP définit pour chaque place, éventuellement vide, des équations qui sont actives lorsque la place est marquée ;
- EM définit pour chaque configuration de marquage, où au moins deux places sont marquées, un ensemble d'équations actives qui lui sont associées. Cet ensemble peut être vide et ne se substitue pas au précédent mais s'y ajoute, c'est-à-dire qu'à chaque instant l'ensemble des équations actives est la réunion des éléments de EP qui correspondent aux places marquées et des éléments de EM correspondant à la configuration de marquage validée ;
- Jump associe à chaque transition un vecteur dont chaque composante correspond à la fonction de saut, qui peut être l'identité, sur chacune des variables ;
- Guard associe à chaque transition la condition sur les variables continues qui doit être vérifiée pour la franchir.

Le nombre de marques de toute place lorsque celle-ci est associée à une équation différentielle est limité à 1. Cela permet de garantir la cohérence du modèle. Quant aux places n'ayant aucune équation, le nombre de marques qu'elles peuvent contenir n'est pas limité.

A chaque instant, l'état du réseau de Petri mixte est donné par la paire (M, X) correspondant à l'association d'un marquage et d'une valeur du vecteur d'état. L'état du système évolue selon la dynamique continue définie par les équations activées par le marquage, en considérant qu'une variable continue n'évolue pas lorsqu'il n'y a pas d'équation active définissant son évolution. La valeur de toutes les variables continues est donc connue à tout instant, quel que soit le marquage du réseau de Petri mixte. Cette évolution continue se prolonge jusqu'au premier instant pour lequel au moins une transition est validée.

II.B.3 Comparaison entre RdP PTD et RdP Mixtes

Les deux approches réseaux de Petri Prédicats-Transitions Différentiels et réseaux de Petri Mixtes partagent un même principe de base : l'évolution des variables continues est définie par un ensemble d'équations différentielles et algébriques. L'ensemble des équations est défini à partir du marquage du réseau de Petri et le franchissement de transitions dépend du dépassement de seuils sur les variables continues ou leurs dérivées. Soit les variables continues sont supposées globales (réseaux de Petri Mixtes), soit elles correspondent aux attributs des jetons (RDP PTD).

Dans les réseaux de Petri Mixtes, les équations peuvent être associées soit aux places, soit directement aux marquages, alors que dans les RdP PTD elles sont associées uniquement aux places car les variables continues ne sont pas globales mais associées localement aux jetons.

Les RdP Prédicats-Transitions Différentielles sont bien adaptés pour une approche modulaire et compositionnelle de représentation des systèmes complexes, ce qui permet une conception structurée du modèle. Par contre, cette approche n'est possible que lorsque la décomposition en modules de la partie discrète (ensemble de réseaux de Petri partiels) coïncide bien avec la décomposition de la partie continue (en sous-ensembles de variables continues et en sous-ensembles d'équations différentielles et algébriques). Quand ces décompositions ne coïncident pas, ceci implique la présence de variables continues partagées entre les modules. Dans ce cas, on parle d'interaction continue. Il faut donc spécifier explicitement que certaines variables sont partagées entre certains modules [Champagnat 98c]. On se rapproche alors des réseaux de Petri mixtes car on ne peut rien déduire sur la dynamique continue sans connaître entièrement le marquage.

Des travaux sont en cours [Villani 02] sur l'application des concepts orientés objets au RdP PTD. En effet, la construction et l'analyse d'un modèle de système complexe sous la forme d'un unique réseau global ne sont pas envisageables. Le modèle du système doit être construit comme un ensemble d'objets qui interagissent pour exécuter les diverses tâches du système. Pour cela, différents concepts ont été définis pour l'orientation objet et la décomposition de tels modèles.

En conclusion, les RdP PTD sont mieux adaptés pour la modélisation des systèmes complexes car ils permettent une approche modulaire. C'est le formalisme que nous retenons pour la modélisation des systèmes hybrides que sont les systèmes mécatroniques. Toutefois, pour étudier la sûreté de fonctionnement de ces systèmes, nous avons besoin de prendre en compte l'aspect stochastique. Cet aspect est lié aux mécanismes de défaillances et de reconfigurations inhérents aux systèmes mécatroniques. Pour cette raison, nous définissons un nouveau formalisme intégrant ces aspects non déterministes.

II.C Les réseaux de Petri Predicats-Transitions Différentiels et Stochastiques (RdP PTDS)

Afin de prendre en compte les aspects stochastiques propres aux mécanismes de défaillance et de reconfigurations des systèmes mécatroniques, nous définissons un nouveau formalisme appelé réseau de Petri Prédicats-Transitions Différentiels Stochastiques et noté RdP PTDS. Cette approche est similaire à celle introduite par E. Dubois et H. Alla dans [Dubois 93]. En effet, ils ont ajouté l'aspect stochastique aux réseaux de Petri hybrides dans le cadre de l'évaluation de performance des systèmes de production.

II.C.1 Principe général

Les RdP Prédicats-Transitions Différentiels Stochastiques sont une extension des RdP PTD définis dans le paragraphe précédent. Ce dernier modèle (les RdP PTD) associe à toute transition une fonction de sensibilisation déterministe : la transition est franchie exactement à la date à laquelle le seuil est atteint. Dans le cas des RdP PTDS, nous distinguons deux types de transitions : les transitions déterministes et les transitions stochastiques. Aux transitions déterministes sont associées des fonctions de sensibilisation et aux transitions stochastiques sont associées des **fonctions stochastiques**. Ces fonctions stochastiques sont associées à toute transition modélisant l'occurrence d'une défaillance d'un composant ou sa réparation si elle est non déterministe. Les lois propres à ces défaillances et réparations sont générales. Nous ne nous restreignons pas aux lois exponentielles (les plus connues dans le domaine de la SdF).

II.C.2 Définition

Ce modèle est défini par le triplet <PI, T, M> où PI est l'ensemble des places du réseau, T l'ensemble des transitions et M le marquage. Les places de ce modèle sont décrites, comme pour le formalisme des RdP PTD, par la paire <PI, F> où F est l'ensemble des fonctions associées aux places. Quant aux transitions de ce réseau, elles sont réparties en deux sous-ensembles disjoints : T_s et T_d tel que $T = T_s \cup T_d$ et $T_s \cap T_d = \varnothing$. T_s est l'ensemble des transitions stochastiques. T_d est l'ensemble des transitions déterministes (par opposition à stochastiques).

L'ensemble des transitions déterministes est caractérisé par le triplet <T_d, E, J> défini dans le paragraphe précédent sur les RdP PTD. Quant aux transitions stochastiques, elles sont décrites par le triplet <T_s, F_s, J>. Fs et J représentent respectivement l'ensemble des fonctions stochastiques et l'ensemble des fonctions de jonction associées aux transitions stochastiques. Les fonctions de jonction permettent une représentation simple des défaillances qui se traduisent par l'affectation de certaines valeurs à certaines variables continues. Quant aux fonctions stochastiques, elles sont de deux types : des fonctions stochastiques temporisées ou immédiates. Les fonctions stochastiques temporisées sont décrites par le couple <I, D> où I est l'ensemble des intervalles de tirs associés aux transitions stochastiques temporisées et D est l'ensemble des densités de probabilités associées à ces mêmes transitions. A une transition stochastique temporisées t_k, nous associons la fonction densité de probabilité D (t_k) = $d_k(x)$ définie sur un intervalle de tir statique I (t_k) = [a, b] (intervalle de réels avec $b \in \Re \cup \{+\infty\}$) avec $\int_a^b d_k(x)dx = 1$. Il n'y a pas d'effet mémoire, c'est-à-dire que chaque fois que t_k est sensibilisée par un nouveau n-uplet de jetons, un nouvel intervalle I (t_k) est associé au franchissement de t_k par ce n-uplet. Cet intervalle est effacé si t_k cesse d'être sensibilisée par ce n-uplet, par exemple si l'un des jetons de ce dernier est consommé par un franchissement de transition.

En ce qui concerne les fonctions stochastiques immédiates, elles sont telles qu'une probabilité fixe est associée à chaque transition concernée. Ces transitions sont utiles pour la prise en compte des défaillances à la sollicitation et sont immédiates. De manière générale, un composant qui peut être défaillant à la sollicitation est représenté par une place possédant deux transitions en sortie. La place représente l'état « attente » du composant. A ces deux transitions sont associées des probabilités p_1 (probabilité de panne à la sollicitation) et p_2 (probabilité de bon fonctionnement après sollicitation) et tel que $p_1 + p_2 = 1$.

II.C.3 Exemple

Considérons l'exemple de la figure 2.2. La partie droite de la figure illustre l'évolution des variables continues associées à deux jetons j_1 et j_2 dans un RdP PTDS (partie gauche de la figure). Ce réseau contient trois places P_1, P_2 et P_3 et deux transitions t_1 et t_2. Aux places sont associées trois équations différentielles. La transition t_1 est déterministe. Une fonction de sensibilisation (le seuil [x = 5]) et une fonction de jonction (x := 1) lui sont associées. Une variable aléatoire uniformément répartie entre les instants t = 2 et t = 6 est associée à la transition t_2 qui est stochastique. Cette variable aléatoire représente l'instant de tir de la transition t_2. Le jeton j_1 est introduit dans la place P_1 à l'instant t = 0 et le second à l'instant t = 3. A l'arrivée des jetons dans P_1 les variables x valent zéro (j1.x = 0 et j2.x = 0).

La transition t_1 est sensibilisée mais ne peut être tirée avant que la variable j_1.x n'atteigne le seuil défini par la fonction de sensibilisation. La transition t_2 est elle aussi sensibilisée mais ne peut être franchie qu'entre t = 2 et t = 6. Une loi uniforme U[2,6] est attribuée à t_2. Une

réalisation de la variable aléatoire associée à cette transition correspond à t = 4. A cet instant, t_1 n'a pas encore été franchie car $j_1.x < 5$. La transition t_2 est franchie et le jeton j_1 est introduit dans la place P_3 avec la valeur $j_1.x = 4$. L'évolution de cette variable est maintenant régie par l'équation associée à P_3.

Une évolution possible de la variable associée au jeton j_2 est celle donnée par la courbe de $j_2.x$. Le jeton j_2 étant mis dans la place P_1 à t = 3, la transition t_2 ne peut être tirée qu'entre les instants t = 2+3 = 5 et t = 6+3 = 9 (en supposant que, cette fois, la réalisation de la variable aléatoire soit 6, ce qui donnerait une date de franchissement de 9). L'évolution de la variable $j_2.x$ est alors identique à celle dans la figure 2.1 car la transition t_2 n'a pu être tirée avant t_1.

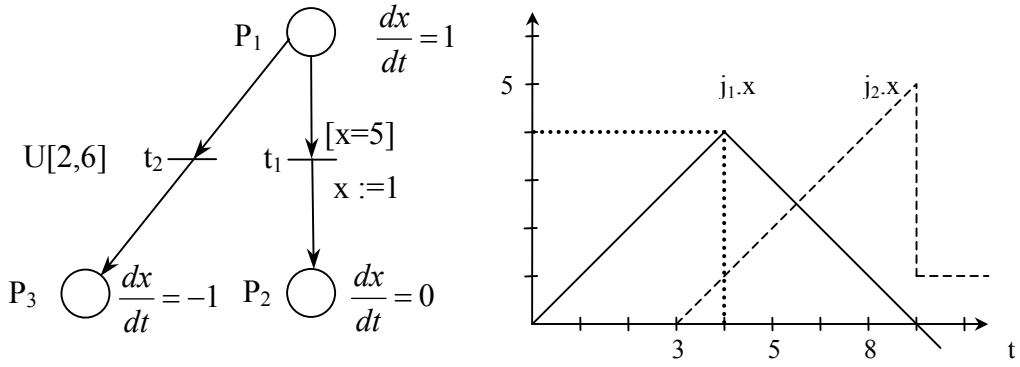


Figure 2.2. **Principe des réseaux Prédicats-Transitions Différentiels Stochastiques**

II.D Discussion sur le modèle hybride

Comme cela a été mentionné auparavant, l'avantage des réseaux de Petri Prédicats Transitions Différentiels par rapport à d'autres modèles hybrides est la bonne délimitation entre la partie discrète et la partie continue. Si l'on considère le réseau de Petri ordinaire sous-jacent au réseau de Petri Prédicats-Transitions Différentiels modélisant le système étudié, ce réseau de Petri doit être cohérent et montrer de façon exhaustive tous les changements de configurations possibles. Il en sera de même avec les réseaux de Petri Prédicats-Transitions Différentiels Stochastiques. C'est pourquoi la recherche des relations de cause à effet peut se baser sur la seule partie discrète. Mais pour cela, il nous faut avoir une vision logique de cette partie discrète. C'est le but de la section suivante.

III Aspect logique et accessibilité

III.A Introduction

Les études de sécurité des systèmes mécatroniques commencent par une analyse qualitative. Cette analyse vise à mettre en évidence tous les types de comportement amenant à des états pour lesquels la sécurité des automobilistes n'est plus assurée. Elle a pour but de déterminer les scénarios redoutés et de les caractériser en terme d'actions et de changements d'états. Comme on l'a vu dans l'état de l'art, une façon de faire est de générer, dans un premier temps, le graphe d'accessibilité à partir d'un modèle qualitatif construit à l'aide des réseaux de Petri et prenant en compte l'aspect continu. Ensuite, l'exploration de ce graphe permettrait de trouver l'ensemble des chemins menant vers l'état redouté en question. Cette approche a, en particulier, été utlisée par [Moncelet 98]. Comme nous l'avons montré dans le chapitre précédent, outre le problème de l'explosion combinatoire du nombre d'états due à la complexité des systèmes mécatroniques, les scénarios obtenus ne sont pas minimaux car ils sont entrelacés avec des événements qui n'ont aucune relation de cause à effet avec l'atteignabilité de l'état partiel redouté. En effet, tous les événements se produisant en parallèle avec le scénario redouté, par exemple dans des éléments du système non concernés par l'état partiel redouté, vont apparaître dans le scénario obtenu. Pourtant, seuls les scénarios minimaux contribuent à une bonne compréhension des conséquences qualitatives des choix de conception. De plus l'évaluation qualitative de la sûreté de fonctionnement doit, elle aussi, reposer sur des scénarios minimaux si l'on veut être efficace.

Un moyen de contourner le problème de l'explosion combinatoire est d'utiliser directement le modèle réseau de Petri pour établir les scénarios possibles, plutôt que d'utiliser le graphe d'accessibilité associé, qui contient bon nombre d'informations sans intérêt du point de vue sûreté de fonctionnement car décrivant des comportements sans défaillance. La logique Linéaire offre un cadre théorique permettant d'interpréter les modèles réseaux de Petri et d'en extraire des scénarios. Le point clé de cette approche est l'équivalence entre l'accessibilité dans le réseau de Petri et la prouvabilité du séquent associé en logique Linéaire. Seule la logique permet d'analyser les relations de cause à effet.

III.B Logique Linéaire

III.B.1 Introduction

La logique Linéaire a été proposée par J.Y. Girard [Girard 87] comme une restriction de la logique propositionnelle classique pour prendre en compte la notion de ***ressource***. Cela veut dire que les propositions logiques sont considérées, non plus comme des vérités pérennes, mais comme des ressources qui sont consommées et produites pendant les preuves.

Une déduction en logique Linéaire ***consomme*** les propositions qu'elle prend pour prémisses, et ***produit*** les propositions qui forment sa conclusion. Cela signifie que pour réutiliser une prémisse qui vient d'être consommée, soit il aurait fallu initialement poser deux fois cette prémisse, soit il faut préalablement la produire à nouveau, par le biais d'une autre déduction.

Prenons un exemple afin d'illustrer la différence fondamentale entre la logique classique et la logique Linéaire : le principe de consommation / production. Considérons le raisonnement suivant :

	pour un Euro j'ai un pain	*[A ⇒ B]*
(or)	*j'ai un Euro*	*[A]*
(donc)	*j'ai un pain*	*[B]*

Ce raisonnement ne serait pas réaliste s'il était formulé en logique classique. En effet, les valeurs de vérité des propositions classiques ne sont pas révocables. Cela pourrait nous mener à la conclusion suivante : l'achat d'un pain est gratuit puisqu'on aura toujours, après achat, un Euro.

Afin de pouvoir prendre en compte ce principe de consommation / production, Girard a été amené à refaire intégralement le système des connecteurs classiques et à introduire quatre nouveaux connecteurs au lieu de deux (le « et » et le « ou »). Une des principales raisons de cette refonte est que l'idée de conjonction de ressources ne vérifie pas une propriété fondamentale de la conjonction classique : ***l'idempotence***. Cette propriété est le fait que toute proposition *A* est équivalente à sa conjonction avec elle-même *A et A*. En conséquence, certaines lois de base de la logique classique ont été éliminées : la contraction et l'affaiblissement. La règle de contraction conduit à l'idempotence du « et » puisqu'il est possible de dériver *A et A* à partir de *A*. La seconde règle qui est abandonnée est celle dite d'affaiblissement : elle permet notamment de déduire la proposition A à partir de A et B. Elle permet donc de traiter des causes sans effets, c'est-à-dire d'oublier des hypothèses dans un raisonnement, d'où son rejet.

III.B.2 Quelques connecteurs

L'ensemble des connecteurs de la logique Linéaire est réparti en 3 groupes : les connecteurs multiplicatifs, additifs et exponentiels. Nous ne présenterons ici que ceux qui concernent notre travail sur les réseaux de Petri. Ces connecteurs font partie du fragment MILL (Multiplicative Intuitionnistic Linear Logic) de la logique Linéaire. Ce fragment comprend le connecteur multiplicatif ***FOIS*** et le connecteur ***implication linéaire***. Il n'y a pas de négation et le meta-connecteur « **,** » est commutatif. Le lecteur intéressé pourra trouver une présentation détaillée des autres connecteurs dans [Girault 98].

Le connecteur **Fois**, noté $\otimes$, est la conjonction multiplicative. Il correspond au connecteur « et » de la logique classique auquel on a enlevé la propriété d'idempotence. Ce connecteur exprime l'accumulation de ressources : ainsi la proposition $A \otimes A$ représente la présence de deux exemplaires de *A*. Cette proposition n'est pas équivalente à *A*.

Le connecteur ***implication linéaire***, notée $\multimap$, exprime la **causalité** entre la production et la consommation de ressources. Par exemple, $A \multimap B$ traduit le fait qu'en consommant la proposition *A*, la proposition *B* est produite. C'est donc le résultat d'un changement d'état qui est ainsi modélisé.

Considérons deux exemples de déduction utilisant ces deux connecteurs. A partir de la proposition $A \otimes A$ et de $A \multimap B$ il est possible de déduire $A \otimes B$ mais pas B : le second exemplaire de *A* ne peut être « oublié ». Partant de *A* et de $A \otimes A \multimap B$ il n'est pas possible de déduire *B* car un seul exemplaire de *A* n'est pas suffisant pour la déduction. Dans les deux cas, l'usage des connecteurs classiques (« et » et « *implique* ») aurait permis ces déductions.

III.B.3 Calcul des séquents et arbres de preuve

Toutes les déductions de la logique Linéaire se font dans un cadre clair et formel : le calcul des séquents. Introduit par Gentzen en 1934, le calcul des séquents est un formalisme

syntaxique permettant d'étudier les lois de la logique. Girard a montré qu'il était parfaitement adapté pour définir la logique Linéaire et l'a repris pour en définir la syntaxe.

Un **séquent** est une formule de la forme : $\Gamma, \Gamma' \vdash \Delta, \Delta'$ où le symbole $\vdash$, dit le tourniquet, est un méta-symbole qui sépare la partie gauche (ici Γ, Γ') de la partie droite (ici Δ, Δ'). Ces deux parties sont constituées de suites finies de formules. La virgule est aussi un méta-symbole dont le sens est fonction de sa position par rapport au tourniquet : la conjonction dans la partie gauche et la disjonction dans la partie droite. Le séquent peut se lire de la façon suivante : la conjonction « Γ *et* Γ' » permet de déduire la disjonction « Δ *ou* Δ' ». Dans le cadre de la logique Linéaire Intuitionniste, et a fortiori dans le fragment MILL, le membre droit sera toujours réduit à une seule formule (qui peut contenir plusieurs atomes). Reprenons l'exemple du raisonnement concernant l'achat d'un pain. Le séquent s'écrit : $A, A \multimap B \vdash B$ et représente la possibilité d'acheter un pain à condition de disposer des ressources nécessaires : un Euro et une boulangerie.

Prouver un séquent consiste à montrer qu'il est entièrement construit à partir d'un ensemble de règles introduisant les atomes (propositions) et les connecteurs. Les règles représentées ci-dessous sont celles du fragment multiplicatif MILL. Ces règles sont réparties en 3 groupes : le groupe identité, le groupe structurel et le groupe logique. Le groupe identité et le groupe structurel expriment des propriétés intrinsèques à la logique tandis que le groupe logique définit une règle d'introduction à gauche et une règle d'introduction à droite pour chacun des connecteurs spécifiques à la logique considérée (à savoir ici $\otimes$ et $\multimap$). Le meta-symbole « , » et le connecteur $\otimes$ sont commutatifs.

Groupe Identité

$$\frac{}{A \vdash A}\,Identité \qquad\qquad \frac{\Gamma \vdash F \quad \Delta, F \vdash H}{\Gamma, \Delta \vdash H}\,Cut$$

Groupe Structurel

$$\frac{\Gamma, F, G, \Delta \vdash H}{\Gamma, G, F, \Delta \vdash H}\,Echange$$

Groupe Logique

$$\frac{\Gamma \vdash F \quad \Delta, G \vdash H}{\Gamma, \Delta, F \multimap G \vdash H} \multimap_L \qquad \frac{\Gamma, \Delta, F \vdash G}{\Gamma, \Delta \vdash F \multimap G} \multimap_R$$

$$\frac{\Gamma, F, G \vdash H}{\Gamma, F \otimes G \vdash H} \otimes_L \qquad \frac{\Gamma \vdash F \quad \Delta \vdash G}{\Gamma, \Delta \vdash F \otimes G} \otimes_R$$

Figure 2.3. **Règles du calcul des séquents du fragment MILL**

A est un atome. F, G et H sont des formules, pas nécessairement atomiques. Γ et Δ sont des blocs de formules séparées par des virgules.

Pour chaque règle, le séquent à prouver est écrit en dessous de la barre tandis que le (ou les) séquent(s) utilisé(s) pour cette preuve est (sont) écrit(s) au dessus. L'attribut indique que l'on applique la règle à gauche (indice L comme Left) ou à droite (indice R comme Right) du

tourniquet du séquent concerné. S'il n'y a pas d'attribut, cela signifie que la règle s'applique à l'ensemble du séquent.

$$\frac{Séquent\ \text{Pr}émisse\ 1 \quad Séquent\ \text{Pr}émisse\ 2}{Séquent\ Conclusion}\ Nom\ Règle_{attribut}$$

Partant de ces règles, une preuve est conduite de bas en haut en posant comme racine de l'arbre de preuve le séquent à démontrer. On applique ensuite une à une les règles logiques pour éliminer successivement les connecteurs du séquent. Chaque nœud de l'arborescence est alors un séquent prémisse plus simple que celui dont il est conclusion. Ce séquent est prouvé si chaque feuille de l'arbre se termine par un séquent axiome (dans le sous-ensemble MILL, le seul axiome est le séquent identité). On dit qu'un séquent est prouvable si et seulement si il existe une preuve dont il est la racine. Par exemple, la démonstration du séquent $A, A \multimap B \vdash B$ est donnée par l'arbre de preuve suivant :

$$\frac{\dfrac{}{A \vdash A}\,Id \qquad \dfrac{}{B \vdash B}\,Id}{A, A \multimap B \vdash B} \multimap L$$

Figure 2.4. **L'arbre de preuve de** $A, A \multimap B \vdash B$

III.B.4 Elimination de la règle de coupure

Lue de bas en haut, la règle de coupure (notée *Cut*) est la seule règle qui introduit dans les prémisses du séquent à prouver une formule exogène (la formule F dans la règle de la figure 2.3). Ceci est similaire à l'introduction d'un lemme intermédiaire lors de la preuve d'un théorème. La logique Linéaire, comme la logique classique, vérifie le théorème de la redondance de la règle de coupure [Girard 87]. En effet, quand un séquent est prouvable, il existe une démonstration qui n'utilise pas cette règle. L'usage de la règle de coupure complique le mécanisme de preuve. C'est pour cela qu'on adoptera par la suite des preuves sans l'usage de cette règle.

III.C Traduction des réseaux de Petri en logique Linéaire

Comme nous venons de le voir, l'une des bases de la logique Linéaire est sa capacité à raisonner sur les ressources. Les réseaux de Petri sont également un outil qui manipule des ressources. Du fait de cette similitude, plusieurs travaux ont traité du lien entre ces deux formalismes. Nous ne présenterons, par la suite, que ceux qui traitent de la logique Linéaire comme un outil permettant d'interpréter les modèles RdP et d'en extraire les relations d'ordre.

III.C.1 Raisonnement logique sur les réseaux de Petri

Plusieurs auteurs se sont intéressés à la logique Linéaire comme un outil de raisonnement sur des modèles obtenus grâce aux réseaux de Petri. Ces travaux peuvent être classés selon deux tendances : celle qui ne prend pas en compte la notion de marquage mais qui permet d'expliciter les relations d'ordre structurelles dans un réseau de Petri, approche dite « sans marquage », et celle dite « avec marquage ». Comme son nom l'indique, cette dernière consiste à prendre en compte de manière explicite la notion de marquage. Elle permet, de ce fait, une meilleure caractérisation des relations d'ordre en tenant compte l'influence du marquage sur les relations d'ordre structurelles dans le réseau de Petri.

III.C.1.1 Approche sans marquage

V. Gehlot [Gehlot 92] a été le premier à avoir l'idée d'utiliser la logique Linéaire pour extraire les relations d'ordre partiel dans les réseaux de Petri. Toutefois, sa traduction des réseaux de Petri en logique Linéaire était structurelle et ne prenait pas en compte explicitement les marquages. D'autre part, cette approche nécessitait d'ajouter une liste d'axiomes propres, représentant les transitions, au calcul des séquents.

Le principe de cette approche est le suivant :

- un atome propositionnel P est associé à toute place p du réseau,
- un séquent axiome est défini pour chaque transition t du réseau de Petri, à partir des vecteurs Pre(t) et Post(t), et rajouté à la liste des axiomes du calcul des séquents.

$$t : \bigotimes_{i \in Pre(pi,t)} P_i \vdash \bigotimes_{o \in Post(po,t)} P_o$$

La preuve d'un séquent se termine lorsque toutes les feuilles conduisent à un axiome qui est, soit l'axiome *Identité*, soit l'un des axiomes générés par le réseau de Petri.

Cette approche permet de déterminer les relations d'ordre structurelles dans un réseau de Petri. Toutefois, elle présente deux inconvénients. Tout d'abord, les preuves ne se font pas dans le cadre strict de la logique Linéaire. En effet, elle est systématiquement augmentée par autant d'axiomes que de transitions dans le réseau. Cette extension des axiomes remet en cause la redondance de la règle de coupure et conduit à une preuve qui ne peut plus se faire sans utiliser cette règle, et cela complique le mécanisme de preuve. Le second inconvénient de cette approche est la non prise en compte de la notion de marquage. Cette approche ne permet, en effet, de caractériser que le parallélisme structurel dans un réseau de Petri. Or, le parallélisme dynamique (induit par le marquage) modifie les relations de parallélisme structurel. Le marquage influence les relations de causalité entre les franchissements des transitions et change les relations d'ordre partiel dans le réseau de Petri. C'est ce qui a motivé les travaux sur une autre traduction du fonctionnement des réseaux de Petri basée sur une représentation des instances de franchissement des transitions et une prise en compte explicite du marquage, c'est l'approche dite « avec marquage ».

III.C.1.2 Approche avec marquage

Afin de permettre une meilleure caractérisation des relations d'ordre dans un réseau de Petri, des travaux [Pradin 99 a et b, Girault 97, Kunzle 97] ont été menés et ont abouti à une nouvelle approche, dite avec marquage. Cette approche permet de prendre en compte le parallélisme dynamique d'un réseau de Petri. C'est celle que nous avons retenue pour notre travail et qui sera détaillée par la suite.

Dans le cadre de cette approche, les transitions d'un réseau de Petri ne sont plus représentées par un séquent mais par une proposition implicative. Nous représentons ainsi des instances de franchissement de transitions et non plus des transitions, et les propositions implicatives pourront être consommées au cours de la preuve, ce qui indiquera que la transition est effectivement franchie. Si la proposition implicative représentant la transition est consommée deux fois au cours de la preuve, cela signifie que cette transition a été franchie deux fois. En ce qui concerne le marquage, il est explicitement pris en compte.

Cette approche reste dans le strict cadre de la logique Linéaire puisqu'il n'y a pas d'axiomes propres rajoutés. Par conséquent, les preuves sont conduites sans utiliser la règle de coupure.

III.C.2 Traduire un réseau de Petri en logique Linéaire : approche avec marquage

La traduction d'un réseau de Petri en logique Linéaire, selon l'approche avec marquage, a été présentée dans [Pradin 99a]. Cette approche diffère du travail effectué dans [Gehlot 92] par l'introduction explicite de la notion de marquage. Une formule logique est associée à chaque marquage et à chaque transition.

Un marquage correspond à la présence simultanée de jetons dans un ensemble de places. A chaque place correspond une proposition atomique. Un marquage est alors décrit par une formule conjonctive (connecteur $\otimes$) de propositions atomiques. Ainsi, un marquage constitué d'un jeton dans la place A et d'un jeton dans la place B se traduira par la formule suivante : $A \otimes B$. Alors que le marquage constitué d'un jeton dans la place A et aucun jeton dans la place B se traduira par la formule A. Par contre, un marquage constitué de deux jetons dans la place A et d'un jeton dans la place B se traduira par : $A \otimes A \otimes B$.

Une transition exprime une **relation de causalité** (dite aussi une **relation de cause à effet**) entre deux formules de marquage. En effet, le marquage final est causalement lié au marquage initial car, quand ce dernier est consommé, ceci produit le marquage final. Une transition est traduite par une formule implicative (connecteur $\multimap$). Le côté gauche de la formule établit le marquage minimal pour franchir la transition, tandis que le côté droit représente le marquage atteint après le franchissement de cette transition à partir du marquage minimal.

Pour un réseau de Petri donné, cette traduction peut être formalisée de la façon suivante :

- un atome propositionnel P est associé à toute place p du réseau,
- un monôme en $\otimes$ (FOIS : la conjonction multiplicative) est associé à tout marquage ainsi qu'à toute précondition (Pre) ou postcondition (Post) de transition,
- une formule implicative est définie pour chaque transition t du réseau de Petri :

$$t : \bigotimes_{i \in Pre(pi,t)} P_i \multimap \bigotimes_{o \in Post(po,t)} P_o$$

Le franchissement d'une transition est représenté par un séquent valide. A partir d'une formule décrivant un marquage et d'une formule décrivant la transition à franchir (les prémisses du séquent), on obtient le marquage produit (la conclusion).

L'accessibilité entre deux marquages M_0 et M_f est représentée par un séquent. Il est nécessaire que la partie gauche de ce séquent contienne la liste de toutes les transitions permettant d'atteindre le marquage M_f à partir du marquage M_0. Cette partie du séquent doit aussi contenir la formule représentant le marquage initial. Quant à la partie droite du séquent, elle contient la formule représentant le marquage final. Le séquent exprimant l'accessibilité entre les marquages M_0 et M_f s'écrit sous la forme $M_0, t_1, ..., t_n \vdash M_f$, et spécifie quelles sont les transitions franchies (t_i représente la formule implicative correspondant à la transition t_i et doit être répétée autant de fois qu'elle est franchie pour atteindre M_f).

III.C.3 Equivalence entre accessibilité et prouvabilité

[Girault 97] a montré qu'il y a équivalence entre la prouvabilité de certains séquents du fragment MILL de la logique Linéaire et l'accessibilité dans un réseau de Petri.

Afin d'illustrer cette équivalence entre accessibilité et prouvabilité, considérons un réseau de Petri muni d'un marquage initial M_0 et d'un marquage final M_f. Soit $\bar{s}$ une liste non

ordonnée de transitions ($\bar{s}$ est le vecteur caractéristique, solution de l'équation fondamentale $M_f = M_0 + C.\bar{s}$). Il y a équivalence entre le fait de prouver le séquent $M_0, \bar{s} \vdash M_f$ et celui de trouver une séquence σ de franchissements de transitions menant de M_0 à M_f dans le réseau de Petri telle que les transitions de σ soient celles de $\bar{s}$ avec la même arité (on franchit autant de fois la transition t_i qu'elle se trouve dans la liste $\bar{s}$).

Puisqu'il y a équivalence entre l'accessibilité dans un réseau de Petri et la prouvabilité de certains séquents en logique Linéaire, nous allons nous intéresser à comment conduire la preuve du séquent exprimant cette accessibilité au travers de la construction d'un arbre de preuve canonique.

III.D Construction de l'arbre de preuve canonique

Comme nous l'avons vu en III.C.1, nous disposons en logique Linéaire de tout un ensemble de règles pour faire une preuve. Un séquent donné peut, en conséquence, être prouvé de diverses manières. Nous avons choisi une stratégie de preuve qui n'utilise pas la règle de coupure (elle est redondante). Ceci donne naissance à un arbre de preuve dit canonique.

La procédure de construction de l'arbre de preuve canonique est relativement simple car elle ne nécessite ni la règle de coupure ni la règle d'introduction à droite de l'implication Linéaire : la taille de l'arbre de preuve est, par conséquent, strictement proportionnelle au nombre de franchissements inclus dans le séquent. Cette procédure utilise principalement les règles d'introduction à gauche de l'implication Linéaire ($\multimap_L$) et du connecteur $\otimes$ ($\otimes_L$) que nous allons présenter ci-dessous.

III.D.1 Application de la règle $\otimes_L$

Lue de bas en haut, la règle $\otimes_L$ transforme le connecteur $\otimes$, dit *Fois*, (figurant dans la partie gauche du séquent prémisse) en une virgule ($\dfrac{\Gamma, F, G \vdash H}{\Gamma, F \otimes G \vdash H} \otimes_L$). L'intérêt d'appliquer cette règle est de rendre les deux formules concernées (ici $F \otimes G$) indépendantes (F, G). Cela veut dire qu'elles peuvent être consommées indépendamment l'une de l'autre. En effet, la proposition $F \otimes G$ signifie la disponibilité simultanée des formules F et G et aussi le fait qu'elles constituent une seule entité indivisible. La non utilisation de cette règle empêche toute autre règle d'opérer sur une seule des deux formules (les deux doivent être consommées à la fois). En d'autres termes, l'application de cette règle permet de diviser l'entité $F \otimes G$ en deux sous-entités indépendantes F et G.

Dans un contexte de preuve d'accessibilité dans un réseau de Petri, cette formule est utilisée pour transformer, dans le membre gauche du séquent à prouver, une formule représentant un marquage M ($M \equiv P_i \otimes P_j \otimes ... \otimes P_k$) en un ensemble d'atomes (ou de jetons) indépendants ($P_i, P_j, ..., P_k$). Cet ensemble d'atomes est appelé ***étape courante***. Cette notion, très importante pour notre travail, est plus générale que la notion de marquage puisqu'elle n'impose pas la présence simultanée de tous les jetons dans les places correspondantes. Ces jetons peuvent être consommés ou produits indépendamment les uns des autres.

III.D.2 Application de la règle $\multimap_L$

Cette règle opère sur le membre gauche d'un séquent conclusion ($\Gamma, \Gamma', F \multimap G \vdash H$) et s'applique de la manière suivante : $\dfrac{\Gamma \vdash F \quad \Gamma', G \vdash H}{\Gamma, \Gamma', F \multimap G \vdash H} \multimap_L$. Elle génère deux séquents prémisses ($\Gamma \vdash F$ et $\Gamma', G \vdash H$) et permet d'éliminer la formule implicative $F \multimap G$.

Dans le cadre de l'interprétation d'un réseau de Petri à l'aide de la logique Linéaire, l'application de cette règle, au cours d'un calcul de séquents, correspond à un franchissement particulier d'une transition du réseau. Considérons le cas où F et G sont des formules atomiques et $\Gamma = F$. Le séquent précédent s'écrit : $\dfrac{F \vdash F \quad \Gamma', G \vdash H}{F, \Gamma', F \multimap G \vdash H} \multimap_L$. La preuve du séquent conclusion gauche s'arrête puisqu'on aboutit à l'axiome *Identité* ($\dfrac{}{F \vdash F}\ Identité$). Nous pouvons constater que l'application de cette règle dans ce cas particulier consiste à consommer l'atome F et la formule implicative $F \multimap G$ puisqu'ils disparaissent du séquent prémisse droit ($\Gamma', G \vdash H$). L'application de cette règle correspond donc bien au franchissement effectif de la transition ayant pour place d'entrée F et pour place de sortie G (représentée par la formule implicative $F \multimap G$). La consommation du jeton dans la place F est représentée par le séquent $F \vdash F$ et la production d'un jeton dans la place G est représentée par l'apparition de l'atome G dans le séquent prémisse $\Gamma', G \vdash H$.

III.D.3 Algorithme de construction

Pour montrer l'accessibilité entre deux marquages M_0 et M_f, la preuve du séquent $M_0, t_1, ..., t_n \vdash M_f$ est conduite de la manière suivante : on commence par remplacer le marquage initial M_0 par une liste d'atomes indépendants en appliquant la règle $\otimes_L$ autant de fois que cela est nécessaire. Il est possible ensuite, en appliquant la règle d'introduction à gauche de l'implication Linéaire $\multimap_L$, d'extraire les relations de causalité des atomes en allant de M_0 vers M_f. A chaque fois que l'on applique la règle $\multimap_L$, on applique ensuite si nécessaire la règle $\otimes_L$ afin de séparer les atomes liés par le connecteur $\otimes$ et d'obtenir par conséquent une nouvelle étape courante. La preuve se poursuit essentiellement sur le côté droit de l'arbre car à chaque application de la règle $\multimap_L$ le membre gauche est prouvé par utilisation si nécessaire de la règle $\otimes_R$. La preuve du séquent se termine quand toutes les formules implicatives (représentant les transitions) ont été éliminées.

Revenons sur la notion d'***étape courante***, notion capitale pour notre travail. Nous appelons ***étape courante*** la liste de tous les atomes séparés par des virgules (méta-connecteurs), figurant dans le membre gauche du séquent à prouver. Un marquage est une étape courante tandis qu'une étape courante ne correspond pas nécessairement à un marquage effectivement atteint. Elle correspond à un marquage pouvant être atteint par une stratégie particulière de franchissements. L'existence d'une étape courante garantit que chaque jeton sera produit dans la place correspondante, et cela, indépendamment des autres jetons de la même étape courante.

A titre de remarque, le marquage M_f ne peut pas être décomposé car c'est le connecteur $\wp$ (PAR) qui correspond à la virgule à droite [Khalfaoui 01b].

Partant d'un séquent donné, l'arbre de preuve canonique est obtenu par l'algorithme suivant (extrait de [Pradin 03]) :

APPLIQUER la règle $\otimes_L$ autant de fois que nécessaire pour transformer le marquage de départ en une liste d'atomes séparés par le meta-connecteur « , »

TANT QUE la règle $\multimap_L$ est applicable (*c'est-à-dire si le marquage d'entrée d'une ou plusieurs formules de transitions est inclus dans la liste des atomes de l'étape courante*)

- Appliquer la règle $\multimap_L$ à la formule implicative candidate de plus petit ordre lexicographique,
- Terminer la preuve du séquent gauche généré en utilisant, si nécessaire, la règle $\otimes_R$,
- Appliquer, si nécessaire, la règle $\otimes_L$ au marquage produit dans le séquent droit (partie gauche de celui-ci).

FIN TANT QUE

Cet algorithme conduit à une preuve du séquent de départ si toutes les formules implicatives correspondant aux franchissements de transitions ont été éliminées et donc si toutes les feuilles de l'arbre se terminent par le séquent *Identité*.

III.D.4 Exemple illustratif

Afin de bien illustrer cette méthode de traduction des réseaux de Petri et de preuve de séquents en logique Linéaire selon l'approche avec marquage, nous l'avons appliquée à titre d'exemple sur le réseau de Petri suivant :

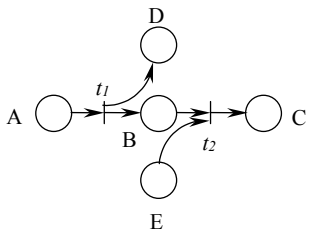


Figure 2.5. **Exemple de réseau de Petri**

Nous nous posons le problème de l'accessibilité entre deux marquages M_0 et M_f en franchissant une fois chacune des transitions t_1 et t_2. Le marquage initial M_0 est constitué d'un jeton dans la place A et d'un jeton dans la place E. Le marquage final M_f est constitué d'un jeton dans la place C et d'un jeton dans la place D.

Les transitions du réseau de Petri se traduisent de la façon suivante :

- $t_1 \equiv A \multimap B \otimes D$,
- $t_2 \equiv B \otimes E \multimap C$.

Les marquages initial et final se traduisent par :

- $M_0 \equiv A \otimes E$,
- $M_f \equiv C \otimes D$.

Le problème d'accessibilité entre les marquages M_0 et M_f en franchissant une fois chacune des transitions t_1 et t_2 se traduit par le séquent suivant : $A \otimes E, t_1, t_2 \vdash C \otimes D$. Afin de ne pas alourdir les notations et la représentation des séquents, nous avons choisi de représenter les transitions du réseau de Petri et les formules implicatives représentant ces transitions avec les mêmes notations. Par exemple, la formule t_1 dans le séquent représente la formule implicative ($A \multimap B \otimes D$) associée à la transition t_1 du réseau de Petri de la figure 2.6.

La preuve se déroule, canoniquement, comme suit :

$$\dfrac{\dfrac{A \vdash A\ id \qquad \dfrac{\dfrac{\dfrac{B \vdash B\ id \quad E \vdash E\ id}{B, E \vdash B \otimes E}\otimes R \quad \dfrac{C \vdash C\ id \quad D \vdash D\ id}{C, D \vdash C \otimes D}\otimes R}{B, D, E, B \otimes E \multimap C \vdash C \otimes D}\multimap L}{B \otimes D, E, B \otimes E \multimap C \vdash C \otimes D}\otimes L}{A, E, A \multimap B \otimes D, B \otimes E \multimap C \vdash C \otimes D}\multimap L}{A \otimes E, t_1, t_2 \vdash C \otimes D}\otimes L$$

Figure 2.6. **Arbre de preuve du séquent** $A \otimes E, t_1, t_2 \vdash C \otimes D$

La preuve se lit de bas en haut. Partant du séquent à prouver (ici $A \otimes E, t_1, t_2 \vdash C \otimes D$), on applique tout d'abord la règle $\otimes_L$ qui permet d'obtenir l'étape courante « A, E » à partir de la conjonction $A \otimes E$. A étant le seul atome permettant d'éliminer une formule implicative dans le membre gauche du séquent, on peut appliquer la règle $\multimap_L$ pour consommer l'atome A et la formule implicative $A \multimap B \otimes D$. Ceci correspond au franchissement de la transition t_1. L'application de cette règle génère deux nouveaux séquents ($A \vdash A$ et $B \otimes D, E, B \otimes E \multimap C \vdash C \otimes D$). La preuve du séquent gauche se termine en appliquant la règle *Identité*. Quant à celui de droite, on lui applique la règle $\otimes_L$ pour obtenir l'étape courante « B, D, E ». Poursuivons la preuve à partir de ce séquent. Seule peut s'appliquer la règle $\multimap_L$. Elle permet d'éliminer la formule $B \otimes E \multimap C$. Ceci correspond au franchissement de la transition t_2 et génère deux nouveaux séquents dont la preuve se termine en appliquant les règles $\otimes_R$ puis *Identité*.

Cet arbre de preuve permet donc de montrer l'accessibilité entre les marquages M_0 et M_f par franchissement des transitions t_1 et t_2. Nous avons présenté dans ce paragraphe comment construire un arbre de preuve en s'appuyant sur un exemple de réseau de Petri qui ne présente ni un conflit de transitions ni de jetons. Nous verrons ci-dessous que cette construction de l'arbre de preuve n'est pas tout à fait la même quand il s'agit d'un conflit de ce type.

III.D.5 Conflit de transitions et de jetons

La donnée de la structure d'un réseau de Petri, celle d'un marquage initial et celle d'un marquage final, ne définissent pas nécessairement un ordre partiel unique. Cela veut dire que lors de la construction de l'arbre de preuve du séquent correspondant, un certain nombre de décisions devront être prises, et que l'ordre partiel ne sera complètement fixé qu'une fois

toutes les décisions prises. Ces décisions correspondent à la résolution de conflits de transitions ou de jetons. Définissons tout d'abord ces notions avant de présenter la manière dont on les traite dans le cadre de la conduite de preuve.

On dit qu'il y a conflit entre deux transitions si l'un des atomes de l'étape courante est inclus dans les préconditions de ces deux transitions du séquent. Quant au conflit de jetons, celui-ci a lieu lorsque l'étape courante contient un nombre de jetons dans une place P supérieur au nombre de jetons nécessaires au tir d'une transition du séquent. Dans le premier cas, il faut choisir laquelle des deux transitions sera franchie. Dans le second cas, il faut décider du ou des jetons qui seront consommés.

A la rencontre d'un conflit (de transitions ou de jetons), une décision est donc à prendre quant à sa résolution. Chaque décision engendre un ordre partiel différent. C'est pour cette raison que nous avons opté pour la construction d'un arbre de preuve pour chaque choix. De manière générale, tout conflit de transitions (de jetons) entraîne la construction d'autant d'arbres de preuve différents que de transitions (de jetons) dans ce conflit. Ce qui revient à associer un ordre partiel par arbre de preuve canonique. Nous allons voir maintenant comment représenter un ordre partiel par un graphe de précédence.

III.E Graphe de précédence

La logique Linéaire permet non seulement de montrer l'accessibilité entre deux marquages dans un réseau de Petri (par preuve du séquent correspondant) mais permet également de construire l'ensemble des relations de précédence entre les franchissements des transitions menant du marquage initial au marquage final. Pour cela, il suffit de compléter l'arbre de preuve du séquent avec les informations nécessaires pour établir des relations de causalité entre les tirs des transitions. Nous rajouterons donc des annotations aux règles et aux atomes dans l'arbre de preuve. Nous définirons tout d'abord ce qu'est un graphe de précédence avant de présenter une méthode d'annotation de l'arbre de preuve à partir de laquelle le graphe sera construit.

III.E.1 Définition

Un **graphe de précédence** est défini par un ensemble fini d'événements E et une relation de précédence qui est un sous ensemble A de $E \times E$. C'est un graphe dont les sommets sont les événements de E et les arcs les éléments de A. Si un arc relie l'événement e_i à l'événement e_j ($e_i \rightarrow e_j$) alors e_i précède e_j et le couple $(e_i, e_j) \in A$. Un graphe de précédence cohérent est acyclique. Un graphe de précédence acyclique définit une relation d'ordre partiel entre les éléments de E.

Si on s'intéresse aux relations de précédence dans un réseau de Petri, les événements du graphe de précédence sont assimilés aux instances de franchissements des transitions du réseau. Reprenons le réseau de Petri de la figure 2.5 avec le même marquage initial (un jeton dans A et un jeton E) et le même marquage final (un jeton dans C et un jeton dans D). Le graphe de précédence entre les instances de franchissement des transitions de ce réseau est celui de la figure 2.7.

I^1 et I^2 sont les événements initiaux correspondant à la production des jetons du marquage initial. Quant aux événements finaux F^1 et F^2, ils correspondent à la consommation des jetons du marquage final. Pour les transitions, t_1^1 signifie que t_1 est franchie une première fois. Les arcs sont étiquetés par les noms des jetons consommés.

Reste maintenant à savoir comment construire un graphe de précédence à partir d'un réseau de Petri à l'aide de la logique Linéaire. Cela est possible grâce aux annotations des règles appliquées et des atomes consommés ou produits au cours de la preuve. Avec les annotations, nous notons précisément quelle règle a produit un jeton et quelle règle le consomme.

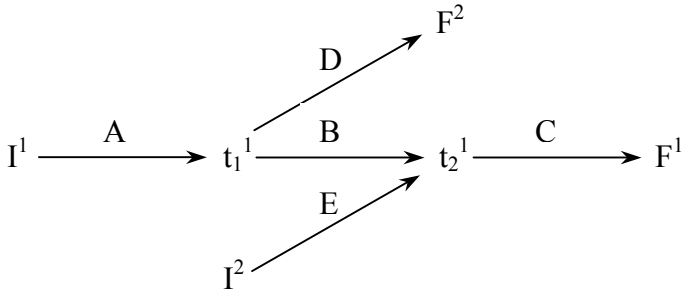


Figure 2.7. **Graphe de précédence**

III.E.2 Annotation de l'arbre de preuve canonique

Afin de construire un graphe de précédence à partir d'un arbre de preuve canonique, nous allons associer quelques annotations aux atomes logiques et aux applications des règles. Ceci permet de mettre en évidence le fait qu'il existe un ordre partiel pour l'application des règles lors de la construction de l'arbre de preuve canonique et que cet ordre est complètement spécifié par un seul arbre canonique annoté.

III.E.2.1 Annotation des règles

A chaque fois que nous appliquons la règle $\multimap L$ (pour éliminer une instance de formule décrivant une transition), nous associons une étiquette avec le nom de la transition correspondant à la formule implicative éliminée. Lorsqu'une transition est franchie plusieurs fois, nous mettons en exposant un indice égal au nombre de franchissements effectués. Ainsi l'étiquette t_i^j signifie que c'est la j^{ème} élimination d'une formule associée à la transition t_i. Cette annotation permet de différencier les tirs d'une même transition.

III.E.2.2 Annotations des atomes

Chaque atome d'une étape courante est étiqueté différemment selon qu'il est à gauche ou à droite du tourniquet du séquent à prouver. Lorsque cet atome est à gauche du tourniquet, cette étiquette est celle de la règle qui l'a produit. Quand il est à droite, il prend l'étiquette de la règle qui l'a consommé. Supposons que l'on applique la règle $\multimap L$ à la formule $\Pr e(t) \multimap Post(t)$ pour la j^{ème} fois dans le séquent $\Gamma_1, \Gamma_2, \Pr e(t) \multimap Post(t), l \mid\!- M_f$ produisant ainsi les deux séquents $\Gamma_1 \mid\!- \Pr e(t)$ et $\Gamma_2, Post(t), l \mid\!- M_f$ dans la ligne supérieure de l'arbre de preuve. Alors tous les atomes de *Post(t)* seront étiquetés par t^j (c'est la j^{ème} instance de franchissement de la transition t qui a produit ces atomes). Quant aux atomes de *Pre(t)*, ils seront étiquetés aussi par t^j car c'est cette j^{ème} instance qui a consommé ces atomes.

S'il y a n atomes initiaux dans la formule du marquage initial M_0, ils seront étiquetés par I^k pour k variant de 1 à n. Quant aux atomes de la formule représentant le marquage final, ils seront étiquetés par F^k pour k variant de 1 à m si M_f contient m jetons.

Maintenant que nous avons montré comment annoter les atomes et les règles, nous présentons ci-dessous comment construire le graphe de précédence.

III.E.3 Construction du graphe de précédence

Le graphe de précédence est construit à partir de l'arbre de preuve annoté et est basé sur la notion de causalité liant le tir de deux transitions lorsqu'on ne peut tirer l'une sans avoir au préalable tiré l'autre. Autrement dit, cette notion de causalité lie les jetons consommés par l'une aux jetons produits par l'autre par l'application de la règle d'élimination à gauche de l'implication Linéaire. Cette règle ne peut pas s'appliquer n'importe quand pour n'importe quelle transition. En effet, il faut que les jetons soient disponibles dans l'étape courante. Cela veut dire qu'ils ont été soit initialement présents, soit préalablement produits par le franchissement d'une autre transition. Dans l'exemple du paragraphe précédent, il faut que les atomes du bloc Γ_1 (listes d'atomes séparés par des virgules) soient identiques à ceux de la formule *Pre(t)* sinon le séquent $\Gamma_1 \vdash \mathrm{Pr}e(t)$ ne sera pas prouvable.

Avec les annotations, nous notons précisément quelle règle a produit un jeton et quelle règle le consomme. Ce sont les applications de la règle Identité qui associent les jetons produits aux jetons consommés. A partir des feuilles terminales de l'arbre de preuve, nous construisons le graphe de précédence de la manière suivante : chaque arc est défini par l'application de la règle Identité (nœud terminal de l'arbre de preuve). Chacune de ces règles associe un atome logique produit à un atome logique consommé. Les nœuds sont les applications des règles d'élimination de l'implication Linéaire c'est-à-dire les instances de franchissement des transitions. Les nœuds sources des arcs sont les franchissements ayant produit les atomes tandis que les nœuds destinations sont les franchissements les ayant consommés. Nous avons choisi de mettre des étiquettes initiales et finales pour représenter des événements virtuels de production de jetons initiaux et de consommation de jetons finaux.

III.E.4 Exemple sans conflit

Reprenons l'arbre de preuve de la figure 2.6. Cet arbre donne la preuve du séquent $A \otimes E, t_1, t_2 \vdash C \otimes D$ et prouve l'accessibilité entre les marquages initial et final dans le réseau de Petri de la figure 2.5. Si on applique à cet arbre les annotations adéquates, on obtient l'arbre de la figure 2.8. Si l'on construit le graphe de précédence à partir de cet arbre, on trouve celui de la figure 2.7.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{B(t_1^1) \vdash B(t_2^1)}\,id \quad \dfrac{}{E(I^2) \vdash E(t_2^1)}\,id}{B(t_1^1), E(I^2) \vdash B(t_2^1) \otimes E(t_2^1)} \otimes R \quad \dfrac{\dfrac{}{C(t_2^1) \vdash C(F^1)}\,id \quad \dfrac{}{D(t_1^1) \vdash D(F^2)}\,id}{C(t_2^1), D(t_1^1) \vdash C(F^1) \otimes D(F^2)} \otimes R}{B(t_1^1), D(t_1^1), E(I^2), B \otimes E \multimap C \vdash C(F^1) \otimes D(F^2)} \multimap L(t_2^1) \quad \dfrac{}{A(I^1) \vdash A(t_1^1)}\,id}{B(t_1^1) \otimes D(t_1^1), E(I^2), B \otimes E \multimap C \vdash C(F^1) \otimes D(F^2)} \otimes L}{A(I^1), E(I^2), A \multimap B \otimes D, B \otimes E \multimap C \vdash C(F^1) \otimes D(F^2)} \multimap L(t_1^1)}{A(I^1) \otimes E(I^2), t_1, t_2 \vdash C(F^1) \otimes D(F^2)} \otimes L$$

Figure 2.8. **Arbre de preuve annoté**

Cet arbre de preuve annoté permet de montrer l'accessibilité entre les marquages $A \otimes E$ et $C \otimes D$ par franchissement des transitions t_1 et t_2. L'information supplémentaire extraite de cet arbre de preuve est l'ordre dans lequel les transitions du réseau sont franchies pour parvenir au marquage final. La transition t_1 est en séquence avec t_2 (t_1 ; t_2). L'arbre de preuve permet donc de définir des relations de causalité, liées aux marquages, entre les transitions du réseau de Petri. Ici, t_2 ne peut être tirée avant t_1 car seul le tir de t_1 peut produire un jeton dans la place B qui est une place d'entrée de t_2. Toutefois, cette relation de causalité changerait si la

place B était initialement marquée. Ceci illustre le fort lien entre la notion de marquage et les relations de causalité dans un réseau de Petri.

Jusqu'ici, nous avons vu comment conduire la preuve d'un séquent, comment gérer les conflits en dupliquant l'arbre de preuve et comment construire les différents graphes de précédence associés à ce séquent. Nous voulons introduire ci-dessous une méthode d'étiquetage des atomes dans les séquents qui permet d'associer un ordre partiel unique à chaque séquent annoté.

III.F Séquent caractéristique d'un seul ordre partiel

Comme nous l'avons vu dans le paragraphe III.D.5, la preuve d'un séquent peut donner naissance à plusieurs arbres de preuve canonique, et chaque arbre correspond à un ordre partiel (sous la forme d'un graphe de précédence). Nous voulons modifier l'écriture du séquent afin qu'il corresponde à un seul ordre partiel, une fois que tous les conflits sont résolus. Pour cela, nous allons revenir sur le processus d'étiquetage défini au paragraphe précédent qui permettait d'extraire l'ordre partiel.

Dans le paragraphe précédent, nous avons introduit un étiquetage qui n'influait pas sur la construction de l'arbre de preuve. Ici, nous utilisons un étiquetage qui modifie l'identité des atomes et qui va donc influencer la construction de l'arbre car la règle identité ne s'appliquera que si les atomes sont identiques compte tenu de l'étiquetage. La règle Identité associe en effet un atome produit (celui qui se trouve à gauche dans le séquent) à un atome consommé (celui qui se trouve à droite). Si l'on étiquette l'atome consommé par le nom du franchissement de la transition qui l'a produit et si l'on porte cette étiquette dans la formule associée au franchissement de transition correspondant dans le séquent de départ, et si, dans cette même formule, l'on étiquette les atomes produits (à droite de l'implication linéaire) par le nom du franchissement correspondant, alors on restreint les possibilités de construction des arbres de preuve au seul ordre partiel retenu.

En effet, ceci permet d'imposer la consommation d'un jeton bien spécifique par le franchissement de la transition en question, ce qui n'est pas le cas dans l'arbre de preuve ne prenant pas en compte les annotations. Lorsque l'on a plusieurs jetons dans la même place d'un réseau de Petri, l'arbre de preuve sans annotation ne différencie pas les divers atomes qui leur sont associés puisqu'ils sont tous nommés par le nom de la place qui les contient. Une fois l'annotation faite, ils sont pré-affectés à des franchissements de transitions spécifiques.

Soit une formule implicative t du séquent (t est de la forme $\mathrm{Pr}\,e(t) \multimap Post(t)$). Chaque atome de Pre(t) ou de Post(t) sera étiqueté par l'instance de franchissement de la transition qui l'a produit. Si j est un atome de Pre(t), alors j sera étiqueté par $(t')^m$ si c'est la transition t' qui l'a produit à son $m^{ème}$ franchissement. Si p est un atome de Post(t), il sera étiqueté par t^n (le $n^{ème}$ franchissement de t).

III.G Exemple avec conflit

Considérons le réseau de Petri de la figure 2.9 et le problème d'accessibilité posé par le séquent : $A \otimes B, t_1, t_2, t_3, t_4 \mid\!\!- D \otimes E$. La preuve de ce séquent est donnée en annexe.

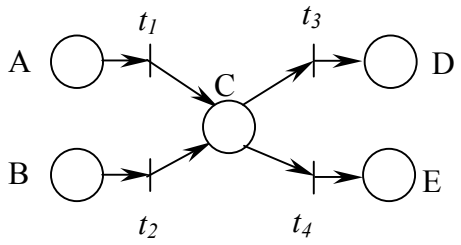


Figure 2.9. **Exemple de réseau de Petri avec conflits**

Cette preuve met en évidence deux ensembles de relations de précédence correspondant chacun à un choix des jetons dans la place C pour le tir des transitions t_3 et t_4. Le premier ensemble est représenté par le graphe de précédence de la figure 2.10. Il correspond au choix de franchir la transition t_3 par le jeton dans C produit par t_1, c'est-à-dire le jeton initialement dans A, et celui de t_4 par le jeton dans C produit par t_2, c'est-à-dire le jeton initialement dans la place B.

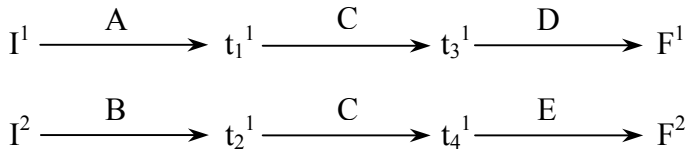


Figure 2.10. **Premier graphe de précédence**

Quant au deuxième ensemble de relations de précédence (figure 2.11), il correspond au tir de t_3 avec le jeton dans C initialement présent dans B et à celui de t_4 par le jeton dans C initialement présent dans la place A.

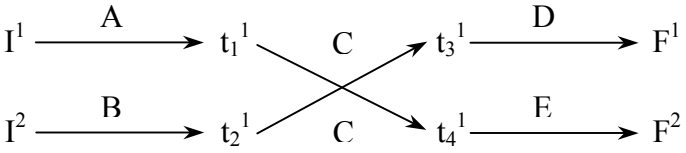


Figure 2.11. **Deuxième graphe de précédence**

Afin d'associer à chaque graphe de précédence un séquent caractéristique unique, nous appliquons au séquent $A \otimes B, t_1, t_2, t_3, t_4 \mid\!\!- D \otimes E$ les annotations introduites dans le paragraphe précédent. Nous obtenons ainsi les deux séquents annotés suivants correspondant respectivement au premier et au deuxième ensemble de relations de précédence ci-dessus :

$$A.I_1 \otimes B.I_2, \underbrace{A.I_1 \multimap C.t_1^{\,1}}_{t_1}, \underbrace{B.I_2 \multimap C.t_2^{\,1}}_{t_2}, \underbrace{C.t_1^{\,1} \multimap D.t_3^{\,1}}_{t_3}, \underbrace{C.t_2^{\,1} \multimap E.t_4^{\,1}}_{t_4} \mid\!\!- D.t_3^{\,1} \otimes E.t_4^{\,1} \quad (1)$$

$$A.I_1 \otimes B.I_2, \underbrace{A.I_1 \multimap C.t_1^{\,1}}_{t_1}, \underbrace{B.I_2 \multimap C.t_2^{\,1}}_{t_2}, \underbrace{C.t_2^{\,1} \multimap D.t_3^{\,1}}_{t_3}, \underbrace{C.t_1^{\,1} \multimap E.t_4^{\,1}}_{t_4} \vdash D.t_3^{\,1} \otimes E.t_4^{\,1} \quad (2)$$

Nous pouvons remarquer que les relations de précédence peuvent être déduites directement à partir du séquent. Elles sont données par la liste des franchissements. En effet, l'étiquette du franchissement est portée par les atomes produits (ceux de Post(t)) et chaque atome consommé (ceux de Pre(t)) porte l'étiquette de l'événement qui l'a produit. Nous avons donc tous les arcs entrant dans le nœud événement. Les arcs aboutissant aux nœuds terminaux sont donnés par les étiquettes du marquage final. Il suffit de rajouter un nœud initial par atome du marquage initial et un nœud terminal par atome du marquage final.

III.H Apport de la logique Linéaire

Comme il y a équivalence entre accessibilité dans un réseau de Petri et prouvabilité de certains séquents en logique Linéaire, la question de l'apport de cette nouvelle technique de recherche d'accessibilité dans les réseaux de Petri par rapport aux techniques habituelles se pose tout naturellement. Rappelons tout d'abord ces techniques avant de souligner l'apport de la logique Linéaire.

L'accessibilité entre deux marquages dans un réseau de Petri peut être traitée de deux façons différentes. La première permet d'aborder ce problème en faisant appel à l'algèbre linéaire en résolvant l'équation caractéristique : $M_f = M_0 + C.\bar{s}$. La résolution de cette équation permet de déterminer une condition nécessaire mais non suffisante d'accessibilité. Le vecteur caractéristique $\bar{s}$ ne donne aucun ordre de franchissement des transitions mais uniquement le multi-ensemble (non ordonné) des transitions à franchir. Cette condition nécessaire, mais non suffisante, peut s'interpréter de la sorte : s'il existe un chemin entre les marquages M_0 et M_f, il passera forcément par le multi-ensemble $\bar{s}$ calculé. Par contre, l'algèbre linéaire ne définit ni l'ordre des franchissements ni la possibilité effective de ces franchissements (qui dépend du marquage initial).

La seconde méthode fait appel au graphe des marquages accessibles (quand le réseau de Petri est borné) à partir du marquage initial M_0. Une fois ce graphe généré, il reste à déterminer, par exploration du graphe, une séquence de transitions permettant d'accéder au marquage final M_f. Dans ce cas, l'information est totalement ordonnée et la condition est nécessaire et suffisante. Toutefois, cet ordre total ne permet pas de distinguer les liens de causalité réels car les franchissements sont effectués les uns après les autres en vérifiant à chaque pas que le marquage intermédiaire est suffisant pour poursuivre.

Comme nous venons de l'évoquer, les deux méthodes habituelles de traitement de l'accessibilité dans un réseau de Petri sont incomplètes. En effet, dans un cas, elles ne permettent pas de donner une condition nécessaire et suffisante d'accessibilité, et dans l'autre cas, l'information récoltée ne permet pas de bien caractériser les relations d'ordre partiel entre les instances de tir des transitions.

Que peut donc nous apporter cette nouvelle caractérisation de l'accessibilité en logique Linéaire ? Cette approche transpose le problème de l'accessibilité en un problème de preuve de séquent plus simple et plus rapide, ce qui donne un cadre logique garantissant le respect de la cohérence des causalités et des ordres partiels. Cette preuve fournit une condition nécessaire et suffisante quant à l'accessibilité entre deux marquages en fournissant le multi-ensemble des transitions impliquées. De plus, il est possible d'extraire de l'arbre de preuve canonique des informations sur l'ordre partiel entre les franchissements des transitions

concernées (ce que nous venons de présenter) ainsi que des évaluations temporelles de scénarios de réseaux de Petri temporels (en associant des labels temporels aux jetons : date de production et date de consommation) [Kunzle 97] et [Rivière 00]. La logique Linéaire est considérée, par conséquent, comme un outil d'analyse logique supplémentaire des réseaux de Petri permettant de caractériser les ordres partiels.

Afin d'illustrer l'apport de la logique Linéaire par rapport aux outils classiques de traitement de l'accessibilité dans les réseaux de Petri, nous traitons un exemple simple. Nous nous intéressons à la caractérisation des ordres partiels des franchissements de transitions permettant de passer du marquage de la place P_1 au marquage de la place P_6 dans le réseau de Petri de la figure 2.12.

La première idée qui nous vient à l'esprit est de simuler le réseau de Petri à partir du marquage initial (un jeton dans P_1), de générer le graphe des marquages accessibles et de vérifier a posteriori que le marquage final (un jeton dans P_6) appartienne à ce graphe, ce qui nous permettra par la suite de déterminer les séquences de franchissements de transitions permettant de passer du marquage initial au marquage final. Sur l'exemple, aucune transition n'est franchissable à partir du seul marquage de la place P_1. Ce marquage est insuffisant pour permettre d'accéder au marquage de P_6. La question qui se pose donc est de définir un marquage initial, contenant celui de P_1, nécessaire et suffisant. Il doit être nécessaire pour permettre d'accéder au marquage de P_6 et suffisant pour ne pas induire des franchissements non indispensables, il doit donc être minimal. Il nous faut manipuler des marquages partiels et non des marquages totaux. Dans le chapitre suivant, nous exposerons une méthode basée sur la logique Linéaire qui permet de définir ces marquages partiels nécessaires et suffisants pour permettre d'accéder au marquage partiel final.

Si nous utilisons l'équation fondamentale $M_f = M_0 + C.\bar{s}$, nous en déduisons qu'il faut franchir les transitions t_1 et t_2. Si nous cherchons un marquage suffisant pour franchir ces transitions, nous en déduisons qu'il faut un jeton dans chacune des places P_2, P_3 et P_5 en plus de celui dans P_1. Par conséquent, les transitions t_1 et t_2 peuvent être franchies dans n'importe quel ordre. Pourtant, une analyse intuitive des relations de causalité montre que si l'on franchit t_2 avant t_1, ce n'est pas la consommation du jeton dans P_1 qui permet de produire le jeton dans P_6, alors que si l'on franchit t_1 avant t_2, c'est le cas et le jeton mis initialement dans P_2 est inutile. La logique linéaire va nous permettre de formaliser ce type de raisonnement intuitif.

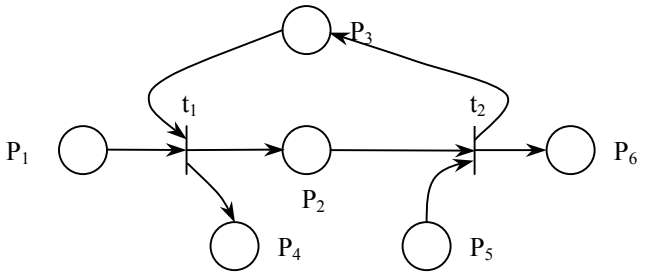


Figure 2.12. **Exemple de réseau de Petri**

L'exemple traité ci-dessus est assez simple et peut être facilement traité à la main sans avoir recours à aucune des techniques évoquées. Pour des modèles de réseau de Petri assez complexes, le traitement de l'accessibilité est inenvisageable sans les outils adéquats. Générer le graphe d'accessibilité à partir d'un marquage initial très réduit peut conduire à un échec (il manque des jetons dans certaines places du réseau). Générer le graphe à partir d'un marquage tel que toutes les places sont marquées est aussi à écarter à cause de l'explosion combinatoire

du nombre d'états. Nous avons développé une méthode qui permet de caractériser les ordres partiels et d'enrichir, quand cela est indispensable, le marquage partiel initial tout en garantissant la cohérence des raisonnements, et ce, dans le cadre formel de la logique Linéaire.

Dans le chapitre suivant, nous présenterons cette méthode qui permet de caractériser l'accessibilité entre deux marquages partiels par des relations de causalité et qui permet, de ce fait, de construire des scénarios minimaux.

Chapitre 3. Extraction des scénarios redoutés à partir d'un modèle RdP

I Introduction

Dans le chapitre précédent, nous avons présenté une modélisation des systèmes mécatroniques basée sur deux points de vue complémentaires : l'aspect hybride et l'aspect logique. L'aspect hybride reflète l'interaction entre la partie continue (modélisée par des équations algébro-différentielles) et la partie discrète (modélisée par un réseau de Petri). L'aspect logique utilise une interprétation en logique Linéaire du modèle réseau de Petri. Il est fondé sur la notion de causalité sous-jacente aux scénarios redoutés. En effet, l'analyse des conditions d'occurrence de l'événement redouté, par exemple la sortie d'une variable continue de son domaine de sécurité (franchissement d'un seuil de sécurité modélisé par le franchissement d'une transition) permet de retrouver les relations de causalité qui expliquent le scénario redouté (menant à l'événement redouté). Comme nous avons choisi une modélisation des systèmes mécatroniques permettant une bonne séparation entre les aspects discret et continu, la recherche des relations de causalité à partir du modèle discret est possible. Nous travaillerons par la suite uniquement sur le modèle réseau de Petri et nous utiliserons la logique Linéaire pour extraire les scénarios redoutés.

L'objectif de ce chapitre est de présenter une méthode permettant de trouver les scénarios redoutés et de les caractériser par des relations de causalité entre événements. Ces relations induisent un ordre partiel entre les événements. Nous commençerons tout d'abord par définir de manière formelle la notion de scénario redouté en terme d'ordre partiel permettant l'accessibilité d'un état donné (dit état redouté). Ensuite, nous présenterons deux approches de l'accessibilité, avant d'introduire une nouvelle notion qui est celle du raisonnement dans un contexte inconnu. C'est cette dernière qui permet de déterminer les scénarios redoutés minimaux, c'est à dire sans entrelacement avec des comportements d'éléments non causalement impliqués dans l'accessibilité de l'état partiel redouté. Nous présenterons, enfin, la méthode de recherche de scénarios redoutés qui est basée sur la logique Linéaire.

II Scénarios redoutés

II.A Définition

Définissons tout d'abord ce que c'est qu'un scénario. Un scénario sous-entend un début, une fin et une histoire qui décrit l'évolution d'un système. Dans le contexte de la sûreté de fonctionnement, un scénario redouté mène à un état catastrophique ou dangereux : c'est l'état final (dit état redouté). L'état initial est un état de bon fonctionnement du système. Le scénario redouté décrit de manière précise (ce qui est nécessaire pour la compréhension) et concise (le juste nécessaire : causalité) comment le système quitte le bon fonctionnement pour évoluer vers un fonctionnement jugé dangereux. C'est en effet une description du système sous la forme de changements d'états et de suites d'événements qui mènent vers l'état redouté. C'est une explication claire des raisons pour lesquelles le système s'est trouvé ou risque de se trouver dans un état redouté donné. Ce n'est donc pas simplement une suite

d'événements sans liens de causalité entre eux mais, comme nous l'avons évoqué à la fin du chapitre précédent (section III.H), nous les considérons comme un ensemble de relations d'ordre qui peuvent être interprétées comme des relations de causalité entre certains événements, ceux qui ne sont pas reliés par une relation d'ordre étant considérés comme des événements parallèles.

En résumé, un scénario redouté est une description de l'évolution de certains composants du système global à partir d'un état de bon fonctionnement jusqu'à l'occurrence de l'événement redouté. Ce scénario fait donc intervenir uniquement les composants ayant un lien de causalité avec l'occurrence de l'événement redouté.

Les combinaisons de défaillances des composants élémentaires d'un système forment la majeure partie des causes possibles pouvant provoquer les scénarios redoutés. Ces combinaisons peuvent être identifiées par un raisonnement en parcourant l'ensemble des combinaisons possibles des défaillances élémentaires. Dans certains cas, ces défaillances élémentaires sont bénignes pour le fonctionnement du système mais, combinées avec des interactions entre certains sous-systèmes (partage de ressource ou partage de variable continue), ces défaillances peuvent être à l'origine des scénarios redoutés. L'objectif de notre méthode est d'identifier ces scénarios redoutés de façon exhaustive. Pour cela, nous utilisons une approche formelle (la logique Linéaire) pour extraire ces scénarios à partir du modèle du système en réseau de Petri. Commençons tout d'abord par formaliser ces scénarios par rapport au formalisme des réseaux de Petri, puis, en logique Linéaire.

II.B Formalisation en RdP et en logique Linéaire

Dans le cadre des réseaux de Petri, les événements sont les instances de tir des transitions et les états partiels sont les marquages partiels. Ainsi, un scénario peut être vu comme une relation d'ordre partiel entre les instances de tirs d'un ensemble de transitions par rapport à un marquage partiel initial et à un marquage partiel final. Compte tenu de l'équivalence entre accessibilité entre deux marquages en réseau de Petri et la prouvabilité du séquent correspondant, un scénario peut être formalisé en logique Linéaire par un séquent de la forme : $M_i, Liste\_Transitions \vdash M_f$, où M_i est la liste des atomes du marquage partiel initial, « *Liste_Transitions »* est la liste des transitions à tirer. M_f est la liste des atomes du marquage partiel final. Dans le cas de la formalisation d'un scénario redouté, le marquage final englobe l'état partiel redouté en y ajoutant les états partiels inévitablement atteints en même temps que ce dernier.

Après avoir défini les scénarios redoutés, nous allons introduire deux approches équivalentes de l'accessibilité entre deux marquages dans les réseaux de Petri. Ces deux approches nous serviront pour expliquer, par la suite, la notion de raisonnement dans un contexte inconnu. Cette notion est à la base de la recherche des scénarios redoutés.

III Accessibilité entre deux marquages : deux approches duales

L'accessibilité entre deux marquages peut être vu sous deux angles : le premier consiste à partir du marquage initial, et à construire les états successeurs en tirant une à une les transitions franchissables de la liste pré-établie jusqu'à atteindre le marquage final, c'est **l'accessibilité avant**. Le marquage initial représente l'état présent alors que le marquage final représente un état futur. Quant à **l'accessibilité arrière**, le marquage final est considéré comme l'état présent et le marquage initial est vu comme un état du passé. On construit, à

partir de l'état présent, les états prédécesseurs en supposant avoir franchi une à une les transitions de la liste pré établie. Commençons par la formalisation de l'accessibilité avant en logique Linéaire.

III.A Accessibilité avant

A partir d'un réseau de Petri R muni d'un marquage initial M_0, l'accessibilité du marquage M_f par les franchissements de transitions de la liste l, est équivalent à la preuve du séquent $M_0, l \vdash M_f$. On construit l'arbre de preuve canonique annoté, ce qui permet d'obtenir le graphe de précédence. Ce graphe représente les relations de causalité entre les instances de franchissement des transitions.

Comme nous l'avons introduit au chapitre précédent (section III.D), la construction de l'arbre de preuve est basée sur l'application de la règle d'introduction à gauche de l'implication Linéaire ($\multimap_L$). L'application de cette règle correspond au franchissement de la transition correspondante du réseau. Elle opère sur la partie gauche du séquent et consomme la formule implicative de la transition concernée ainsi que les atomes de l'étape courante indispensable à son franchissement. A partir du marquage initial et au fur et à mesure des applications successives de cette règle, on consomme et on produit des atomes jusqu'à l'obtention du marquage final. On part effectivement du marquage initial (une liste d'atomes initialement disponibles) pour aboutir au marquage final (une liste d'atomes initialement indisponibles mais qui seront produits par la preuve).

Cette démarche est la plus naturelle dans les problèmes d'accessibilité. Toutefois, il est possible de procéder autrement, et ce en partant non plus du marquage initial M_0 mais du marquage final M_f et de construire ses prédécesseurs jusqu'à obtenir M_0. C'est le principe de l'accessibilité arrière.

III.B Accessibilité arrière

Au lieu de chercher les transitions franchissables à partir d'un marquage donné du réseau de Petri et avancer en les tirant une à une, on cherche celles qui ont du être tirées par une démarche abductive, et on remonte dans le passé en cherchant les causes de ces franchissements.

Intuitivement, on pourrait penser que l'accessibilité arrière de M_f vers M_0 est finalement la même que l'accessibilité avant de M_0 vers M_f, c'est à dire que les marquages et les séquences explorées sont identiques. Nous allons voir que ce n'est pas le cas. Nous présenterons par la suite une formalisation de l'accessibilité arrière basée sur l'inversion du réseau de Petri ainsi qu'un résultat d'équivalence entre les deux types d'approches d'accessibilité.

III.B.1 Définition de l'accessibilité arrière

L'accessibilité arrière d'un marquage est l'ensemble des marquages M_i tels qu'il existe une séquence σ_i telle que $M_i \xrightarrow{\sigma_i} M_f$. Illustrons cette notion au travers d'un exemple.

III.B.2 Exemple

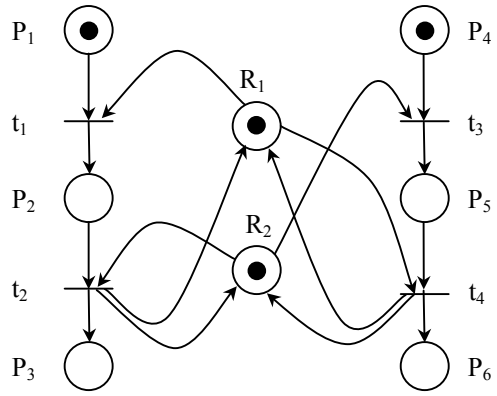


Figure 3.1. Exemple de réseau de Petri marqué

Nous voulons souligner au travers de cet exemple de réseau de Petri la différence entre l'accessibilité avant et l'accessibilité arrière. Nous étudierons pour cela le problème de l'accessibilité entre un marquage initial M_0 et un marquage final M_f en franchissant une fois chacune des transitions du réseau de Petri de la figure 3.1. M_0 est constitué d'un jeton dans chacune des places P_1, P_4, R_1 et R_2 du réseau. Quant à M_f, il est constitué d'un jeton dans chacune des places P_3, P_6, R_1 et R_2.

En partant de M_0 on trouve deux séquences menant à M_f : $M_0 \xrightarrow{t_1;t_2;t_3;t_4} M_f$ et $M_0 \xrightarrow{t_3;t_4;t_1;t_2} M_f$. On trouve aussi deux séquences menant au blocage mortel M_b (un jeton dans P_2 et un jeton dans P_5) : $M_0 \xrightarrow{t_1;t_3} M_b$ et $M_0 \xrightarrow{t_3;t_1} M_b$. Par contre, si l'on cherche tous les prédécesseurs de M_f, on ne rencontre pas M_b (bien évidemment puisque M_b n'a pas de successeur) et seules les séquences $M_0 \xrightarrow{t_1;t_2;t_3;t_4} M_f$ et $M_0 \xrightarrow{t_3;t_4;t_1;t_2} M_f$ sont obtenues. Bien évidemment, les deux séquences menant avec succès de M_0 vers M_f sont trouvées aussi bien par l'accessibilité avant que par l'accessibilité arrière, mais les processus de recherche ne sont pas les mêmes.

De manière générale, l'accessibilité avant permet de mettre en évidence non seulement les scénarios menant du marquage initial au marquage final, mais également ceux qui ne mènent pas vers ce marquage final. Réciproquement, l'accessibilité arrière identifie les scénarios qui mènent du marquage initial au marquage final, et aussi ceux qui mène au marquage final sans passer par le marquage initial.

Nous présentons dans la suite de ce paragraphe une formalisation de l'accessibilité arrière basée sur l'accessibilité avant appliquée au réseau de Petri inversé. Définissons tout d'abord ce qu'est un réseau de Petri inversé.

III.B.3 Réseau de Petri inversé

III.B.3.1 Définition informelle

Soit R un réseau de Petri. On appelle réseau de Petri inversé le réseau de Petri obtenu une fois que l'on a inversé tous les arcs du réseau R. Notons R^{-1} ce nouveau réseau de Petri.

III.B.3.2 Définition formelle

Soit un réseau de Petri $\boldsymbol{R}$ = <*P, T, Pre, Post*> où :

- *P* est un ensemble fini de places,
- *T* est un ensemble fini de transitions,
- *Pre* est l'application incidence avant (places précédentes),
- *Post* est l'application incidence arrière (places suivantes).

Nous appellons réseau de Petri inversé le réseau $\boldsymbol{R}^{-1}$ tel que $\boldsymbol{R}^{-1} = <P', T', Pre', Post'>$ où :

- P' = P,
- T' = T,
- Les applications Pre' et Post' sont définies telles que $\forall\ p \in P, \forall t \in T \begin{cases} Pre'(p,t) = Post(p,t) \\ Post'(p,t) = Pre(p,t) \end{cases}$.

III.B.4 Equivalence entre accessibilité arrière et accessibilité avant sur le RdP inversé

Nous allons maintenant prouver que l'accessibilité arrière entre deux marquages M_0 et M_f est équivalente à l'accessibilité avant entre M_f et M_0 sur le réseau de Petri inversé. Ainsi, nous n'aurons pas besoin de développer deux algorithmes d'accessibilité (avant et arrière). Il suffira de travailler sur le réseau de Petri inversé chaque fois que nous aurons un problème d'accessibilité arrière.

Nous voulons montrer que l'ensemble des séquences de franchissement et des marquages obtenus lors de l'analyse de l'accessibilité arrière sur $\boldsymbol{R}$ est le même que celui obtenu pour l'accessibilité avant sur $\boldsymbol{R}^{-1}$ (après inversion des séquences). Comme l'analyse de l'accessibilité est un processus itératif, il suffit de faire la preuve pour une itération de base entre un marquage M_i et un marquage M_{i+1} tel que $M_i \xrightarrow{t_j} M_{i+1}$.

Dans $\boldsymbol{R}$, le fait que M_{i+1} soit accessible à partir de M_i par le franchissement de t_j implique que $M_i = M_{i+1} - Post(t_j) + Pre(t_j)$ et $M_i \geq Pre(t_j)$. Comme $M_i - Pre(t_j) = M_{i+1} - Post(t_j)$, $M_i \geq Pre(t_j)$ implique $M_{i+1} \geq Post(t_j)$.

Dans $\boldsymbol{R}^{-1}$, nous avons donc $M_{i+1} \geq Pre'(t_j)$ et $M_i = M_{i+1} - Pre'(t_j) + Post'(t_j)$, ce qui exprime l'accessibilité avant de M_i à partir de M_{i+1} par t_j dans $\boldsymbol{R}^{-1}$.

Réciproquement, si nous avons M_i accessible à partir de M_{i+1} par t_j dans $\boldsymbol{R}^{-1}$, alors $M_i = M_{i+1} - Pre'(t_j) + Post'(t_j)$ et $M_{i+1} \geq Pre'(t_j)$. Nous avons donc $M_i = M_{i+1} - Post(t_j) + Pre(t_j)$ et $M_i \geq Pre(t_j)$ qui expriment l'accessibilité arrière de M_i à partir de M_{i+1} par t_j dans $\boldsymbol{R}$.

Remarque : Dans [Khalfaoui 02] et [Demmou 02], nous avons utilisé une traduction du réseau de Petri et des marquages fondée sur le connecteur $\wp$ (par) pour l'étude de l'accessibilité arrière. En utilisant la loi de De Morgan et en remplaçant tous les atomes par leur négation, le séquent exprimant l'accessibilité arrière sous cette forme devient identique au séquent d'accessibilité avant dans $\boldsymbol{R}^{-1}$. Les deux approches sont donc bien identiques.

IV Raisonnement dans un contexte inconnu

Dans le paragraphe précédent, les scénarios menant d'un marquage initial à un marquage final (que ce soit pour l'accessibilité avant ou pour l'accessibilité arrière) sont complètement spécifiés. En effet, on connaît le marquage initial, le marquage final ainsi que la liste des transitions à tirer. Ce que l'on cherche à caractériser, c'est l'ordre partiel régissant le tir de ces transitions par des relations de causalité entre ces tirs. Or, dans notre problème, nous ne connaissons que partiellement les marquages initial ou final et nous ne connaissons pas la liste des transitions à tirer. En effet, nous ne connaissons que l'état partiel redouté et nous ignorons les contextes (c'est à dire les états des autres composants du système étudié) dans lesquels cet état partiel peut être atteint. Nous avons donc à chercher les scénarios qui peuvent être une conséquence logique d'un marquage partiel donné. C'est pour cette raison que l'on distingue l'accessibilité entre deux marquages (connus au préalable) du raisonnement. Ce dernier opère sur un scénario incomplètement spécifié et doit caractériser les conséquences logiques d'un marquage partiel. Nous avons vu précédemment qu'il est possible de ramener l'accessibilité arrière sur $\boldsymbol{R}$ à l'accessibilité avant sur $\boldsymbol{R}^{-1}$. Nous nous placerons par conséquent dans le seul contexte de l'accessibilité avant.

Cette approche est essentielle pour autoriser la modularité. Elle permet, en effet, d'analyser les conséquences logiques de l'état d'un module indépendamment de l'état des autres modules en l'absence d'interaction. Dès qu'il y a interaction entre le module étudié et un autre module, nous devons faire une hypothèse sur l'état de l'autre module pour continuer le raisonnement. Cette hypothèse concerne la présence d'un jeton dans une place et dans le formalisme que nous avons retenu cela se traduit par un enrichissement du marquage que nous expliciterons par la suite.

Le problème qui se pose est donc de savoir comment on doit écrire un séquent qui va provoquer la bonne recherche de scénarios à partir d'un marquage initial qui n'est que partiellement connu.

IV.A Raisonnement avant

IV.A.1 Principe

Le raisonnement avant consiste à chercher les conséquences logiques possibles à partir d'un marquage partiel donné. Il permet de répondre à la question suivante : comment peut-on quitter ce marquage partiel ?

Le séquent exprimant cette question est le suivant : $M_0 \otimes \Gamma, \mathrm{T} \vdash M_f \otimes \Delta$ où :

- M_0 est le marquage partiel initial connu.
- Γ est un contexte inconnu représentant un ensemble de jetons qui s'avèreront nécessaires au cours de la preuve. A la fin de la preuve, Γ contiendra, par exemple, les hypothèses sur les états des autres modules qui doivent être remplies pour expliquer le comportement anormal du module étudié.
- M_f est un marquage partiel final connu.
- Δ est la partie du marquage final initialement inconnu. A la fin de la preuve, Δ contiendra, par exemple, les effets de bord sur les états d'autres modules lorsqu'un module évolue vers un état redouté.

- T est la liste des formules de transitions qui seront consommées durant la preuve. A la fin de la preuve, T contiendra l'ensemble des franchissements de transitions correspondant au scénario étudié.

A partir de ce séquent, on déroule le raisonnement avant afin de trouver les scénarios évoluant à partir du marquage partiel initial M_0. Progressivement, les éléments Γ, T et Δ sont de mieux en mieux connus et, à la fin, nous pouvons écrire le séquent caractéristique du scénario trouvé. Illustrons cette démarche par un exemple.

IV.A.2 Exemple

Reprenons l'exemple du réseau de Petri de la figure 2.5 dans le chapitre précédent. Nous avons illustré au travers de cet exemple le principe de la conduite de preuve en logique Linéaire quand le contexte est complètement spécifié. Nous avons choisi ce même exemple (figure 3.2) pour illustrer le raisonnement avant.

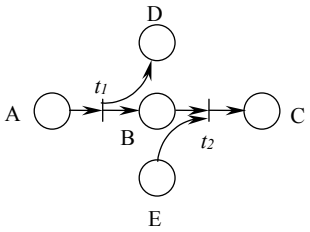


Figure 3.2. Exemple de réseau de Petri

Nous voulons déterminer les scénarios qui mènent du marquage partiel initial M_0 tel qu'il y a un jeton dans la place A (M_0 = A) au marquage partiel final M_f tel qu'il y a un jeton dans C (M_f = C). Afin de faciliter la compréhension de l'arbre de preuve, nous l'avons scindé en trois parties (1/3, 2/3 et 3/3), chacune illustrant une notion importante dans le processus de recherche de scénarios, à savoir la construction de la liste des transitions à tirer (partie 1/3), l'enrichissement de marquage (partie 2/3) et la fin de la preuve (dernière partie 3/3). La preuve commence comme suit :

$$\dfrac{\dfrac{\dfrac{\dfrac{}{A(I^1)\vdash A(t_1^{\,1})}\,Identité \quad \dfrac{\vdots}{B(t_1^{\,1})\otimes D(t_1^{\,1}),\Gamma,t_2,\mathrm{T}_2 \vdash C(F^1)\otimes\Delta}(\mathrm{T}_1 = t_2,\mathrm{T}_2)}{A(I^1),\Gamma,A \multimap B\otimes D,\mathrm{T}_1 \vdash C(F^1)\otimes\Delta}\multimap L\,(t_1^{\,1})}{A(I^1)\otimes\Gamma,t_1,\mathrm{T}_1 \vdash C(F^1)\otimes\Delta}(\otimes L)}{A(I^1)\otimes\Gamma,\mathrm{T} \vdash C(F^1)\otimes\Delta}(\mathrm{T} = t_1,\mathrm{T}_1)$$

Figure 3.3. Raisonnement avant (M_0 = A et M_f = C) : arbre de preuve annoté 1/3

La preuve commence par l'annotation des jetons initiaux et finaux (par exemple $A(I^1)$ et $C(F^1)$) dans le séquent initial racine de l'arbre : $A(I^1)\otimes\Gamma,\mathrm{T} \vdash C(F^1)\otimes\Delta$. Le raisonnement opère sur la partie gauche du séquent. Partant du marquage initial ($A(I^1)\otimes\Gamma$), on détermine les transitions pouvant consommer l'atome $A(I^1)$. La transition t_1 est la seule vérifiant ce critère. Etant donné qu'elle ne figure pas dans la liste des transitions T, un enrichissement de cette liste est indispensable pour avancer dans la preuve. On extrait par conséquent t_1 de T en

réécrivant $T : T = t_1, T_1$. T est considérée comme la concaténation de t_1 et d'une nouvelle liste de transitions T_1. A partir du séquent obtenu en remplaçant T par « t_1, T_1 », on applique autant de fois que cela est nécessaire (ici une fois) la règle $\otimes L$ pour éliminer le connecteur $\otimes$ dans le membre gauche du séquent. En remplaçant t_1 par sa formule implicative ($A \multimap B \otimes D$), on obtient le séquent suivant : $A(I^1), \Gamma, A \multimap B \otimes D, T_1 \vdash C(F^1) \otimes \Delta$. On franchit maintenant la transition t_1 en appliquant la règle $\multimap L$ et en annotant tous les atomes de la formule implicative $A \multimap B \otimes D$ par t_1^1 ($A(t_1^1)$, $B(t_1^1)$, $D(t_1^1)$), t_1^1 est la première instance de franchissement de t_1. On retrouve l'atome dans le séquent gauche ($A(I^1) \vdash A(t_1^1)$) et les autres atomes dans le séquent droit ($B(t_1^1) \otimes D(t_1^1), \Gamma, T_1 \vdash C(F^1) \otimes \Delta$). Le séquent gauche se termine par l'application de la règle *Identité*. La preuve continue pour le séquent droit. Quelles sont donc les transitions que l'on pourrait franchir à partir des atomes $B(t_1^1)$ et $D(t_1^1)$? t_2 est la seule transition. Or, elle ne figure pas dans la liste T_1. Cette dernière est par conséquent enrichie ($T_1 = t_2, T_2$).

$$\dfrac{\dfrac{\dfrac{\dfrac{}{B(t_1^1) \vdash B(t_2^1)}\,Identité \quad \dfrac{}{E(I^2) \vdash E(t_2^1)}\,Identité}{B(t_1^1), E(I^2) \vdash B(t_2^1) \otimes E(t_2)} \otimes R \quad \dfrac{\vdots}{D(t_1^1), C(t_2^1), \Gamma_1, T_2 \vdash C(F^1) \otimes \Delta}}{B(t_1^1), D(t_1^1), E(I^2), \Gamma_1, B \otimes E \multimap D, T_2 \vdash C(F^1) \otimes \Delta} \multimap L(t_2^1)}{B(t_1^1) \otimes D(t_1^1), E(I^2) \otimes \Gamma_1, t_2, T_2 \vdash C(F^1) \otimes \Delta} (\otimes L)^2}{B(t_1^1) \otimes D(t_1^1), \Gamma, t_2, T_2 \vdash C(F^1) \otimes \Delta} (\Gamma = E(I^2) \otimes \Gamma_1)$$

Figure 3.4. Raisonnement avant (M_0 = A et M_f = C) : arbre de preuve annoté 2/3

Passons maintenant à l'explication de la deuxième partie (figure 3.4). Partant du séquent enrichi par l'ajout de t_2 dans la liste T_1 à savoir $B(t_1^1) \otimes D(t_1^1), \Gamma, t_2, T_2 \vdash C(F^1) \otimes \Delta$, nous ne pouvons franchir t_2 car il lui manque un jeton dans la place E, d'où la nécessité d'**enrichir le marquage**. Un atome $E(I^2)$ annoté par l'événement qui l'a produit (I^2) est ajouté dans Γ ($\Gamma = E(I^2) \otimes \Gamma_1$). Nous appliquons ensuite la règle $\otimes L$ deux fois pour éliminer le connecteur $\otimes$ dans le membre gauche du séquent. Nous obtenons le séquent suivant : $B(t_1^1), D(t_1^1), E(I^2), \Gamma_1, B \otimes E \multimap D, T_2 \vdash C(F^1) \otimes \Delta$. A partir de ce séquent, on applique la règle $\multimap L$ pour franchir la transition t_2. Cela donne lieu à deux nouveaux séquents. Le séquent $D(t_1^1), C(t_2^1), \Gamma_1 \vdash C(F^1) \otimes \Delta$ sera prouvé dans la partie 3/3 de la figure 3.5. Quant au séquent $B(t_1^1), E(I^2) \vdash B(t_2^1) \otimes E(t_2^1)$, il est prouvé en appliquant successivement les règles $\otimes R$ et *Identité*.

$$\dfrac{\dfrac{\dfrac{}{C(t_2^1) \vdash C(F^1)}\,Identité \quad \dfrac{}{D(t_1^1) \vdash D(F^2)}\,Identité}{D(t_1^1), C(t_2^1) \vdash C(F^1) \otimes D(F^2)} \otimes R}{D(t_1^1), C(t_2^1), \Gamma_1, T_2 \vdash C(F^1) \otimes \Delta} (\Gamma_1 = 1; T_2 = 1; \Delta = D(F^2))$$

Figure 3.5. Raisonnement avant (M_0 = A et M_f = C) : arbre de preuve annoté 3/3

Intéressons nous maintenant à la fin de la preuve consistant à éliminer les parties inconnues à savoir ici Γ_1 et Δ. Le franchissement de la transition t_2 a produit l'atome C qui

constitue le marquage final M_f. Nous avons donc atteint à ce stade de la preuve le marquage voulu. La preuve doit donc s'arrêter. Pour cela, on suppose que le contexte résiduel est vide ($\Gamma_1 = 1, T_2 = 1$, 1 étant l'élément neutre du connecteur $\otimes$ que l'on peut éliminer) et qu'il y a identité entre les atomes des parties gauche et droite du séquent afin qu'il puisse être prouvable ($D \otimes C \equiv C \otimes \Delta$) d'où $\Delta = D$. La preuve se termine en appliquant successivement les règles $\otimes R$ et *Identité*.

Le séquent caractérisant cette accessibilité est $A \otimes E, t_1, t_2 \vdash C \otimes D$ ($\Gamma = E$, $T = \{t_1, t_2\}$, $\Delta = D$).

A partir de ce raisonnement avant, on construit le graphe de précédence (figure 3.6) qui est identique à celui donné dans le chapitre précédent. Nous avons utilisé des arcs en pointillé afin de mettre en évidence les jetons enrichis et les effets de bord. L'arc en pointillé dirigé vers le nœud final (F^2) indique un jeton du marquage final produit obligatoirement (effet de bord) avec M_f (le jeton D dans la place D). Quent àcelui sortant du nœud initial (I^2), il met en évidence un enrichissement de marquage et porte le nom du jeton enrichi (le jeton E dans la place E).

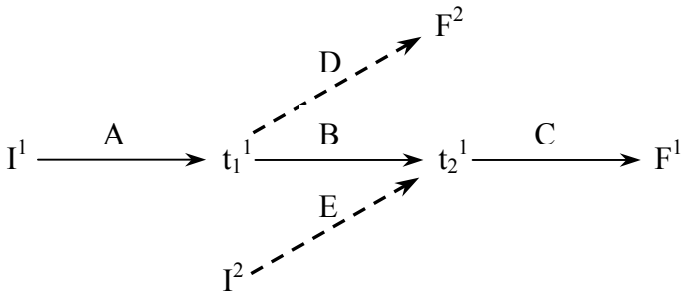


Figure 3.6. Graphe de précédence construit à la suite du raisonnement avant

Que peut-on conclure suite à ce raisonnement avant ? Une première conclusion consiste à dire que pour aller du marquage initial partiel de la place A au marquage final partiel de la place C, on a besoin d'un jeton dans la place E (enrichissement de marquage).De plus, on obtient forcément un jeton dans la place D (conséquence logique de ce scénario). On conclut aussi que le tir de t_2 suit nécessairement celui de t_1. Par conséquent, on produit un jeton dans D avant de produire un jeton dans C. Egalement, on a besoin d'un jeton dans la place E uniquement au moment de tirer t_2 et pas avant cet instant.

Maintenant que nous avons montré comment est menée la preuve du séquent incomplètement spécifié dans le cadre du raisonnement avant, nous traiterons ci-dessous le raisonnement arrière.

IV.B Raisonnement arrière

IV.B.1 Principe

Comme nous l'avons montré plus haut, le raisonnement arrière peut se ramener à un raisonnement avant sur le réseau de Petri inversé. Le raisonnement arrière permet de déterminer les scénarios qui mèneraient vers un marquage partiel donné. On part, par conséquent, du marquage partiel donné comme marquage initial et on explore tous les scénarios permettant de quitter cet état partiel sur le réseau inversé.

IV.B.2 Exemple

En reprenant l'exemple précédent de la figure 3.2, supposons que l'on cherche les scénarios permettant de mettre un jeton dans la place C en consommant un jeton dans la place A. Nous construisons tout d'abord le réseau de Petri inversé (par rapport à celui de la figure 3.2). Cela donne le réseau suivant :

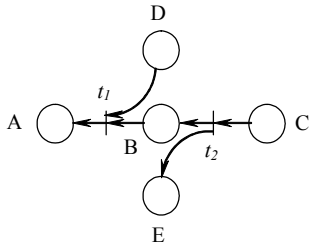


Figure 3.7. Réseau de Petri inversé

Nous voulons déterminer sur ce réseau les scénarios qui consomment un jeton dans C et qui produisent un jeton dans A. Le séquent de départ est le suivant : $C(I^1) \otimes \Gamma, T \vdash A(F^1) \otimes \Delta$. Le raisonnement avant dans ce cas particulier est facile à obtenir du fait de la parfaite symétrie entre les places A et C, les places D et E et les transitions t_1 et t_2. Il suffit par conséquent de permuter dans l'arbre de preuve obtenu au paragraphe précédent l'atome A et l'atome C, l'atome D et l'atome E et la transition t_1 et la transition t_2.

On obtient le graphe de précédence suivant :

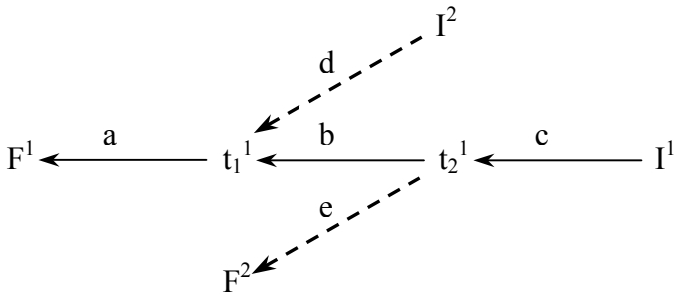


Figure 3.8. Graphe de précédence construit à la suite du raisonnement arrière

C'est le seul scénario répondant à la question. Pour retrouver les relations de causalité du réseau de départ, il faut inverser les sens des flèches et permuter les événements initiaux (I^k) et finaux (F^k). On retrouve ainsi le graphe de précédence de la figure 3.6.

Maintenant que nous avons introduit les différentes notions de raisonnement avant et raisonnement arrière, nous allons présenter comment celles-ci sont utilisées au sein de la méthode de recherche de scénarios.

V Méthode de recherche de scénarios redoutés

Le but de cette méthode est de caractériser les scénarios redoutés par des relations de causalité entre les événements menant vers un état redouté donné. Pendant la phase de conception des systèmes mécatroniques, les scénarios redoutés sont, en effet, inconnus du fait de la complexité inhérente à ces systèmes et à la multitude de modules en interaction, comme

nous l'avons vu dans le premier chapitre. Leur caractérisation au travers d'une analyse qualitative basée sur le modèle réseau de Petri du système permet de contourner les problèmes posés par une approche énumérative fondée sur le graphe des marquages accessibles (l'explosion combinatoire du nombre d'état et l'entrelacement). Les modèles réseaux de Petri représentent le comportement de ces systèmes dans les conditions de fonctionnement normal ainsi que leur comportement en cas de défaillance de leurs composants. Ils mettent également en lumière les changements de configurations des systèmes en fonction de l'évolution des variables continues des ressources matérielles disponibles (au sens composants partagés). Ce critère de disponibilité de ressources est primordial car il illustre les interactions possibles entre les différents modules du système. Ces interactions peuvent également provenir du partage d'une même variable continue par plusieurs modules. L'occurrence d'une défaillance dans un des composants du système pourrait affecter la variable partagée, ainsi, les conséquences de cette défaillance se propageraient au sein du système par l'intermédiaire de cette variable et causeraient la défaillance d'un autre composant. Pour bien prendre en compte ce type d'interaction dans notre modélisation, nous excluons toute interaction avec une variable continue ne figurant pas comme attribut sur un jeton du réseau. Cela suppose que si une variable partagée est commune entre plusieurs modules (modélisés chacun par un réseau de Petri), l'évolution de cette variable doit être représentée, dans le réseau global, par une unique place, à laquelle est associé un ensemble d'équations algébro-différentielles régissant l'évolution de cette variable.

En résumé, cette méthode de recherche de scénarios permet d'extraire et de rendre explicite les scénarios redoutés (sous forme d'ordres partiels) à partir d'un modèle (en réseau de Petri couplé avec des équations algébro-différentielles) agrégeant un ensemble de connaissances sur le fonctionnement du système (vue comme un ensemble de sous-systèmes en interaction) en présence et en absence de défaillances de ces composants. La recherche des causalités est basée uniquement sur le modèle réseau de Petri du système et non sur le modèle continu (les équations algébro-différentielles).

V.A Principe

Soit un modèle en réseau de Petri PTDS d'un système mécatronique. On cherche les scénarios menant à un état redouté caractérisé par un marquage partiel c'est à dire, dans le cas le plus simple, par le fait qu'une place particulière du réseau contienne un jeton. Cette place, notée E.P.R (pour Etat Partiel Redouté) peut correspondre au déclenchement d'une alarme suite au franchissement d'un seuil de sécurité par une variable continue du système. La figure 3.9 montre différents scénarios menant à un état partiel redouté. Les étoiles représentent des états partiels du système et les flèches représentent des enchaînements de relations de cause à effet menant de l'état partiel, origine de la flèche, à l'état partiel destination. Nous expliquons ci-dessous comment s'applique notre méthode de recherche de scénarios en s'appuyant sur la figure 3.9.

Partant de l'état partiel redouté E.P.R, nous commençons un raisonnement arrière permettant de construire les prédécesseurs immédiats de cet état partiel. Ici, nous trouvons deux états partiels : l'état partiel normal E.P.N.1 et l'état partiel dégradé E.P.D. L'état E.P.N.1 étant considéré comme état de fonctionnement normal, nous ne chercherons plus ces prédécesseurs car cela ne nous apportera pas plus d'informations du point de vue sûreté de fonctionnement. Quant à l'état partiel dégradé E.P.D, nous poursuivons le raisonnement arrière et la construction de ces prédécesseurs immédiats car il s'agit d'un état dégradé et non d'un état normal. Ceci nous donne les états partiels normaux E.P.N.2 et E.P.N.3. Etant des états de fonctionnement normal, construire leurs prédécesseurs ne nous apportera aucune

information pertinente du point de vue sûreté de fonctionnement. Le raisonnement arrière s'achève ici.

Qu'avons nous obtenu à ce stade de la recherche de scénarios ? Nous avons construit trois scénarios menant à l'état partiel redouté E.P.R : celui partant de l'état partiel E.P.N.1, celui reliant les état partiels E.P.N.2 et E.P.R en passant par l'état partiel dégradé E.P.D et enfin celui passant par ce même état mais reliant E.P.N.3 et l'état partiel redouté.

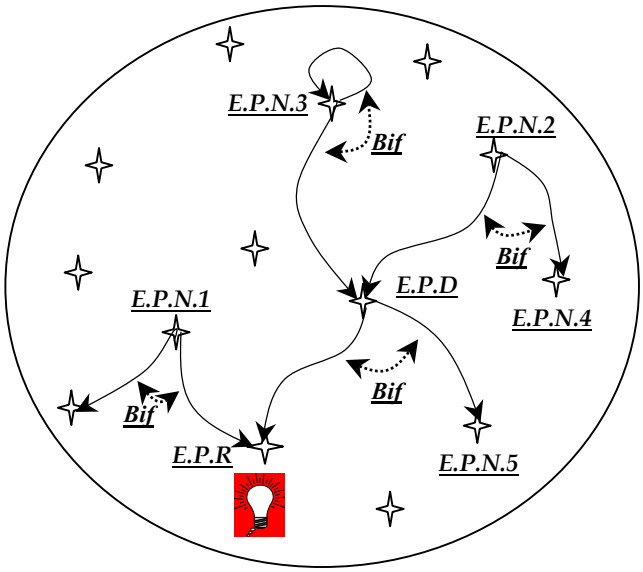


Figure 3.9. **Principe de la méthode de recherche de scénarios**

Pour résumer, nous avons donc trois scénarios menant vers l'état partiel redouté en question. Ces informations, récoltées à ce stade de la recherche, sont elles suffisantes quant à la connaissance d'occurrence de ces scénarios ? Nous connaissons effectivement les événements et/ou les changements d'états qui ont accompagné l'évolution du système vers l'état redouté, toutefois, nous n'avons obtenu aucune information sur les causes à l'origine de ces scénarios redoutés. Comment peut-on identifier ces causes ? Dans le cas des systèmes mécatroniques, l'évolution du système vers un état redouté s'explique généralement par l'échec d'une stratégie de reconfiguration. Cette dernière a pour but d'empêcher le système d'évoluer vers l'état redouté en assurant un fonctionnement dégradé ou en compensant complètement ce dysfonctionnement. Le système se trouverait ainsi dans un nouvel état de bon fonctionnement. La reconfiguration consiste donc à détourner l'évolution du système vers un nouvel état non redouté, c'est ce qu'on a appelé une ***bifurcation***. Son échec rend l'évolution vers l'état redouté inévitable si aucune autre reconfiguration n'a été prévue.

Une façon de connaître les causes de l'occurrence du scénario redouté est de suivre l'évolution du système à partir des états partiels normaux obtenus à la suite du raisonnement arrière, à savoir ici E.P.N.1, E.P.N.2 et E.P.N.3. Ces états sont appelés ***états partiels conditionneurs***. Pourquoi donc cette appellation ? Ces états représentent les derniers états de bon fonctionnement du système pendant son évolution jusqu'à l'état redouté. Le passage par ces états conditionne donc l'évolution vers l'état redouté. Afin de suivre cette évolution, nous commençons un raisonnement avant à partir de ces états.

Comme nous l'avons expliqué dans la section IV.A, le raisonnement avant consiste à chercher les conséquences logiques possibles à partir d'un état partiel donné. A partir de chacun de ces états partiels conditionneurs (par exemple ici E.P.N.2), on construit ses successeurs immédiats (E.P.D ou E.P.N.4). On arrête le raisonnement avant pour E.P.N.4 car

il s'agit d'un état de bon fonctionnement pour le système, poursuivre le raisonnement n'aura plus de sens du point de vue SdF. Quant à l'état partiel dégradé E.P.D, la construction des successeurs se poursuit, ce qui donne soit l'état partiel redouté (E.P.R) soit un état partiel de bon fonctionnement (E.P.N.5). Le raisonnement avant s'achève dans les deux cas.

Récapitulons. A l'issue de ce raisonnement avant à partir de l'état partiel normal E.P.N.2, nous avons identifié deux scénarios en plus du scénario redouté (le passage de E.P.N.2 vers E.P.R à travers E.P.D) : le premier est celui partant de E.P.N.2 et aboutissant à E.P.N.4 et le deuxième relie E.P.N.2 à E.P.N.5 en passant par E.P.D. Ces deux scénarios peuvent être considérés comme des tentatives de reconfiguration réussies du système lui évitant d'évoluer jusqu'à l'état redouté. Si le système ne peut évoluer selon ces deux scénarios, il ne peut qu'évoluer selon le scénario redouté. Nous considérons que l'impossibilité du système d'évoluer selon ces scénarios est une cause possible du scénario redouté. L'étude approfondie des raisons de cette impossibilité, impliquant la dynamique des variables continues, pourrait nous renseigner à propos des causes possibles de l'évolution dangereuse. Les causes d'occurrence du scénario redouté s'enrichissent petit à petit en étudiant les évolutions en conflit avec celle du scénario en question.

Pour une meilleure compréhension de ces notions de bifurcation et d'état conditionneur, nous allons les illustrer sur un exemple de réseau de Petri.

Considérons l'exemple de la régulation du niveau de liquide dans un réservoir. Ce dernier est rempli par une pompe et délivre continuellement du liquide en sortie. Quand le volume dans le réservoir dépasse le seuil de contrôle (V_l), la pompe est arrêtée et le volume commence à décroître. Quand le volume dépasse le seuil de sécurité ($V_s > V_l$), on déclenche une alarme correspondant au scénario redouté débordement. Le fonctionnement de ce réservoir peut être modélisé par le réseau de Petri de la figure 3.10. La place P_1 correspond à la phase de remplissage du réservoir. La place P_3 correspond à la phase de vidange. P_2 modélise la disponibilité de la pompe. Le franchissement de la transition t_1 correspond au dépassement de seuil V_l tandis que celui de t_2 correspond au franchissement du seuil V_s (c'est l'événement redouté) et au marquage de la place P_4 (état partiel redouté).

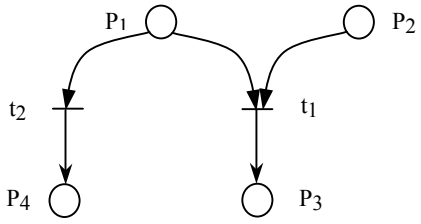


Figure 3.10. Exemple de réseau de Petri

On s'intéresse au scénario redouté ayant provoqué le débordement du réservoir. L'**état partiel redouté** en question correspond au marquage partiel de la place P_4. Cet état partiel est obtenu par le tir de la transition t_2 à partir du marquage partiel de la place P_1. Ce dernier étant un état partiel de bon fonctionnement (remplissage du réservoir), il est un **état partiel conditionneur**. A partir de cet état, il est possible d'atteindre soit l'état partiel redouté (par tir de t_2), soit celui correspondant au marquage partiel tel qu'il y a un jeton dans la place P_3 (par le tir de t_1 s'il y avait initialement un jeton dans P_2). On appelle **bifurcation** toute évolution du système lui permettant d'éviter le scénario redouté. Ici, s'agissant d'un conflit entre transitions (t_1 et t_2), la bifurcation correspond à l'évolution suite au tir de t_1.

Si nous revenons au schéma de la figure 3.10, le raisonnement arrière nous mène de l'état E.P.R aux états E.P.N.1 et E.P.D et de façon similaire, de l'état E.P.D le raisonnement arrière nous mène à E.P.N.2 et E.P.N.3. Afin de ne pas différencier inutilement ces deux raisonnements, nous avons choisi de nommer « **état cible** » l'état partiel de départ du raisonnement. Cet état cible peut correspondre à n'importe quel marquage partiel du réseau de Petri. A partir de ces états, nous appliquons la méthode de recherche de scénarios qui est basée sur 4 étapes.

V.B Les différentes étapes

Cette méthode permet de déterminer, de manière systématique et formelle, comment marquer et démarquer un ensemble de places correspondant à l'état cible. Elle est composée de 4 étapes. Les deux premières permettent de définir l'état cible ainsi que les états conditionneurs. Les deux dernières ont pour rôle de déterminer soit les composants ou sous modules impliqués dans les scénarios redoutés, pour la troisième étape, soit l'évolution du système à partir de l'état de ces composants trouvé à l'étape précédente, en ce qui concerne la dernière étape. Il s'agit respectivement de l'étape de raisonnement arrière et de l'étape de raisonnement avant. Nous décrivons ci-dessous ces différentes étapes.

V.B.1 Détermination des états nominaux

La première étape consiste à déterminer les places dont le marquage représente un état partiel de fonctionnement normal. Ces places nominales seront utilisées comme critère d'arrêt du raisonnement. Cette étape peut être réalisée de deux manières : soit en utilisant une connaissance a priori des états de bon fonctionnement du système, soit en effectuant une simulation de Monte Carlo du modèle sur une courte fenêtre temporelle pour déterminer la probabilité de marquage des places du réseau. Celles qui auront une probabilité de marquage non négligeable seront assimilées à des places normales.

V.B.2 Détermination des états cibles

La deuxième étape détermine l'état cible à étudier. Nous rappelons que cet état cible peut être soit un état partiel redouté, soit un autre état partiel ayant un lien de causalité direct ou indirect avec cet état redouté (par exemple une place qui représente la disponibilité d'une ressource pour assurer un fonctionnement dégradé évitant l'occurrence de l'événement redouté). La détermination des états partiels redoutés (déclenchement d'une alarme suite à un dépassement d'un seuil de sécurité par exemple) peut se faire grâce à une Analyse Préliminaire des Risques.

V.B.3 Raisonnement arrière

La troisième étape génère l'ensemble des chemins qui mènent vers l'état cible et identifie les différents sous-modules impliqués dans le scénario. On effectue un raisonnement arrière, basé sur le modèle RdP inversé. Dans ce réseau inversé, on prend comme marquage initial le seul état cible et l'on cherche de façon exhaustive tous les scénarios permettant de consommer le marquage initial et aboutissant à un marquage final uniquement formé de places associées au fonctionnement normal. Au cours de cette étape, on est en général amené à enrichir le marquage initial (ajouter des jetons dans certaines places comme cela a été expliqué en IV.A). Cela se fera chaque fois que, pour consommer un jeton dans une place non associée à un fonctionnement normal il faut franchir une transition non sensibilisée par un marquage accessible à partir du marquage initial non enrichi.

Les jetons ajoutés lors du processus d'enrichissement du marquage correspondent à des états partiels qui sont des conséquences logiques des scénarios redoutés. Ils seront donc nécessairement observés lors de l'évolution du système vers l'état redouté. En inversant les scénarios obtenus lors de cette étape, nous aurons les actions menant d'un état normal à l'état partiel redouté avec l'ordre partiel que l'on doit respecter.

V.B.4 Raisonnement avant

La dernière étape de la méthode consiste à construire un raisonnement à partir du modèle RdP initial en partant de chaque état conditionneur déterminé à l'étape précédente. C'est l'étape de raisonnement avant. Cela a pour objectif de localiser les bifurcations entre le comportement redouté et le fonctionnement normal du système ainsi que les conditions (de marquage de certaines places du réseau) impliquées dans ces bifurcations.

V.C Exemple d'application de la méthode

Reprenons l'exemple de la figure 3.10. Sans détailler ce qui concerne la logique Linéaire, nous allons appliquer la méthode de recherche de scénarios. Nous voulons déterminer tous les scénarios menant au débordement du réservoir. L'état partiel redouté en question correspond au marquage partiel de la place P_4. Quant aux états de fonctionnement normal, ce sont ceux représentés par le marquage de P_1 ou de P_2 ou de P_3.

L'étape de raisonnement arrière peut maintenant commencer. Dans la section IV.B, nous avons défini ce type de raisonnement comme un raisonnement avant opérant sur le réseau de Petri inversé $\boldsymbol{R}^{-1}$ du modèle du système étudié. Un jeton est donc mis dans la place P_4. Comment peut-on quitter cet état partiel ? Seul le tir de la transition t_2 (dans le réseau de Petri inversé) le permet. Une fois cette transition tirée, cela produit un jeton dans la place P_1. Le raisonnement arrière s'arrête puisqu'il s'agit d'un état partiel de bon fonctionnement. C'est un état partiel conditionneur.

A partir de cet état, nous menons un raisonnement avant qui a pour but de déterminer les différentes évolutions possibles du système. A partir de ce marquage partiel, le réseau de Petri peut évoluer de deux façons possibles : l'évolution lui permettant d'aller vers l'état partiel redouté (par le tir de t_2) et celle correspondant au franchissement de t_1 à condition d'avoir enrichi le marquage de la place P_2 (un jeton dans P_2). Le raisonnement s'achève à ce stade.

Synthétisons ce que l'on a obtenu suite à l'application de la méthode. L'étape de raisonnement arrière a montré qu'il est possible d'atteindre l'état redouté en partant du marquage partiel tel qu'il y a un jeton dans P_1 et en franchissant t_2. L'étape de raisonnement avant a permis de mettre en évidence une bifurcation possible du scénario redouté. Cette bifurcation correspond au tir de la transition t_1 à condition de disposer au préalable d'un jeton dans P_2 (enrichissement de marquage). L'absence d'un jeton dans la place P_2 favorise, par conséquent, l'occurrence de l'événement redouté débordement. En fait, le seuil de contrôle V_l étant inférieur au seuil de sécurité V_s, si t_1 et t_2 sont toutes les deux sensibilisées, t_1 sera toujours franchie avant t_2. C'est donc l'absence de jeton dans P_2 pendant un certain intervalle de temps quand P_1 contient un jeton qui est la cause indirecte du scénario redouté.

Afin de mieux cerner les conditions d'occurrence du scénario redouté, il est donc indispensable d'étudier les conditions de tir de t_1, à savoir la présence simultanée d'un jeton dans P_1 et d'un jeton dans P_2 (s'il y a des transitions en amont des places P_1 et P_2). On voit donc ici naître l'idée de réitérer l'application de la méthode de recherche de scénarios à partir

du nouvel état partiel initial tel qu'il y a un jeton dans chacune des places P_1 et P_2. C'est un nouvel état cible. C'est le principe que nous détaillerons ci-dessous.

V.D La recherche des scénarios est un processus itératif

Comme nous venons de l'expliquer, la méthode de recherche de scénarios est basée sur un processus itératif qui a pour but de récolter, à chaque itération, des morceaux d'information (correspondant à des sous scénarios) du scénario redouté global. L'explication complète des conditions dans lesquelles est apparu le scénario sera obtenue en combinant ces sous-scénarios.

Ainsi, la méthode de recherche de scénarios redoutés est composée de plusieurs itérations, chacune basée sur les quatre étapes présentées à la section précédente. Chaque itération i+1 démarre à partir d'un état conditionneur parmi ceux donnés par l'itération i plus tous les états partiels correspondant à un enrichissement de marquage pendant cette même itération i. L'itération i+1 permet de savoir comment on peut atteindre ces états et génère à son tour de nouveaux états conditionneurs ainsi que des états correspondant à des enrichissements de marquage.

Ce processus itératif se termine quand l'utilisateur juge suffisante la connaissance qu'il a récoltée à propos du scénario redouté en question.

Afin de faciliter le travail de l'utilisateur, nous avons élaboré un algorithme qui permet de rendre automatique les étapes de raisonnement arrière et avant de la méthode de recherche de scénarios.

VI Algorithme pour la recherche de scénarios

Comme nous l'avons évoqué dans la section IV.B, le raisonnement arrière est basé sur un raisonnement avant opérant sur le réseau de Petri inversé. Ceci nous a permis de développer un unique algorithme pour les deux types de raisonnement. Nous le présentons dans le cadre du raisonnement avant. Avant de présenter les structures de données utilisées ainsi que l'algorithme en question, nous approfondirons le processus d'enrichissement de marquage qui est primordial pour notre approche. Nous allons montrer qu'il doit suivre certaines restrictions.

VI.A Enrichissement du marquage

L'enrichissement de marquage permet de compléter l'information sur le scénario redouté en ajoutant des hypothèses sur les états des composants (des jetons dans des places non marquées) ayant un lien avec ce scénario. Comme nous l'avons vu dans le paragraphe IV, l'enrichissement de marquage a lieu au cours du raisonnement avant (ou arrière) pendant la construction des successeurs (ou des prédécesseurs). Parfois, pour construire un successeur immédiat d'un état partiel donné (ce qui revient à franchir une transition donnée dans le réseau de Petri), il est indispensable d'ajouter des jetons dans des places non marquées pour franchir cette transition. L'ajout de ces jetons peut parfois causer des problèmes de cohérence. En effet, partant du principe que l'enrichissement de marquage consiste à supposer qu'un composant est dans un état donné, l'enrichissement de marquage peut amener à supposer qu'un composant est dans deux états différents en même temps, ce qui est incohérent. Ainsi, il faut se doter des mécanismes nécessaires pour s'assurer de la cohérence de l'enrichissement, et ce, à chaque pas du raisonnement.

Avant de présenter les deux mécanismes de contrôle de la cohérence pour l'enrichissement de marquage, nous introduirons tout d'abord la notion de transition potentiellement franchissable.

VI.A.1 Définition

Soit ***R*** un réseau de Petri marqué. Une **transition potentiellement franchissable** est une transition dont au moins une place amont est marquée (elle possède au moins un jeton) et au moins une place amont ne l'est pas (il lui manque au moins un jeton).

Reprenons le réseau de Petri de la figure 3.2 avec le marquage suivant : un jeton dans la place A et un autre dans la place E (Figure 3.11). La transition t_2 est une transition potentiellement franchissable (B, place amont de t_2, n'est pas marquée mais E l'est) alors que t_1 est une transition franchissable (son unique place amont A est marquée).

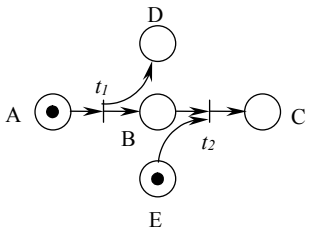


Figure 3.11. **Exemple de réseau de Petri marqué**

Le processus d'enrichissement de marquage consiste à ajouter les jetons manquants à la transition potentiellement franchissable de sorte qu'elle devienne franchissable. Sur l'exemple précédent, ceci revient à ajouter un jeton dans la place B. Ainsi t_2 devient franchissable. Mais si A et B représentent deux configurations différentes d'un unique composant, l'hypothèse consistant à mettre un jeton dans B pour rendre t_2 franchissable avant le franchissement de t_1 ne pourra jamais être vérifiée. Le composant ne peut se trouver dans deux configurations différentes au même instant. Il est inutile de construire le scénario découlant de cet enrichissement car, ne correspondant pas à la réalité du système physique, il n'apportera rien au concepteur. Il faut donc, chaque fois que cela est possible, interdire les enrichissements de marquage contradictoires avec la structure du système.

VI.A.2 Mécanismes de contrôle de la cohérence

Nous distinguons deux types de mécanismes de contrôle de la cohérence de l'enrichissement du marquage : le contrôle par calcul des invariants de place et celui dit par priorité.

VI.A.2.1Contrôle par calcul d'invariants de place

Ce mécanisme passe par le calcul des invariants de place [Valette 92] du réseau. Il a pour but de vérifier que l'ajout d'un ou plusieurs jetons dans une ou plusieurs places du réseau ne viole pas un invariant de place. Si c'est le cas, l'enrichissement de marquage correspondant n'est pas autorisé.

Les invariants sont calculés une fois pour toute, en prenant pour marquage initial celui qui correspond à l'état initial de tous les composants du système. Nous ne considérons que les invariants positifs. Pendant le raisonnement, nous ne considérons qu'un sous-ensemble de composants, ceux qui sont nécessairement impliqués dans le scénario étudié. La valeur

obtenue pour le marquage correspondant à l'étape courante du raisonnement doit donc toujours être inférieure ou égale aux invariants. Elle sera inférieure si certains composants du système impliqués dans l'invariant ne sont pas nécessaires au scénario étudié. Si la valeur est supérieure, cela veut dire qu'il y a incohérence. Il serait en effet impossible d'associer un état du système complet à l'étape courante du raisonnement en ajoutant des jetons. Les invariants considérés étant positifs, leur valeur ne peut décroître si l'on ajoute des jetons.

Revenons à l'exemple de la figure 3.11. Un invariant de place pour le marquage initial (un jeton dans A et un jeton dans E supposés correspondre à l'état initial du système physique) est le suivant : $M(A)+M(B)+M(C)=1$. Ceci se lit de la manière suivante : le nombre de jetons dans la place A plus le nombre de jetons dans la place B plus le nombre de jetons dans C doit être égal à 1 pour tous les marquages accessibles à partir du marquage initial. Si l'on doit enrichir le marquage de la place B, ceci violerait l'invariant de place car on aurait le résultat suivant : $\underbrace{M(A)}_{1}+\underbrace{M(B)}_{1}+\underbrace{M(C)}_{0}=2>1$. On ne peut, en effet, avoir aucun marquage accessible, à partir du marquage initial donné, tel qu'il y a un jeton dans A et un jeton dans B. Ce résultat met en lumière le fait que la production d'un jeton dans la place B est liée à la présence d'un jeton dans la place A.

Si l'on franchit t_1 en premier, on produit un jeton dans B et t_2 devient franchissable. Il paraît donc intéressant de retarder l'enrichissement de marquage de t_2 en espérant que le tir des transitions franchissables produira les jetons manquants pour la franchir. C'est le principe du contrôle dit par priorité.

VI.A.2.2Contrôle par priorité

Comme nous venons de l'évoquer, le but de ce type de contrôle est de franchir toutes les transitions franchissables par rapport à la liste des jetons de l'étape courante avant d'envisager un enrichissement de marquage. Les transitions franchissables sont en effet prioritaires, dans notre algorithme, devant les transitions potentiellement franchissables, d'où l'appellation contrôle par priorité. Nous ne rajoutons donc des jetons dans des places non marquées que lorsque cela est absolument nécessaire pour poursuivre le raisonnement (il ne reste aucune transition franchissable).

Maintenant que nous avons présenté les mécanismes de contrôle de la cohérence de l'enrichissement de marquage, nous allons expliquer ci-dessous les différentes structures de données que l'algorithme de recherche de scénarios utilise.

VI.B Structures de données

L'algorithme utilise trois sortes de structures de données : des données d'entrée, des données de sortie et des données internes. Commençons par présenter la structure des données d'entrée.

VI.B.1 Données d'entrée

Elles sont constituées de la liste des jetons initiaux (notée L_i) et de la liste des jetons normaux (notée L_n). La première liste est formée de l'ensemble des états partiels de départ à partir desquels s'effectue la recherche de scénarios. A la première itération de la méthode de recherche de scénarios, cette liste contiendra dans un raisonnement arrière l'état cible, par exemple l'état partiel redouté. Quant à la liste des jetons normaux, elle contient l'ensemble des états partiels correspondant à des fonctionnements normaux. Cette liste permet à

l'algorithme de ne pas pousser la recherche au delà de ces états partiels car cela n'apporterait pas d'informations supplémentaires pertinentes. Ceci permet de ne pas explorer tous les scénarios possibles et d'éviter le problème de l'explosion combinatoire du nombre de scénarios possibles. Passons maintenant aux données de sortie de l'algorithme.

VI.B.2 Données de sortie

Résultats de l'algorithme, celles-ci contiennent tous les ordres partiels correspondant aux différents scénarios menant à l'état partiel initial (la liste des jetons L_i) dans le cas d'un raisonnement arrière. Chaque ordre partiel est donné sous la forme d'un triplet de la forme (E, A, L_e) avec E l'ensemble des instances de tirs de transitions, A est l'ensemble des arcs reliant des éléments de E, et L_e est la liste des jetons créés suite à un enrichissement de marquage.

VI.B.3 Données internes

Les données internes manipulées par l'algorithme sont nombreuses. Nous détaillons ci-dessous les plus importantes :

- Chaque jeton est représenté sous la forme d'un couple (e, p) où e est l'événement qui a produit ce jeton et p est la place le contenant. Cette notation sert pour la construction de l'ensemble A des ordres partiels comme nous le verrons par la suite.

- La liste courante, notée L_c, est une donnée interne. Elle représente l'étape courante (voir chapitre 2, paragraphe III.D.3) qui est un ensemble de jetons. Cette liste est mise à jour par l'algorithme à chaque franchissement de transition. Cela consiste à enlever les jetons consommés par la transition et à ajouter ceux qui sont produits.

- La liste des transitions interdites, notée L_{int}, contient un ensemble de transitions à ne pas franchir à partir d'une étape courante donnée. Cette liste nous est utile pour gérer les conflits de transitions. Dès qu'une transition, en conflit avec une autre, est franchie, elle est ajoutée à cette liste pour ne pas la franchir une deuxième fois. En effet, deux transitions en conflit génèrent deux scénarios différents (un scénario par transition). Une fois l'un des scénarios construit, nous re-considérons l'étape courante pour construire l'autre scénario.

- Le contexte, noté C, est une liste de quintuplets de la forme (L_c, L_{int}, E, A, L_e) où L_c est la liste courante, L_{int} est la liste des transitions interdites, E et A représentent l'ordre partiel dérivé jusqu'alors et L_e est la liste des jetons enrichis. C contient toute l'information nécessaire pour construire un nouvel ordre partiel suite à la résolution d'un conflit de transitions. Il est mémorisé à chaque fois qu'un conflit est résolu en choisissant une transition à tirer. Cette dernière est mise dans la liste des transitions interdites et on stocke un nouvel élément dans C pour dériver le nouvel ordre partiel.

Outre ces données internes, il en existe d'autres qui concernent plus particulièrement l'ensemble des transitions franchissables et potentiellement franchissables à partir de la liste courante L_c. En effet, comme nous l'avons évoqué dans le paragraphe IV.A.2.2, les transitions franchissables sont prioritaires par rapport à celles qui sont potentiellement franchissables. Si l'on rajoute à cela les conflits entre transitions, on obtient une répartition de l'ensemble des transitions franchissables et potentiellement franchissables en quatre sous-ensembles disjoints :

- L'ensemble des transitions franchissables sans conflit, noté Tfsc,
- L'ensemble des transitions potentiellement franchissables sans conflit, noté Tpfsc,
- L'ensemble des transitions franchissables en conflit uniquement avec des transitions franchissables, noté Tfcf,
- L'ensemble des transitions franchissables en conflit avec au moins une transition potentiellement franchissable, noté Tfcpf,
- L'ensemble des transitions potentiellement franchissables en conflit aussi bien avec des transitions franchissables qu'avec des transitions potentiellement franchissables, noté Tpfc.

Après avoir présenté la structure des données utilisées par l'algorithme, nous allons maintenant détailler quelques procédures afin de faciliter la compréhension de ce dernier.

VI.C Quelques procédures

VI.C.1 Tirer Transition

En plus de la mise à jour de la liste courante suite au tir de la transition en question, ce qui revient à enlever les jetons consommés et ajouter les jetons produits dans la liste courante, cette procédure sert également à mémoriser les événements (l'instance de franchissement de cette transition) dans E ainsi que tous les arcs correspondant à une relation de précédence entre deux événements dans A. Un arc relie l'événement qui a produit les jetons consommés à l'instance de franchissement de la transition qui va consommer ces jetons. Cette procédure de tir de transition est la suivante :

Tirer Transition (t_k) *:*

- *Ajouter t_k dans E ;*
- *Pour chaque jeton (t_i,p) nécessaire pour franchir t_k enlever (t_i,p) de la liste L_c et ajouter (t_i,t_k) dans A ;*
- *Pour chaque place de sortie p_s de t_k, ajouter un jeton de la forme (t_k, p_s) dans L_c.*

VI.C.2 Enrichir Marquage

Cette procédure a pour but de contrôler la cohérence de l'enrichissement de marquage et d'autoriser cet enrichissement uniquement quand cela est cohérent par rapport aux invariants de places. On distingue deux types de procédures pour l'enrichissement : la première procédure, appelée ***Enrichir Marquage1***, opère sur une transition appartenant à l'ensemble Tfcpf et enrichit le marquage de toutes les transitions potentiellement franchissables en conflit avec celle-ci, et ce de la manière suivante :

Enrichir Marquage1 *(t_k) :*

L, variable interne de la procédure, est une liste de jetons initialement vide.

- *Pour chaque chaque transition tj, potentiellement franchissable en conflit avec tk, et pour chaque place pl en amont de tj, ajouter un jeton (ek, pl) dans la liste L ;*
- *Si l'enrichissement de marquage est incohérent avec les invariants de places, alors effacer les jetons rajoutés de la liste L ;*
- *Ajouter les atomes de L dans les listes Lc et Le.*

La deuxième procédure, notée ***Enrichir Marquage2***, opère sur les transitions potentiellement franchissables et permet d'enrichir leur marquage :

Enrichir Marquage2 *(t_k) :*
L est une liste de jetons initialement vide.

- *Pour chaque place p_l en amont de t_k, ajouter un jeton (e_k, p_l) dans la liste L ;*
- *Si l'enrichissement de marquage est incohérent avec les invariants de places, alors effacer les jetons rajoutés de la liste L ;*
- *Ajouter les atomes de L dans les listes L_c et L_e.*

VI.C.3 Mémoriser Contexte

Comme nous l'avons vu dans le chapitre précédent, paragraphe III.G, à chaque fois qu'un conflit est rencontré pendant la construction d'un ordre partiel, ce dernier est scindé en autant d'ordres partiels différents (donnés par des arbres de preuve) que de transitions impliquées dans le conflit.

Par rapport à une transition impliquée dans un conflit, cette procédure permet de mémoriser toute l'information nécessaire pour construire un autre ordre partiel correspondant au tir d'une autre transition en conflit avec cette dernière. Ainsi nous avons :

Mémoriser Contexte *(t_k) :*

- *Ajouter la transition t_k à la liste des transitions interdites L_{int} ;*
- *Ajouter un nouveau quintuplet (L_c, L_{int}, E, A, L_e) au contexte C ;*
- *Effacer le contenu de L_{int}.*

Après avoir présenté la structure des données ainsi que les procédures, nous expliciterons ci-dessous l'algorithme de recherche de scénarios.

VI.D Algorithme

L'algorithme de recherche de scénarios comporte différentes étapes qui s'agencent comme suit :

Pas initial :
//Pour pouvoir construire le premier ordre partiel, ce pas initialise le contexte C avec un quintuplet (Lc, Lint, E, A, Le) tel que : $L_c = L_i$, L_{int} et L_e sont vides, E = {I}, A est vide et l'entier ***Inc*** est égal à 1.//
C ← ($L_c = L_i$, $L_{int} = \{\phi\}$, E = {I}, A = $\{\phi\}$, $L_e = \{\phi\}$)
Inc = 1

Pas 1 :
Si C = $\{\phi\}$ alors aller à Pas final
Sinon :
Mémoriser le premier élément de C dans (L_c, L_{int}, E, A, L_e) ;
Effacer cet élément de C
Aller à Pas 2

Pas 2 :
Générer à partir de (L_c, L_{int}, E, A, L_e) toutes les transitions franchissables et potentiellement franchissables ;
Effacer de ces listes :

- les éléments de E (pour éviter les boucles infinies)

- les transitions dans L_{int} et toutes celles qui leur sont parallèles (pour éviter de construire plus d'une fois un même ordre partiel)

Générer les listes suivantes : Tfsc, Tpfsc, Tfcf, Tfcpf et Tpfc
Aller à Pas 3

Pas 3 :
//Ce pas concerne le critère d'arrêt de la construction d'un ordre partiel.//
Si L_c contient uniquement des jetons appartenant à L_n et qui ne sont pas des jetons initiaux ***ou*** les listes Tfsc, Tpfsc, Tfcf, Tfcpf et Tpfc sont toutes vides alors aller à Pas 9
Sinon aller à Pas 4 ;

Pas 4 :
//Les transitions franchissables qui ne sont pas en conflit avec une autre transition sont tirées en priorité car aucune décision n'est à prendre//
Si Tfsc = $\{\phi\}$ alors aller à Pas 5 ;
Sinon :

- Soit t_k la première transition de Tfsc ;
- **Tirer Transition** (t_k) ;
- Aller à Pas 2 ;

Pas 5 :
//Ce pas résout les conflits de transitions en en tirant une et mémorise l'information nécessaire pour la construction des autres ordres partiels relatifs aux tirs des autres transitions impliqués dans le conflit//
Si Tfcf = $\{\phi\}$ alors aller à Pas 6 ;
Sinon :

- Soit t_k la première transition de Tfcf ;
- **Mémoriser Contexte** (t_k) ;
- **Tirer Transition** (t_k) ;
- Aller à Pas 2 ;

Pas 6 :
//Ce pas concerne l'enrichissement de marquage de toutes les transitions potentiellement franchissables en conflit avec une transition franchissable donnée. Cela a lieu quand toute les décisions ne nécessitant pas un enrichissement ont été prises//
Si Tfcpf = $\{\phi\}$ alors aller à Pas 7 ;
Sinon :

- Soit t_k la première transition de Tfcpf ;
- **Enrichir Marquage1** (t_k) ;
- Si t_k est maintenant en conflit avec au moins une transition franchissable alors **Mémoriser Contexte** (t_k) ;
- **Tirer Transition** (t_k) ;
- Aller à Pas 2 ;

Pas 7 :
//Dans ce pas, nous enrichissons le marquage d'une transition potentiellement franchissable et en conflit avec d'autres transitions. Ensuite, nous mémorisons le contexte et nous tirons cette transition//
Si Tpfc = $\{\phi\}$ alors aller à Pas 8 ;

Sinon :

- Soit t_k la première transition de Tpfc ;
- **Enrichir Marquage2** (t_k) ;
- **Mémoriser Contexte** (t_k) ;
- **Tirer Transition** (t_k) ;
- Aller à Pas 2 ;

Pas 8 :

//Dans ce pas, nous enrichissons le marquage d'une transition potentiellement franchissable et en conflit avec aucune autre transition et nous tirons ensuite cette transition//

Si Tpfsc = { ϕ } alors aller à Pas 9 ;

Sinon :

- Soit t_k la première transition de Tpfsc ;
- **Enrichir Marquage2** (t_k) ;
- **Tirer Transition** (t_k) ;
- Aller à Pas 2 ;

Pas 9 :

//Nous mémorisons l'ordre partiel construit et nous revenons au pas 1//

Pour chaque jeton (t_i, p) de la liste L_c ajouter (t_i, f) dans l'ensemble A ; f étant l'événement fin.

Mémoriser l'ordre partiel construit numéro ***Inc*** tel que : E(***Inc***) = E, A(***Inc***) = A et L_e(***Inc***) = L_e ;

Incrémenter ***Inc*** ;

Aller à Pas 1 ;

Pas final :

//C'est la fin de l'algorithme//

Nous donnons en annexe l'organigramme de l'algorithme que nous venons de développer. Nous allons maintenant appliquer la méthode de recherche de scénarios ainsi que cet algorithme sur un cas d'étude élémentaire.

VI.E Application sur un cas d'étude

Nous allons tout d'abord présenter le cas d'étude. Nous appliquerons, ensuite, la méthode de recherche de scénarios et en particulier l'algorithme que nous avons développé ci-dessus sur ce cas d'étude.

VI.E.1 Présentation du cas d'étude

Le réseau de Petri de la figure 3.12 représente un équipement qui peut être dans trois états : à l'arrêt (quand la place OFF est marquée), en état de fonctionnement nominal (quand la place N est marquée) ou en état de fonctionnement dangereux (quand la place D est marquée). Le démarrage de l'équipement est modélisé par le tir de la transition t_1 tandis que son évolution vers l'état de fonctionnement dangereux est modélisée par le tir de t_2. Un actionneur permet de maintenir l'équipement dans un état de fonctionnement nominal (tir de la transition t_3). Cet actionneur peut être dans deux états : un état de fonctionnement nominal (place Ac marquée) et un état où il est défaillant (place AF marquée). L'occurrence de cette défaillance est modélisée par le tir de la transition t_4.

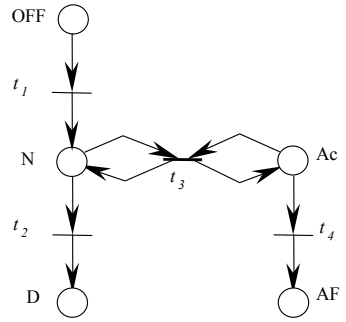


Figure 3.12. Modèle réseau de Petri du cas d'étude

Dans le paragraphe suivant, nous allons appliquer la méthode de recherche de scénarios pour déterminer les scénarios menant à l'état partiel redouté tel qu'il y a un jeton dans la place D.

VI.E.2 Application de la méthode

L'application de la méthode de recherche de scénarios nécessite la détermination de l'état cible et des états nominaux (section V.B). L'état cible est celui correspondant au marquage partiel de la place D. Le marquage partiel de la place N, celui de la place OFF et celui de Ac représentent des états nominaux.

VI.E.2.1Raisonnement arrière

Comme nous l'avons vu en V.B.3, le raisonnement arrière opère sur le réseau de Petri inversé du cas d'étude (figure 3.13).

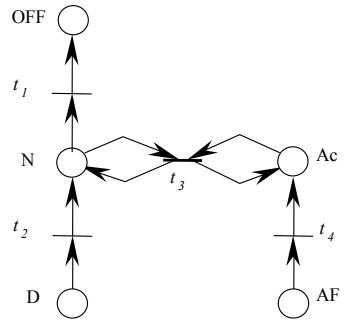


Figure 3.13. Modèle réseau de Petri inversé du cas d'étude

Commençons donc l'application de l'algorithme :

***Pas initial* :**
Nous avons Lc = Li = {(i, D)}, Lint = {ϕ}, Le = {ϕ}, E = {i}, A = {ϕ}, ***Inc*** = 1 et C = {(Lc, Lint, E, A, Le)}.

***Pas 1* :**
C n'est pas vide d'où (Lc, Lint, E, A, Le) = ({i, D}, {ϕ}, {i}, ({i, D}, {ϕ}).
C devient vide.
Aller à Pas 2

***Pas 2* :**
La seule transition franchissable est t_2. Elle n'est en conflit avec aucune autre transition. Aucune transition n'est potentiellement franchissable d'où :

Tfsc = {t_2}, Tfcf = {ϕ}, Tfcpf = {ϕ}, Tpfc = {ϕ}.
Aller à Pas 3

Pas 3 :
Lc ne contient pas uniquement des jetons appartenant à Ln (D n'appartient pas à Ln) d'où aller à Pas 4 ;

Pas 4 :
Tfsc (= {t_2}) n'est pas vide d'où :
Soit tk = t_2
Tirer Transition (t_2) :

- E = {i, t_2}
- Lc = {ϕ} et A = {(i, t_2)}
- Lc = {(t_2, N)}

Aller à Pas 2.

Pas 2 :
Suite à l'étape précédente, la liste courante contient un jeton dans la place N. A partir de ce marquage partiel, il y a une transition franchissable (t_1) en conflit avec une transition potentiellement franchissable (t_3) d'où :
Tfsc = {ϕ}, Tfcf = {ϕ}, Tfcpf = {t_1}, Tpfc = {t_3}.

Pas 3 :
Le critère d'arrêt est satisfait : Lc (= {t_2, N}) ne contient que des jetons de Ln alors aller à Pas 9.

Pas 9 :
L'ordre partiel construit est défini par : E_1 = {i, t_2}, A_1 = {(i, t_2), (t_2, f)} et Le = {ϕ}. Nous avons déterminé un état conditionneur qui est celui correspondant au marquage partiel de la place N.
Aller à Pas 1

Pas 1 :
Le contexte C est vide. C'est donc la fin de l'algorithme.

Suite à l'application de l'algorithme pour le raisonnement arrière, nous avons obtenu un sous-scénario menant à l'état cible (c'est celui correspondant à l'ordre partiel du pas 9) ainsi qu'un état conditionneur (le marquage partiel de la place N). Cet état sera le point de départ de l'étape suivante de la méthode : le raisonnement avant.

VI.E.2.2Raisonnement avant

Ce raisonnement opère sur le réseau de Petri initial (celui de la figure 3.12) muni du marquage partiel initial : un jeton dans la place N. Commençons l'application de l'algorithme : nous avons Li = {(i,N)} = Lc.

Pas 2 :
Tfsc = {ϕ}, Tfcf = {ϕ}, Tfcpf = {t_2}, Tpfc = {t_3}.

Pas 6 :
Tfcpf (= {t_2}) n'est pas vide.

Soit $t_k = t_2$ (la seule transition dans l'ensemble Tfcpf)

Enrichir Marquage1 (t_2) :

Initialement L = { ϕ }

- La seule transition en conflit avec t_2 est t_3. Ac est la seule place en amont de t_3 non marqué. Nous y rajoutons un jeton (e_2, Ac). Ainsi, L = {(e_2, Ac)}.
- Cet enrichissement de marquage est cohérent car il respecte l'invariant de place : M(Ac) + M(AF) = 1 (place AF n'est pas marquée).
- Lc = {(i, N), (e_2, Ac)}, Le = {(e_2, Ac)}

Mémoriser Contexte (t_2) :

Lint = { ϕ }

- Ajouter t_2 à Lint : Lint = {t_2}
- C = (Lc = {(i,N), (e_2, Ac)}, Lint = {t_2}, E = {i}, A = { ϕ }, Le = {(e_2, Ac)}).
- Effacer le contenu de Lint : Lint = { ϕ }

Tirer Transition (t_2) :

- E = {i, t_2}
- Lc = {(e_2, Ac)}. A = {(i, t_2)}
- Lc = {(t_2, D), (e_2, Ac)}

Aller à Pas 2

Pas 2 :

Nous avons : Lc = {(t_2, D), (e_2, Ac)}, Lint = { ϕ }, E = {i, t_2}, A = {(i, t_2)} et Le = {t_2}.

Nous en déduisons que : Tfwc = { ϕ }, Tfcf = { ϕ }, Tfcpf = {t_4}, Tpfc = {t_3}.

Aller à Pas 6

Pas 6 :

Tfcpf (= {t_4}) n'est pas vide. Ainsi nous avons :

Soit tk = t_4.

Enrichir Marquage1 (t_4) :

Initialement L = { ϕ }

- L'unique transition en conflit avec t_4 est t_3. N étant l'unique place non marquée en amont de t_3, nous ajoutons dans la liste L un jeton (e_2, N) : L = {(e_2, N)}.
- Cet enrichissement de marquage (l'ajout d'un jeton dans N) n'est pas cohérent avec l'invariant de place : M(OFF) + M(N) + M(D) = 1 car la place D contient déjà un jeton. Nous effaçons par conséquent le jeton enrichi de L, qui redevient vide : L = { ϕ }
- Lc est toujours égale à {(t_2, D), (e_2, Ac)}
- Le = {(e_2, Ac)}
- Comme t_3 n'est pas franchissable, on ne mémorise pas le contexte.

Tirer Transition (t_4) :

- E = {i, t_2, t_4}
- Lc = {(t_2, D)}, A = {(i, t_2), (e_2, t_4)}
- Lc = {(t_2, D), (t_4, AF)}.

Aller à Pas 2

Pas 2 :

Tous les ensembles de transitions sont vides car aucune transition n'est franchissable ou potentiellement franchissable (le marquage obtenu est un état puit : un jeton dans D et un jeton dans AF).

Pas 3 :

Le critère d'arrêt est satisfait. Aller à Pas 9.

Pas 9 :

L'ordre partiel construit est défini par : $E_1 = \{i, t_2, t_4\}$, $A_1 = \{(i, t_2), (t_2, f), (e_2, t_4), (t_4, f)\}$ et $Le = \{(e_2, Ac)\}$.
Aller à Pas 1

Pas 1 :

$C = (Lc = \{(i,N), (e_2, Ac)\}$, $Lint = \{t_2\}$, $E = \{i, e_2\}$, $A = \{\}$, $Le = \{(e_2, Ac)\}$. C devient vide.

Pas 2 :

- Les transitions franchissables pour la liste courante Lc, sont t_2, t_3 et t_4.
- Nous enlevons t_2 de cet ensemble car elle appartient à Lint.
- De même, t_4 est aussi enlevée de l'ensemble car elle est parallèle à t_2.

Nous obtenons donc : $Tfsc = \{t_3\}$, $Tfcf = \{\phi\}$, $Tfcpf = \{\phi\}$, $Tpfc = \{\phi\}$.
Le critère d'arrêt n'est pas satisfait. Nous allons donc à l'étape 4.

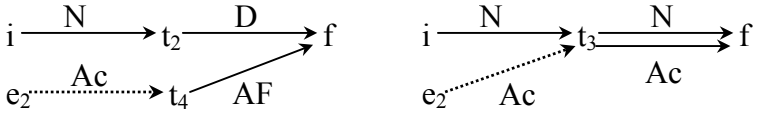
Pas 4 :

La transition t_3 est tirée et elle est ajoutée dans la liste des transitions interdites. Elle ne sera plus tirée. Par conséquent, l'algorithme se termine car toutes les transitions franchissables ou potentiellement franchissables sont dans Lint.

Finalement, on obtient l'ordre partiel suivant : $E_2 = \{i, e_2, t_3, f\}$, $A_2 = \{(i, t3), (e_2, t_3), (t_3, f), (t_3, f)\}$ et $Le = \{(e_2, Ac)\}$.

VI.E.2.3 Synthèse

En partant de l'état conditionneur donné par le raisonnement arrière, nous avons obtenu, suite à l'étape de raisonnement avant, deux ordres partiels possibles. Ces deux ordres partiels résultent du conflit de transitions entre t_2 et t_3. Ils sont représentés ci-dessous :



Premier ordre partiel — Deuxième ordre partiel

Figure 3.14. Ordres partiels

Le premier ordre partiel représente le scénario menant à l'état partiel redouté et le second représente une bifurcation possible de ce scénario. Nous remarquons également qu'il y a eu un enrichissement du marquage durant le raisonnement avant. En effet, celui-ci consiste à ajouter une hypothèse supplémentaire : l'actionneur Ac est dans un état de bon fonctionnement. On évite ainsi le comportement redouté par le tir de t_3 (deuxième ordre partiel). Tandis que la consommation du jeton dans Ac par le tir de t_4 rend inévitable l'occurrence de l'événement redouté, c'est ce qui est représenté par le premier ordre partiel.

La construction des ordres partiels régissant le scénario redouté ne nécessite pas l'exploration de tous les états accessibles du système étudié. En effet, l'application de la méthode de recherche de scénario à ce cas d'étude montre qu'il n'est pas nécessaire de s'intéresser à l'état d'arrêt du système (place OFF). Ceci est rendu possible par le critère d'arrêt qui délimite la recherche en profondeur des relations de cause à effet. De même, s'il y avait un autre équipement en parallèle avec l'équipement étudié, ses états n'auraient pas été explorés (pas d'enrichissement de marquage le concernant) et ses changements d'états n'auraient pas été entrelacés avec ceux des scénarios obtenus.

Pour une meilleure compréhension des conditions d'occurrence du scénario redouté, il est indispensable de faire intervenir l'aspect continu qui y joue un rôle important. Le cas d'étude qu'on a présenté correspond à l'échauffement d'un équipement quand il est en état de fonctionnement. L'actionneur est, en fait, un ventilateur qui a pour rôle de faire baisser la température de cet équipement. L'événement redouté correspond à un dépassement d'un seuil de température (seuil de sécurité) au delà duquel l'équipement ne peut fonctionner normalement. La transition t_3 correspond à une régulation de la température de l'équipement qui se déclenche quand cette dernière dépasse un seuil de contrôle. Ce seuil est inférieur au seuil de sécurité. Ainsi, si les transitions t_2 et t_3 sont sensibilisées en même temps, t_3 sera toujours franchie avant t_2. C'est donc l'absence d'un jeton dans la place Ac qui est la cause indirecte du scénario redouté (tir de t_2). L'absence de jeton dans Ac se justifie par le tir de la transition t_4 (défaillance du ventilateur). Une fois t_4 franchie, la température ne peut plus être régulée et le système évolue irréversiblement vers l'état redouté. Il y a donc échauffement. C'est le seul scénario qui peut y mener.

VII Conclusion

Au cours de ce chapitre, nous avons formalisé la notion de scénario redouté sous la forme de relations de cause à effet entre un état de fonctionnement normal et un état redouté. Pour construire ces relations de cause à effet, nous partons de l'état partiel redouté et nous effectuons un raisonnement arrière. Nous avons proposé une définition de ce raisonnement, différente de la notion d'accessibilité puisqu'elle est fondée sur des marquages partiels. Un marquage partiel peut être vu comme un ensemble de marquages accessibles puisque la localisation de certains jetons n'est pas spécifiée. Ce marquage partiel présente des similarités avec la notion, de même nom, présentée dans les travaux de thèse de A. Benasser [Benasser 00]. Notre démarche est basée sur le déroulement d'un raisonnement avant sur le réseau de Petri inversé. Nous avons présenté ensuite une méthode de recherche de scénarios combinant raisonnement arrière et raisonnement avant et qui est récursive. Le raisonnement arrière permet de construire les relations de cause à effet menant à l'état redouté et le raisonnement avant permet de mettre en évidence les bifurcations entre le scénario redouté et les comportements permettant de l'éviter. Une étude approfondie de ces bifurcations nous renseigne sur les conditions d'occurrence du scénario redouté. Nous pourrons éventuellement appliquer une deuxième fois, si nécessaire, la méthode en partant des états conditionneurs afin d'identifier les scénarios qui y mènent. C'est le principe de la récursivité. Nous avons proposé, enfin, un algorithme qui permet de rendre automatique l'application des raisonnements arrière et avant et nous l'avons appliqué sur un cas d'étude.

Dans le chapitre suivant, nous allons présenter les résultats de l'application de notre méthode de recherche de scénarios sur deux exemples de systèmes mécatroniques.

Chapitre 4 : Application

I Introduction

Dans le chapitre précédent, nous avons présenté une méthode de recherche de scénarios redoutés basée sur un raisonnement arrière puis un raisonnement avant. Ces deux raisonnements sont rendus automatiques grâce à un algorithme que nous avons proposé. Cette méthode est à appliquer à partir d'un modèle réseau de Petri. Ce modèle est le réseau de Petri ordinaire sous-jacent au réseau de Petri Prédicats Transitions Différentiels Stochastiques du système étudié.

Au travers de deux exemples simples de systèmes mécatroniques, nous allons mettre en application notre méthode de recherche de scénarios. Nous en présenterons les résultats sur chacun des systèmes choisis et nous en signalerons également les limites et les perspectives d'amélioration. Nous traiterons tout d'abord l'exemple du conjoncteur disjoncteur électromécanique puis celui du système des réservoirs. Ce dernier est un modèle simplifié de la suspension hydraulique automobile. Ces deux exemples illustrent parfaitement la problématique des systèmes mécatroniques sans toutefois en présenter toute la complexité. Le traitement de systèmes complexes nécessitera le développement du logiciel supportant la méthode, chose non faite à ce jour.

II Le conjoncteur disjoncteur électromécanique

Nous présenterons tout d'abord le fonctionnement du conjoncteur disjoncteur, ses modes de défaillance et de reconfiguration. Nous expliquerons ensuite la modélisation de ce type de système avec le formalisme des réseaux de Petri Prédicats Transitions Différentiels Stochastiques avant de présenter les résultats de l'application de notre méthode sur cet exemple.

II.A Présentation du système

II.A.1 Le fonctionnement

Le conjoncteur-disjoncteur (voir Figure 4.1) est un organe intermédiaire entre la pompe et les asservissements hydrauliques, consommateurs d'huile sous pression (le système de freinage, les suspensions et la boîte de vitesses).

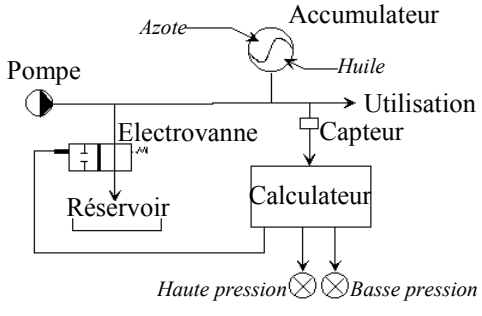


Figure 4.1. **Conjoncteur-disjoncteur électromécanique**

Il doit assurer la régulation de la pression de l'huile dans un accumulateur. Cette pression doit rester dans un intervalle donné. Le contrôle s'opère à l'aide d'un calculateur. Ce dernier décide, suivant la valeur de la pression retournée par un capteur, d'approvisionner ou non l'accumulateur, en commandant ou non l'alimentation d'une électrovanne[5].

En fonction de l'état de l'électrovanne (ouverte ou fermée) et de la consommation des circuits hydrauliques (niveaux de consommation haut ou bas), on distingue les quatre états suivants du système (en absence de défaillances) :

- La phase de conjonction à haute consommation lorsque l'électrovanne est ouverte et les circuits hydrauliques fortement sollicités ; la pression de l'huile dans l'accumulateur est par conséquent croissante durant cette phase,
- La phase de disjonction à forte consommation pendant la fermeture de l'électrovanne et la sollicitation des circuits hydrauliques ; la pression est donc décroissante,
- La phase de disjonction à faible consommation (ici supposée nulle) lorsque l'électrovanne est fermée et la consommation nulle ; la pression est alors constante,
- La phase de conjonction à faible consommation où l'électrovanne est ouverte et la consommation nulle ; par conséquent la pression est croissante.

II.A.2 Les modes de défaillances et les reconfigurations

Dans notre modèle, on suppose que seule l'électrovanne peut subir des défaillances. Ces modes de défaillance sont :

- Le blocage en ouverture
- Le blocage en fermeture

Lorsque l'électrovanne est bloquée, on propose la reconfiguration suivante : l'électrovanne est « secouée » par un train de commande impulsionnel pendant 0.1s. Si elle est débloquée (probabilité de réussite ou de récupération p), la manœuvre est réussie et le système retrouve son fonctionnement nominal. En cas d'échec (probabilité 1-p), le système dérive jusqu'à ce que l'alarme se déclenche.

II.B Modélisation du conjoncteur-disjoncteur

Nous modéliserons tout d'abord le système dans son fonctionnement nominal. Nous nous intéresserons ensuite à la défaillance et au mode de reconfiguration de l'électrovanne. Nous expliquerons par la suite comment nous avons modélisé le déclenchement de l'alarme quand la pression dans l'accumulateur sort de l'intervalle de sécurité. En associant ces différents modules, nous aboutirons au modèle complet du système.

II.B.1 Modélisation du fonctionnement nominal

Le fonctionnement nominal représente l'état du système en l'absence de défaillances. La pression dans l'accumulateur est régie par quatre systèmes d'équations différentielles en fonction de l'état de l'électrovanne (ouverte ou fermée) et de la consommation des circuits hydrauliques (niveau haut ou bas).

Le schéma suivant (voir Figure 4.2) représente un réseau de Petri Prédicat-Transition-Différentiel [Champagnat 98] dans lequel on associe respectivement aux places P_1, P_2, P_3 et P_4 les équations régissant l'évolution des grandeurs continues du système à savoir le volume et la pression de l'huile dans l'accumulateur.

[5] Une électrovanne est un appareil mobile à commande électronique, ici tout ou rien, permettant de régler l'écoulement d'un fluide.

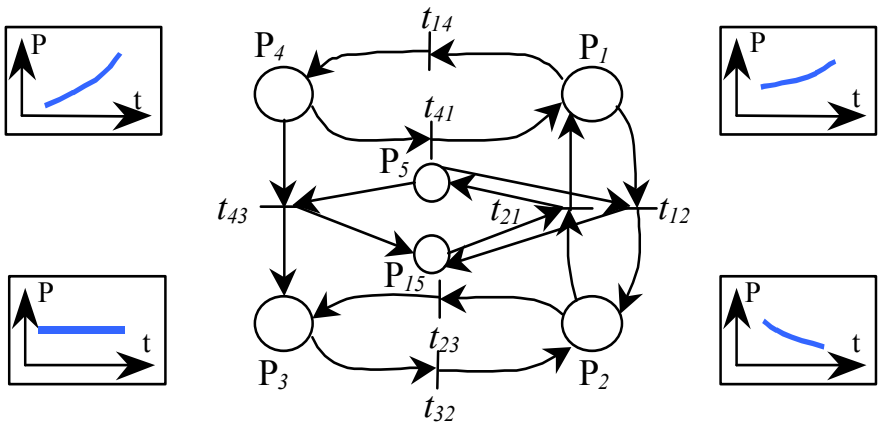


Figure 4.2. **Modèle du fonctionnement nominal et illustrations**

Ainsi, P_1 et P_4 modélisent la phase de conjonction, tandis que P_2 et P_3 modélisent la phase de disjonction. Les places P_5 et P_{15} représentent les états de l'électrovanne respectivement en ouverture et en fermeture. Les transitions t_{14}, t_{41}, t_{23} et t_{32} modélisent le changement de consommation, soit vers la baisse pour t_{41} et t_{32} ou vers la hausse pour t_{14} et t_{23}. Les transitions t_{12}, t_{21} et t_{43} représentent les ordres du calculateur pour commander la fermeture de l'électrovanne (pour t_{12} et t_{43}) quand $P \geq P_{\max}$ ou son ouverture (pour t_{21}) quand $P \leq P_{\min}$.

II.B.2 Modélisation de la défaillance et de la reconfiguration de l'électrovanne

Pour prendre en compte l'apparition d'une défaillance de l'électrovanne, nous introduisons une place P_0. L'occurrence d'une défaillance de l'électrovanne est alors modélisée par la mise d'un jeton dans cette place (voir Figure 4.3).

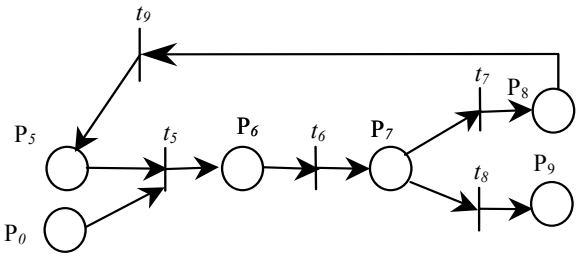


Figure 4.3. **Modèle de reconfiguration de l'électrovanne**

Quand l'électrovanne est ouverte et qu'elle est défaillante (les places P_5 et P_0 sont marquées), et quand la condition $P \geq P_{reconf\_\max}$ est vraie (avec $P_{\max} < P_{reconf\_\max}$), nous commençons la procédure de reconfiguration en tirant la transition t_5 et en marquant la place P_6. Cette dernière représente l'état de l'électrovanne quand elle est bloquée en ouverture. La transition t_6 est immédiatement franchie et la place P_7 est marquée. Cette dernière correspond à l'action de reconfiguration (envoi d'un train d'impulsions pour secouer l'electrovanne). La probabilité de réussite de la procédure de reconfiguration est alors déterminée. En cas de réussite, la transition t_7 est tirée. Dans ce cas, l'électrovanne est débloquée et on remet, par conséquent, un jeton dans la place initiale P_5 par tirage de la transition t_9. En cas d'échec de la procédure, on tire la transition t_8 et on marque la place P_9. Aux transitions t_7 et t_8 sont associées des fonctions stochastiques (respectivement probablités p et 1-p).

II.B.3 Modélisation du déclenchement de l'alarme

L'alarme se déclenche quand la pression sort de l'intervalle de sécurité [P_{alarm_min}, P_{alarm_max}]. Traitons le cas où la pression dépasse sa limite supérieure autorisée (P_{alarm_max}). L'autre cas est tout à fait similaire.

Le réseau de Petri de la Figure 4.4 est constitué de deux places (P_1 et P_h) et d'une transition (t_1).

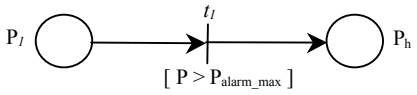


Figure 4.4. Modèle de déclenchement de l'alarme

La place P_1 représente la phase de conjonction à faible consommation (à laquelle on associe le système d'équations correspondant). P_h représente l'état du système quand la pression dépasse sa limite supérieure autorisée. La détection ou la surveillance de cette limite est modélisée par une fonction de sensibilisation sous la forme d'un prédicat ($P \geq P_{alarm\_max}$) associé à la transition t_1.

II.B.4 Modélisation du système complet

Le modèle complet du système (voir Figure 4.5) est composé du modèle du fonctionnement nominal (Figure 4.2), des modèles du déclenchement des alarmes de haute et de basse pression (Figure 4.4) qui sont reliés aux places P_1, P_4 d'une part et à la place P_2 d'autre part, et finalement des modèles de la défaillance de l'électrovanne et de sa reconfiguration (reliés à P_1 et P_5 (Figure 4.3), à P_2 et P_{15}, et à P_4 et P_5).

Considérons à titre d'exemple la situation suivante : l'électrovanne s'est bloquée (un jeton dans P_0) pendant la phase de conjonction à faible consommation (un jeton dans P_1). La procédure de reconfiguration ne commence que lorsque la pression dépasse sa limite autorisée supérieure. On a donc besoin d'une part, d'associer un prédicat ($P \geq P_{max}$) à la transition t_5 et, d'autre part, de relier cette dernière à la place P_1 (par un arc bidirectionnel) afin d'accéder à la valeur de la pression pour vérifier si le prédicat précédent est vrai.

NOTE. — Les places du réseau de Petri complet du système qui sont colorées ont été dupliquées de part et d'autre du réseau afin de ne pas compromettre la lisibilité du modèle.

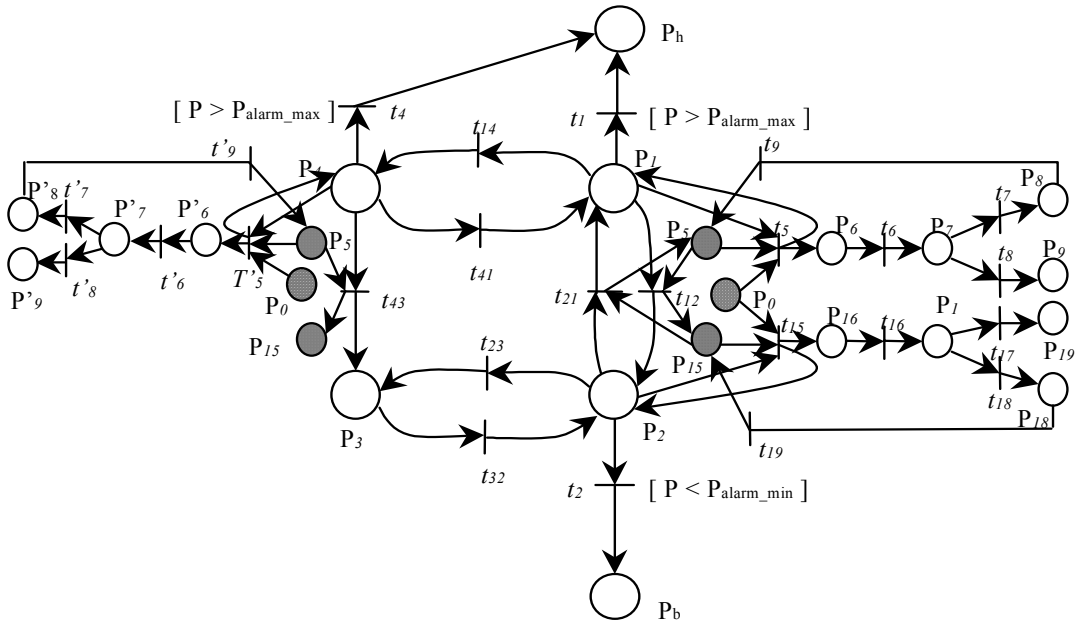


Figure 4.5. Modèle du système complet

Après avoir présenté le cas d'application ainsi que sa modélisation avec les réseaux de Petri Prédicats Transitions Différentiels Stochastiques, nous exposerons, dans la suite, les résultats issus de l'application de notre méthode à cet exemple de système mécatronique.

II.C Application de la méthode de recherche de scénarios

Rappelons que notre démarche a pour but de caractériser les scénarios redoutés et de mettre en évidence les bifurcations entre le fonctionnement nominal et ces scénarios. Nous avons choisi d'étudier les scénarios redoutés menant, par exemple, au déclenchement de l'alarme basse pression. Ceci est modélisé par le marquage partiel de la place P_b du réseau de Petri du système (Figure 4.6). Ce marquage partiel représente ce que nous appelons l'état cible. Il reste à connaître les états normaux. Ces états regroupent tous ceux correspondant au marquage de chacune des places suivantes : P_1, P_2, P_3, P_4, P_5 et P_{15}.

II.C.1 Résultats

Sans détailler les pas de l'algorithme, l'application de la méthode de recherche de scénarios à partir de l'état partiel redouté tel qu'il y a un jeton dans la place P_b se déroule comme suit :

A la suite du premier pas de raisonnement arrière, on obtient le séquent suivant : $P_2, t_2 \vdash P_b$ qui correspond à l'accessibilité de la place P_b à partir de la place P_2 en tirant une fois t_2. On arrête la procédure car P_2 correspond à un état de fonctionnement nominal.

Le raisonnement avant démarre avec un marquage initial contenant la place P_2. Les transitions de sortie de P2 sont t_2, t_{23}, t_{21} et t_{15}. Nous en déduisons cinq scénarios en conflit dont les trois premiers sont décrits par les séquents suivants :

- $P_2, t_2 \vdash P_b$ (1),
- $P_2, t_{23} \vdash P_3$ (2),
- $P_2 \otimes P_{15}, t_{21} \vdash P_1 \otimes P_5$ (3),

Lors de la génération du quatrième séquent, un conflit structurel apparaît entre les transitions t_{17} et t_{18}. Nous obtenons donc deux autres scénarios traduits par les deux séquents suivants :

- $P_2 \otimes P_{15} \otimes P_0, t_{15}, t_{16}, t_{18}, t_{19} \vdash P_2 \otimes P_{15}$ (4),
- $P_2 \otimes P_{15} \otimes P_0, t_{15}, t_{16}, t_{17} \vdash P_2 \otimes P_{19}$ (5).

Le séquent (1) est le séquent obtenu par la procédure de raisonnement arrière. Le séquent (2) représente un changement du niveau de consommation (haut vers bas). Quant au séquent (3), il traduit l'ouverture de l'électrovanne lorsque la pression atteint sa limite inférieure ($P \le P_{min}$). Le séquent (4) représente le scénario de défaillance de l'électrovanne (place P_0 marquée) pendant qu'elle est fermée (P_{15} marquée) avec réussite de la procédure de reconfiguration (tir de t_{18} au lieu de t_{17}). Quant au séquent (5), il traduit l'échec de la procédure de reconfiguration (P_{19} marquée) suite aux mêmes événements.

La preuve de ces séquents ainsi que l'annotation des arbres de preuve canoniques permettent de construire les ordres partiels représentés sous forme de graphes de précédence (figure 4.6). Au séquent (i) est associé le graphe (i).

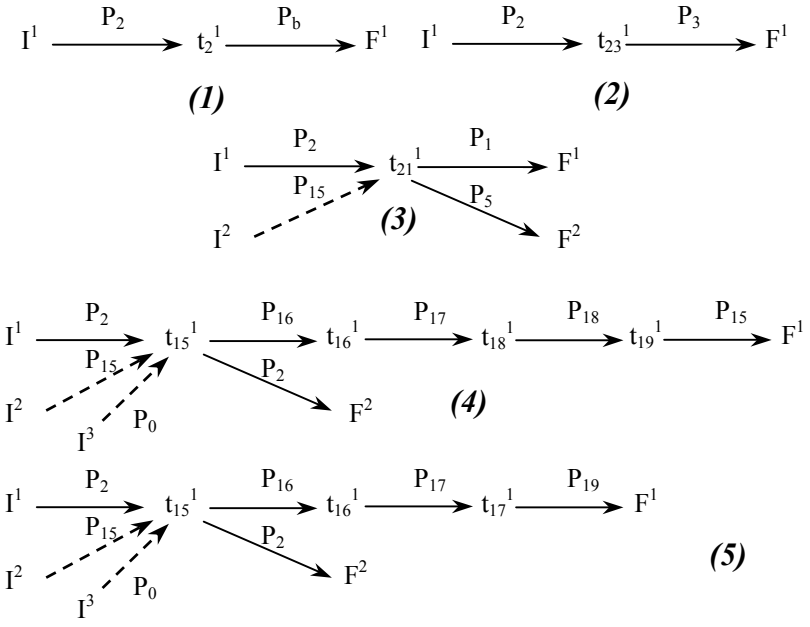


Figure 4.6. Graphes de précédence

II.C.2 Exploitation des résultats

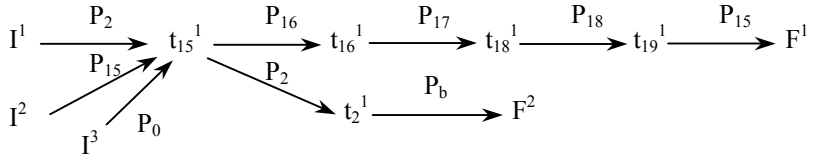
A l'issue de l'application de notre méthode, nous obtenons, en plus de celui menant à l'état redouté (séquent 1), deux scénarios qui correspondent à un fonctionnement normal :

séquents 2 et 3. Quant aux deux séquents restants (4) et (5), ce sont les seuls qui font intervenir un état de défaillance (celui de l'électrovanne, place P_0). Faisons l'hypothèse que tous les scénarios redoutés contiennent au moins un état de défaillance. Cette défaillance sera vérifiée ultérieurement. On ne s'intéresse plus désormais aux séquents (2) et (3).

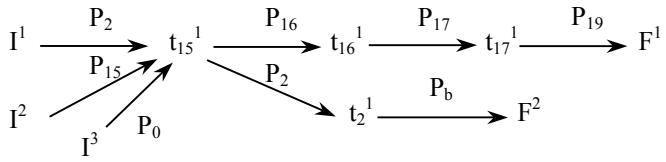
Nous nous intéressons maintenant aux éventuelles interactions entre les scénarios restants. Une étude plus approfondie des deux scénarios (4) et (5) montre que ce sont les seuls qui consomment le jeton dans la place P_2 mais qui en reproduisent également un dans cette même place. En effet, l'atome P_2 est présent dans les membres gauche et droit des deux séquents. Etant donné que le séquent (1), représentant un scénario menant vers l'état redouté, démarre de cette même place, pourrait-il y avoir une quelconque interaction entre l'accumulateur et l'électrovanne qui expliquerait plus en détail l'occurrence de l'événement redouté basse pression ? Il est à noter que le séquent (1) est pauvre en information sur les conditions d'occurrence de cet événement.

On pourrait donc envisager de combiner les scénarios (4) et (5) avec celui représenté par le séquent (1). On s'intéresse désormais aux scénarios qui contiennent à la fois le tir de t2 ainsi que celui de t_{15}.

Sans recommencer un raisonnement avant, nous nous appuyons uniquement sur les graphes de précédence de la figure 4.6. Remarquons qu'il suffit de faire correspondre le début du graphe (1) avec la fin des graphes (4) et (5) pour aboutir à un scénario contenant les deux transitions en question. En effet, il suffit de changer l'événement initial I^1 du graphe (1) par l'instance de tir de la transition t_{15} dans les graphes (4) et (5). Cette transition produit bien évidemment un jeton dans P_2. On obtient ainsi les graphes suivants :



(4-1)



(5-1)

Figure 4.7. Nouveaux graphes de précédence

Ce sont les seuls scénarios possibles (4-1) et (5-1) contenant le tir des transitions t_2 et t_{15}. En effet, tirer t_2 en premier rend impossible celui de t_{15} (marquage insuffisant). Par contre, le tir de cette dernière en premier n'inhibe pas celui de t_2 puisqu'il produit un marquage suffisant pour le tir de t_2.

Nous avons obtenu, à ce stade de l'étude, deux nouveaux scénarios qui pourraient mener, de manière logique, à l'événement redouté. Regardons maintenant si ces scénarios sont toujours possibles en prenant en compte cette fois la partie continue.

A chacune des transitions t_2 et t_{15} est associé un seuil. Le seuil [$P < P_{reconf\_min}$] est associé à t_{15} et le seuil [$P < P_{alarm\_min}$] est associé à t_2. Comme $P_{alarm\_min} < P_{reconf\_min}$, la transition t_2 ne peut être franchie avant t_{15}. Qu'est-ce qui pourrait provoquer l'événement redouté basse pression ? Sachant que, quand elle est sensibilisée, t_{15} doit être tirée avant t_2, et que si t_2 a été tirée, cela signifie que t_{15} n'a pas été sensibilisée pendant une certaine durée au cours de laquelle la pression a continué à décroître jusqu'à atteindre P_{alarm_min}. Le scénario correspondant au graphe de précédence (5-1) vérifie cette contrainte. En effet, suite à l'échec de la reconfiguration, la place P_{15}, en amont de t_{15}, ne peut plus être marquée, ce qui rend le tir de t_{15} impossible. Par conséquent, le scénario (5-1) fait partie des scénarios redoutés menant à l'état basse pression.

Quant au scénario (4-1), cela n'est possible que si l'indisponibilité d'un jeton dans la place P_{15} dure suffisamment longtemps pour que la pression puisse passer du seuil P_{reconf_min} au seuil P_{alarm_min}. Ceci correspond à une période de reconfiguration supérieure au temps nécessaire pour la pression de passer d'un seuil à l'autre en décroissant. Il est donc nécessaire de dimensionner le système de telle sorte qu'il satisfasse à cette contrainte. C'est effectivement le cas pour notre système. Ainsi, on exclut le scénario (4-1) de la liste des scénarios redoutés car il est physiquement impossible.

Récapitulons. Sous l'hypothèse que tous les scénarios redoutés contiennent au moins un état de défaillance, nous avons trouvé deux scénarios menant à l'événement redouté basse pression dont un physiquement impossible. Existe-t-il des scénarios qui ne contiennent pas des états de défaillance ? Si on exclut la défaillance de l'électrovanne, cela suppose sa disponibilité permanente pour éviter que la pression chute en dessous du seuil de sécurité. Ceci implique que la transition t_{21} sera toujours franchie avant t_2, ce qui exclurait l'occurrence de l'événement redouté.

En conclusion, il ressort de cette étude que l'unique scénario de défaillance, mis en évidence par notre modèle, qui mène au déclenchement de l'alarme basse pression (considéré comme événement redouté), est celui qui implique l'échec de reconfiguration de l'électrovanne. Le scénario que nous avons identifié est le scénario minimal qui mène vers l'événement redouté. En effet, il n'inclut aucun événement non indispensable à l'occurrence de cet événement.

Après avoir appliqué la méthode de recherche de scénarios à l'exemple du conjoncteur disjoncteur, nous présenterons par la suite un autre exemple de système mécatronique : le système de régulation des réservoirs. Cet exemple utilise une reconfiguration particulière qui est mise en oeuvre par un partage de ressource (une électrovanne de secours) entre les réservoirs à réguler. Nous appliquerons ensuite notre méthode sur cet exemple.

III Le système de régulation des réservoirs

III.A Présentation

Il s'agit d'un système de régulation du volume de deux réservoirs (cf figure 4.8). Il est constitué d'un calculateur, de deux pompes, de trois électrovannes (tout ou rien), de deux capteurs de volume et des deux réservoirs régulés (Réservoir 1, Réservoir 2) et d'un troisième réservoir de vidange. Les deux réservoirs régulés alimentent des utilisateurs selon un besoin prédéfini (fonction du temps).

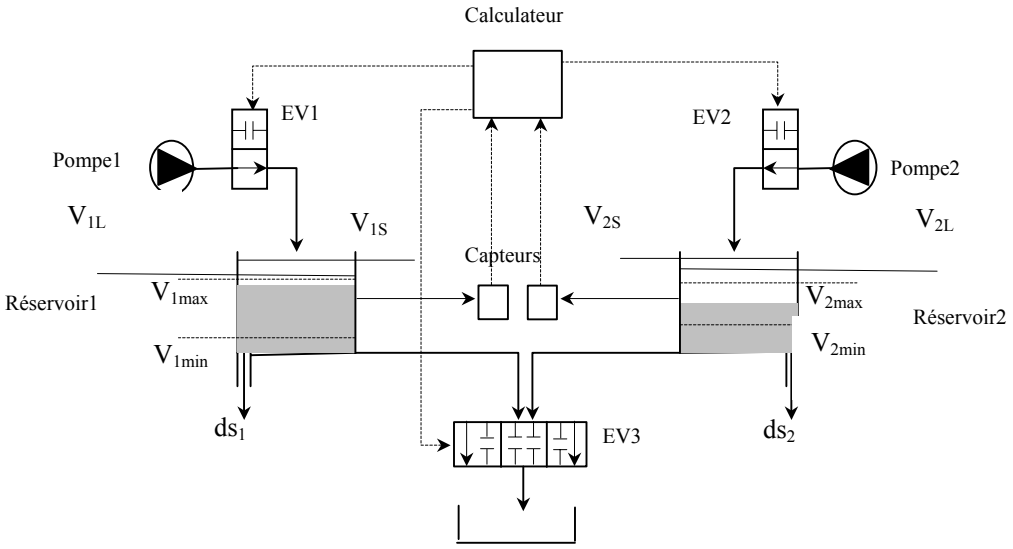


Figure 4.8. **Système de régulation des réservoirs**

Le volume dans chaque réservoir (1 ou 2) doit rester dans un intervalle donné [V_{imin}, V_{imax}]. Le contrôle s'opère à l'aide du calculateur qui décide, selon la valeur du volume (délivrée par le capteur), d'approvisionner (ou non) le réservoir en question en alimentant (ou non) l'électrovanne concernée.

Pour chaque réservoir, on distingue donc deux phases de fonctionnement selon que l'électrovanne alimentant ce réservoir est ouverte ou fermée :

- Une phase de conjonction lorsque l'électrovanne est ouverte. Le volume dans le réservoir est croissant durant cette phase, et cela quel que soit la valeur du débit de sortie vers l'utilisateur (le débit d'alimentation de l'électrovanne est bien supérieur, par hypothèse, au débit de sortie).

- Une phase de disjonction lorsque l'électrovanne est fermée. Le volume dans le réservoir est par conséquent décroissant.

La loi de contrôle du calculateur pour chaque réservoir est telle que lorsque le volume dépasse la limite supérieure de commande V_{imax} pendant la phase de conjonction, alors le calculateur commande la fermeture de l'électrovanne. Lorsque le volume devient inférieur à V_{imin} (limite inférieure de commande) durant la phase de disjonction, alors le calculateur

commande à l'électrovanne de s'ouvrir et on change par conséquent de phase de fonctionnement.

Ce système doit assurer l'approvisionnement des utilisateurs tout en évitant le débordement de l'un des réservoirs. Une troisième électrovanne de secours est prévue pour cet effet. Elle est partagée entre les deux réservoirs et assure leur vidange quand ils débordent. Elle ne peut être utilisée que par un seul réservoir à la fois. Quand le volume dans l'un des réservoirs dépasse la limite supérieure de sécurité (V_{iL}), alors le calculateur commande l'ouverture de cette électrovanne du côté du réservoir qui risque de déborder, et ce, jusqu'à ce que le volume devienne inférieur à V_{imin}. En effet, le débit de vidange de l'électrovanne de secours étant supérieur aux débits des pompes 1 et 2, le volume ne peut que décroître pendant la phase de vidange du réservoir concerné.

Pour simplifier, nous supposons que seules les électrovannes peuvent subir des défaillances. Les électrovannes 1 et 2 (prévues pour l'alimentation des réservoirs) peuvent être bloquées en ouverture. Lorsque l'électrovanne 3 (de secours) est défaillante, elle est mise hors service.

III.B Modélisation

III.B.1 Modèle du fonctionnement nominal

Le fonctionnement nominal du système des deux réservoirs consiste en une succession de phases de conjonction et de disjonction suite à des commandes d'ouverture et de fermeture des électrovannes. Le fonctionnement des deux réservoirs est identique en termes d'états et de succession d'états. En effet, les deux réservoirs possèdent la même loi de commande et les deux électrovannes possèdent les mêmes modes de défaillance. Une fois le modèle du réservoir 1 et de sa commande établi, il suffit de le dupliquer en adaptant tout simplement les seuils de commande et les paramètres de défaillances et de réparation à ceux du réservoir 2.

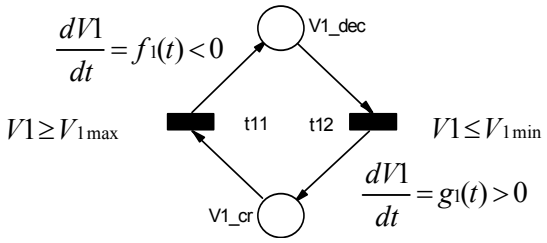


Figure 4.9. **Modèle du fonctionnement nominal du réservoir 1**

La figure 4.9 illustre le modèle de fonctionnement nominal du réservoir 1. La place V_{1_dec} représente la phase de disjonction (le volume décroît) tandis que la place V_{1_cr} représente la phase de conjonction pendant laquelle le volume croit. La place EV_{1_OK} modélise le bon fonctionnement de l'électrovanne 1. Les transitions t_{11} et t_{12} représentent respectivement la commande de fermeture de l'électrovanne 1 quand le volume dépasse V_{1max} et la commande d'ouverture de la même électrovanne quand le volume devient inférieur à V_{1min}.

III.B.2 Modèle de défaillance et de réparation de l'électrovanne 1

Le modèle du blocage en ouverture de l'électrovanne 1 est le suivant :

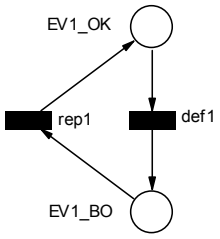


Figure 4.10. **Défaillance et réparation de l'électrovanne 1**

Il représente le fait que l'électrovanne reste bloquée en ouverture après le tir de def_1 et qu'elle peut reprendre un comportement normal après réparation (tir de rep_1).

III.B.3 Modèle d'utilisation de l'électrovanne de secours

Cette électrovanne peut être utilisée de manière identique par les deux réservoirs 1 et 2. Par exemple, quand le volume dans le réservoir 1 dépasse la limite supérieure de sécurité (V_{1L}), et si l'électrovanne de secours est disponible (la place EV_{3_OK} est marquée), alors t_{14} devient franchissable et on commence la procédure de vidange du réservoir 1 via l'électrovanne 3 en marquant la place EV_{3_oc1}. L'électrovanne n'est pas disponible pour une autre utilisation que celle en cours (place EV_{3_OK} vide). Cette phase dure le temps que met le volume pour atteindre le seuil bas V_{1min}. Ensuite, on libère l'électrovanne 3 (on marque de nouveau EV_{3_OK}) et on recommence une phase de conjonction (on remet un jeton dans la place V_{1_cr}) en tirant la transition t_{15}.

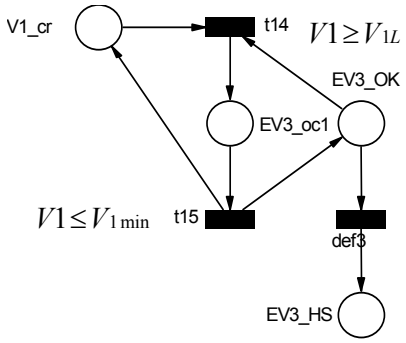


Figure 4.11. **Modèle d'utilisation de l'électrovanne de secours**

L'électrovanne peut subir une défaillance (tir de la transition def_3). Dans ce cas, la place EV_{3_HS} est marquée et l'électrovanne est mise hors service.

III.B.4 Modèle de système complet

Le modèle du système de régulation est le RdP de la figure 4.12. Il regroupe les modèles de fonctionnement nominal des deux réservoirs, les modèles de défaillance et de réparation

des électrovannes 1 et 2, les modèles d'utilisation de l'électrovanne de secours ainsi que les modèles d'occurrence des événements redoutés débordement des réservoirs 1 et 2.

On déclare qu'il y a débordement d'un des deux réservoirs, par exemple le réservoir 1, quand le volume dans ce dernier dépasse V_{1S} (V_{1S} étant supérieur à V_{1max} et à V_{1L}). Dans ce cas, on tire la transition t_{13} et on marque la place E_red_1.

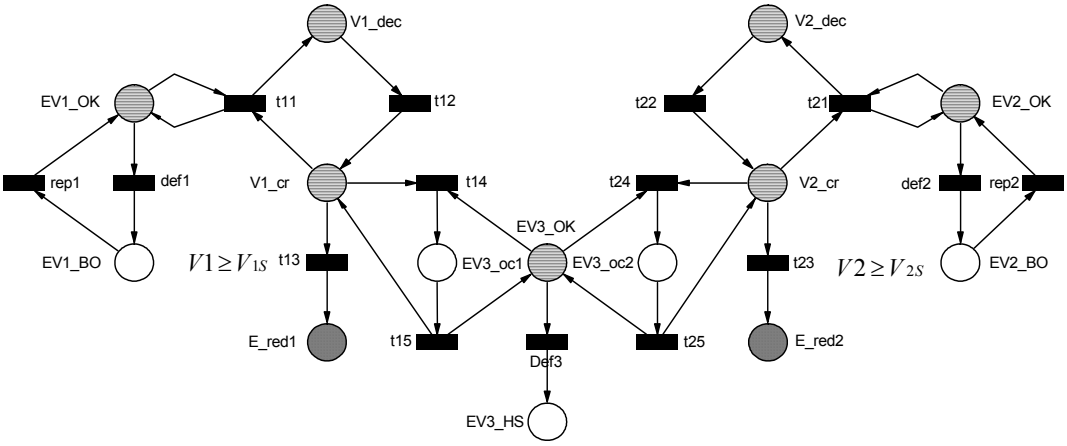


Figure 4.12. Modèle RdP du système complet

III.C Application de la méthode de recherche de scénarios

Nous nous intéressons aux scénarios redoutés menant au débordement du réservoir 1. Ceci correspond au marquage partiel de la place E_red1. Les états représentant un fonctionnement nominal sont ceux associés au marquage des places hachurées sur le modèle RdP du système complet.

III.C.1 Résultats

A partir du marquage partiel de la place E_red1, le premier pas du raisonnement arrière, opérant sur le RdP inversé, correspond au tir de la transition t_{13} et le marquage de la place V_{1_cr}. Celle-ci étant une place de bon fonctionnement, le raisonnement arrière s'achève à ce stade.

Nous commençons donc le raisonnement avant à partir de l'état conditionneur identifié à l'étape précédente : celui associé au marquage partiel de la place V_{1_cr}. A partir de cet état, on recense trois transitions en conflit : t_{11}, t_{13} et t_{14}. La transition t_{13} étant la seule franchissable, elle est franchie en premier. Cela nous redonne le sous-scénario menant vers l'état redouté trouvé suite à l'étape de raisonnement arrière. En ce qui concerne les transitions t_{11} et t_{14}, elles sont potentiellement franchissables. Pour les franchir, il est nécessaire de procéder à un enrichissement du marquage. Intéressons nous au raisonnement consistant à tirer t_{11}.

Nous mettons un jeton dans la place EV_{1_OK}, la seule place en amont de t_{11} non marquée. Cet enrichissement du marquage est cohérent avec les invariants de place du réseau de Petri du système et il est donc mémorisé. Nous obtenons ainsi un marquage partiel contenant un jeton dans V_{1_cr} et un autre dans EV_{1_OK}. A partir de ce marquage partiel, nous mémorisons le contexte avant de tirer t_{11}. Cela nous servira pour construire les ordres partiels en conflit avec cette transition. Une fois t_{11} tirée, nous obtenons un jeton dans V_{1_dec}et un autre dans EV_{1_OK}.

Cette dernière étant une place normale, le raisonnement avant la concernant s'arrête à ce stade. Il en est de même pour la place EV_{1_OK} car l'algorithme, détectant les boucles, met fin à ce raisonnement. Revenons au contexte mémorisé avant le franchissement de t_{11}. Celui-ci permet de franchir les transitons en conflit avec t_{11}. Ce contexte est tel qu'il y a un jeton dans chacune des places V_{1_cr} et EV_{1_OK}. Sachant que nous ne pouvons franchir ni t_{11} (elle est dans la liste des transitions interdites), ni t_{13} (elle a été tirée auparavant), on ne peut tirer que def_1 ou t_14. Nous traitons le cas de def1 car elle est la seule transition franchissable. Son franchissement produit un jeton dans la place EV_{1_BO}. Cette dernière n'étant pas une place normale, le raisonnement se poursuit par le franchissement de rep_1. Cela nous ramène dans la place d'origine : EV_{1_OK}. Ce raisonnement s'arrête ici. En résumé, nous venons de construire les ordres partiels en conflit avec celui issu du tir de t_{11}. Cette dernière étant en conflit avec t_{13}, le fait d'empêcher le tir de t_{11} favorise celui de t_{13}. Ainsi, nous mettons en évidence les scénarios en conflits avec t_{11} et qui pourraient faire partie du scénario redouté global (Figure 4.13).

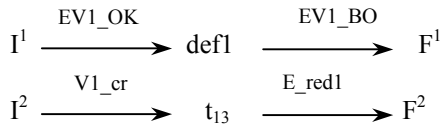


Figure 4.13. **Un sous scénario du scénario global**

Le raisonnement se poursuit de manière similaire en enrichissant le marquage à chaque fois que cela est nécessaire. A partir du marquage partiel courant, seul la transition t_{14} pourrait être franchie. Elle est potentiellement franchissable et son tir nécessite d'enrichir le marquage en ajoutant un jeton dans la place EV_{3_OK}. Après avoir construit l'ordre partiel correspondant au tir de t_{14}, nous nous intéressons ensuite aux transitions qui sont en conflit avec cette dernière. Comme t_{11} et t_{13} ont été déjà tirées, la recherche de scénarios se poursuit autour des transitions en aval de la place EV_{3_OK}, à savoir def_3 et t_{24}. A chaque enrichissement de marquage, nous impliquons les transitions en aval de la place concernée et ainsi de suite, jusqu'à impliquer tous les comportements susceptibles d'intervenir dans le scénario redouté global.

A l'issu de l'application complète de la méthode sur le conjoncteur disjoncteur, nous avons obtenu deux scénarios intéressants (Figure 4.14). Ces deux scénarios sont obtenus par composition entre les sous-scénarios correspondant aux tirs des transitions t_{13} et def_1 (figure 4.13) d'une part et de def_3 ou de t_{24} avec celui de def_2 d'autre part. Ce sont tous des scénarios en conflits avec ceux qui sont eux-mêmes en conflits avec le sous-scénario redouté correspondant au tir de $_{t13}$.

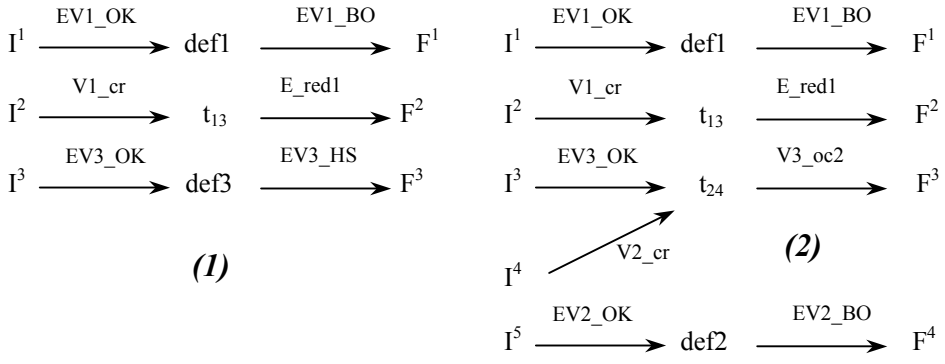


Figure 4.14. **Les deux scénarios redoutés menant au débordement du réservoir 1**

III.C.2 Exploitation des résultats

Le scénario (1) correspond à la défaillance de l'électrovanne 1 combinée avec celle de l'électrovanne de secours. Ce scénario mène effectivement à l'état redouté puisque aucune ressource n'est disponible pour vidanger le réservoir 1 avant son débordement. Notre méthode met également en lumière la compétition entre la réparation de l'électrovanne 1 (tir de rep_1) et l'événement de débordement (tir de t_{13}).

Quant au scénario (2), il concerne également la défaillance de l'électrovanne 1 combinée avec l'indisponibilité de l'électrovanne de secours pendant un certain temps. En effet, le tir de t_{24} correspond au début de la phase de vidange du réservoir 2 par cette électrovanne. Il est clair que cela n'est possible que si l'électrovanne 2 est défaillante (tir de def_2). La période d'indisponibilité de l'électrovanne de secours est importante par rapport à l'occurrence de l'événement redouté dans la mesure où plus elle est longue, plus probable sera le débordement du réservoir 1. Ceci est mis en évidence par notre méthode au travers de la compétition entre les événements suivants : t_{25} (fin de la procédure de vidange du réservoir 2) et t_{13}. Ce scénario correspond à un scénario redouté possible même s'il pourrait paraître très peu probable puisqu'il résulte d'une double défaillance. Cela n'est pas si évident à affirmer. Prenons par exemple le cas où le système du conjoncteur disjoncteur fait intervenir des électrovannes 1 et 2 qui ne sont pas très fiables (et donc bon marché). L'électrovanne de secours est par conséquent beaucoup sollicitée par les deux réservoirs. La probabilité pour que cette dernière soit demandée à la fois par les deux réservoirs n'est pas négligeable, de même que celle du débordement de l'un deux. Pour évaluer cette probabilité, nous ne pouvons nous passer de la prise en compte des caractéristiques de la partie continue du système, à savoir les débits des pompes et les débits de vidange. En effet, plus la période de régulation des niveaux des réservoirs est longue, plus forte sera la probabilité d'occurrence du scénario redouté.

Ce deuxième scénario a mis en évidence un problème de partage de ressource qui peut dans certains cas être dangereux pour le système et provoquer le débordement d'un des deux réservoirs. Afin de pallier ce problème, nous pouvons ajouter un simple programme qui partagerait l'utilisation de l'électrovanne de secours entre les deux réservoirs en cas de sollicitations simultanées. Alterner la vidange des deux réservoirs par l'électrovanne de secours contribue à rendre la probabilité du deuxième scénario effectivement négligeable. L'analyse quantitative qui pourra être développée par la suite peut donner des résultats précieux à ce niveau des études de conception.

Comme nous l'avons souligné plus haut, notre méthode permet de mettre en évidence les interactions entre différents composants du système susceptibles d'être à l'origine d'un scénario redouté (l'exemple du scénario 2). La méthode des arbres de défaillance permet également de déterminer les combinaisons de défaillance menant vers l'état redouté. Nous allons comparer dans le paragraphe suivant les résultats de cette méthode avec ceux issus de notre méthode.

III.C.3 Comparaison avec la méthode des arbres de défaillance

L'arbre de défaillance classique ne fait intervenir que les états (défaillants ou en bon fonctionnement) des composants nécessaires à l'occurrence d'une situation redoutée. L'arbre relatif au débordement du réservoir 1 est présenté sur la figure 4.15.

Il exprime le fait qu'il est suffisant que les deux électrovannes 1 et 3 soient défaillantes (EV_{1_HS} et EV_{3_HS}) pour que le réservoir 1 déborde (c'est la situation redoutée notée E_red1).

En fait cet arbre est incorrect car l'électrovanne 3 peut ne pas être disponible sans être pour autant hors service. Nous avons affaire à un système dynamique.

Avec une connaissance des états de l'électrovanne 3, il est possible d'utiliser la notion d'arbres de défaillances et les outils associés pour générer un arbre de défaillances ne se restreignant pas uniquement aux états de bon ou mauvais fonctionnement.

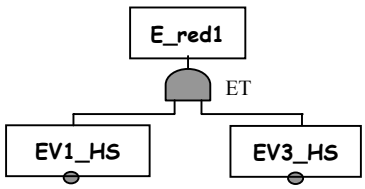


Figure 4.15. **Arbre de défaillance du débordement du réservoir 1**

Dans l'exemple des deux réservoirs, il existe alors un deuxième scénario qui mène à la situation redoutée correspondant à la défaillance de l'électrovanne 1 et à l'utilisation de l'électrovanne 3 pour vider le réservoir 2 (cf Figure 4.16).

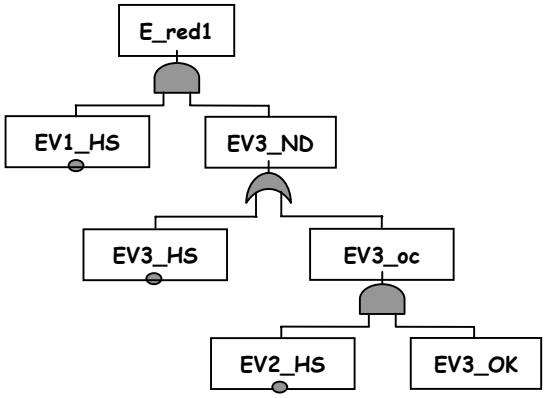


Figure 4.16. **Arbre de défaillance avec prise en compte des états de l'électrovanne 3**

L'arbre de défaillances de la figure 4.16 ne fait référence qu'à des états de composants et n'explicite pas les changements d'états. On trouve ainsi les deux situations critiques (électrovannes 1 et 3 hors service ou électrovanne 1 et 2 hors service et électrovanne 3 occupée à vidanger le réservoir 2) sans pour autant savoir quelle est la suite de changements d'états qui mène d'un état de bon fonctionnement à l'une des deux situations redoutées. En conséquence, les scénarios qui mènent à ces situations redoutées ne peuvent pas être déduits de cet arbre, et comme le système est dynamique, la connaissance des probabilités des états partiels de chaque composant ne suffit pas pour déduire la probabilité des situations redoutées. Notre méthode a donc l'avantage de mieux représenter, sous forme d'un ensemble d'ordres partiels entre les événements, les scénarios menant à l'occurrence d'un événement redouté tout en décrivant les changements d'états du système à partir d'un état de bon fonctionnement jusqu'à l'occurrence de l'événement redouté en question.

Certes la méthode des arbres de défaillance ne représente pas de manière complète et précise l'évolution du système vers l'état redouté, toutefois, elle permet de regrouper l'ensemble des combinaisons de défaillances menant à cet état sur un seul graphique.

Plusieurs combinaisons possibles peuvent être associées à un seul événement redouté, tandis que notre méthode représente chaque ensemble d'événements menant vers l'état redouté sous la forme d'un unique ordre partiel. Plusieurs ordres partiels peuvent donc être associés à un seul état redouté.

Comme le problème du nombre de coupes minimales se pose dans le cas des arbres de défaillances, notre méthode peut également générer un nombre important de scénarios comme c'était le cas pour le conjoncteur disjoncteur. Nous proposons une technique qui permet de réduire le nombre de scénarios générés par notre méthode en favorisant la lisibilité et la compréhension de ces derniers en augmentant le nombre d'itérations de la méthode.

III.C.4 Comment diminuer le nombre de scénarios ?

Comme nous l'avons évoqué dans la section précédente, l'application de notre méthode au conjoncteur disjoncteur a généré un grand nombre de scénarios. Cela provient du fait qu'à chaque fois que nous enrichissons le marquage pour pouvoir franchir une transition donnée, chaque jeton ajouté, comme tout autre jeton non enrichi, fait intervenir toutes les transitions qui sont en aval de la place le contenant. Cela signifie qu'à chaque fois qu'on ajoute une hypothèse sur l'état d'un nouveau composant non impliqué jusque là dans la situation dangereuse pour pouvoir avancer dans le raisonnement, nous faisons intervenir tous les états de ce même composant ayant un lien avec l'état concerné (que nous appellerons état cible). Ceci contribue à accroître le nombre de scénarios issus de l'application de la méthode.

Si, maintenant, au lieu de traiter de manière identique les jetons enrichis et les jetons non enrichis nous n'autorisons plus le franchissement des transitions sensibilisées uniquement par des jetons enrichis, nous diminuerions le nombre de scénarios. Ceci revient en effet à ne s'intéresser à une étape donnée du raisonnement qu'aux évolutions possibles du composant concerné et non aux autres composants. L'étude de chacun de ces derniers se fera au cours d'une itération ultérieure de la méthode qui aura pour but de déterminer les scénarios menant à l'état cible en question. Ainsi, même si nous augmentons le nombre d'itération nous privilégions tout de même la modularité de la recherche et la compréhension des sous-scénarios donnés par chaque itération. Le scénario redouté global sera donné par la combinaison des différents sous-scénarios propres à chaque sous-système impliqué dans le scénario redouté.

IV Conclusion

Nous avons appliqué, au cours de ce chapitre, la méthode de recherche de scénarios à deux exemples de systèmes mécatroniques : le conjoncteur disjoncteur électromécanique et le système de régulation des réservoirs.

Le premier exemple met en oeuvre une reconfiguration par sollicitation d'une électrovanne bloquée en ouverture. Soit elle est débloquée (avec une probabilité de réussite fixe p), soit elle reste bloquée et, dans ce cas, elle est déclarée définitivement hors service. L'application de la méthode sur cet exemple, pour l'étude des scénarios menant à l'événement redouté basse pression, montre l'existence de deux scénarios redoutés possibles. Un des deux scénarios est invalidé par des considérations relatives à la prise en compte de la partie continue (la durée de reconfiguration nettement inférieure à la période de régulation). Le second scénario met en évidence un échec de la reconfiguration.

Le second exemple est celui du système de régulation de deux réservoirs. Ce système utilise une reconfiguration de type partage de ressource. Une électrovanne de secours peut être utilisée en effet par les réservoirs mais un seul à la fois. Ce partage de ressource permet certes d'optimiser le nombre d'électrovannes de secours, mais pourrait générer des scénarios redoutés inattendus. Ces scénarios peuvent échapper à la méthode des arbres de défaillances, basée uniquement sur la prise en compte des états de défaillance des composants. Notre méthode permet de déterminer les deux scénarios redoutés de manière systématique car basée sur une modélisation adéquate. Nous avons également soulevé le problème du nombre de scénarios obtenus suite à l'application de la méthode. La solution que nous avons proposée consiste à ne pas autoriser le franchissement des transitions sensibilisées uniquement par des jetons enrichis, dans une même itération, à condition de traiter a posteriori chacune de ces transitions par une itération supplémentaire de la méthode. Ceci a l'avantage de bien structurer la recherche de scénarios et de rendre les résultats de chaque itération plus compréhensible par l'utilisateur. Le scénario redouté global est obtenu en composant les sous-scénarios donnés par chaque itération. L'utilisation de notre méthode sur des exemples plus complexes nécessite le développement d'un logiciel qui sera couplé avec un joueur de réseaux de Petri. Nous pourrons ainsi tester et comparer les deux techniques concernant le traitement des jetons enrichis ainsi que plusieurs stratégies pour le critère d'arrêt qui permet de fixer la profondeur de la recherche.

Conclusion générale

Les travaux développés durant cette thèse nous ont permis de contribuer à l'analyse qualitative des systèmes mécatroniques et plus particulièrement à la recherche des scénarios redoutés. Les systèmes mécatroniques étant des systèmes dynamiques hybrides, nous avons proposé de les représenter à l'aide des réseaux de Petri et d'équations algébro-différentielles. Pour cela, nous avons repris l'approche développée par Ronan Champagnat en y ajoutant une nouvelle classe de fonctions de sensibilisations (les fonctions stochastiques) de façon à modéliser correctement les défaillances.

Pour mettre en évidence les scénarios redoutés, nous nous sommes appuyés sur la logique afin d'exploiter les relations de cause à effet présentes dans le modèle. Nous avons choisi d'utiliser la logique Linéaire à cause de l'équivalence entre accessibilité dans un réseau de Petri et prouvabilité dans cette logique. L'analyse des relations de cause à effet présente un certain nombre de spécificités par rapport à la preuve d'un séquent en logique Linéaire. Tout d'abord, au lieu de raisonner d'un état initial vers un état final, nous avons montré qu'il valait mieux raisonner à partir de l'état redouté et remonter les chaînes de causalité jusqu'à un état normal : il s'agit d'un raisonnement arrière. Nous avons montré au chapitre 3 que ce raisonnement arrière pouvait se faire en logique Linéaire comme le raisonnement avant, à condition d'avoir inversé les arcs du réseau de Petri.

Une autre spécificité de cette analyse des relations de cause à effet est que le marquage de départ du raisonnement est incomplètement connu. En effet, pour éviter l'exploration exhaustive de tous les états accessibles d'un système et pour obtenir des scénarios « minimaux », c'est-à-dire ne faisant intervenir que les événements strictement nécessaires à l'obtention de l'état redouté, nous ne mettons des jetons que dans les places strictement nécessaires, le reste du marquage restant inconnu. Au fur et à mesure du déroulement du raisonnement, nous sommes amenés à faire des hypothèses sur les marquages de certaines places : c'est l'enrichissement du marquage. C'est cette notion originale qui nous permet, dans un système complexe, de n'impliquer que les composants causalement liés au scénario menant à l'état redouté.

Nous avons proposé pour le raisonnement arrière, expliquant comment le système peut arriver dans l'état redouté en question, qu'il soit complété par un raisonnement avant. Ce raisonnement a pour but de mettre en évidence les comportements qui permettent d'éviter d'atteindre l'état redouté. Une bonne caractérisation des bifurcations entre le comportement menant à l'état redouté et ceux qui l'évitent est en effet essentielle pour comprendre les conditions d'occurrence du scénario redouté et pour envisager postérieurement une étude qualitative ciblée.

Nous avons mené ce travail jusqu'à la proposition d'un algorithme permettant d'automatiser les raisonnements avant et arrière en mettant en œuvre de façon contrôlée les

enrichissements de marquage. Finalement, nous avons traité deux exemples de systèmes mécatroniques simples.

Une fois ce travail effectué, nous pouvons nous poser le problème de l'apport de la logique Linéaire car les mécanismes qui sont au cœur de l'algorithme peuvent tout à fait être expliqués de façon intuitive directement sur le réseau de Petri. L'apport de la logique Linéaire est de pouvoir formaliser et justifier ces mécanismes. En effet, dans le cadre de cette logique, il nous est imposé, par mesure de cohérence logique entre les causes et les effets, de faire un seul type de raisonnement à la fois. Par exemple, il faut choisir entre un raisonnement avant et un raisonnement arrière. Au cours d'un raisonnement reliant causes et effets, nous ne pouvons pas combiner les deux en faisant tantôt l'un tantôt l'autre. Nous ne pouvons pas passer du déductif à l'abductif au cours d'un même raisonnement. Dans notre approche, raisonnement arrière et raisonnement avant sont clairement séparés et s'enchaînent l'un après l'autre. Ceci dit, l'algorithme de recherche de scénarios peut parfaitement être expliqué et utilisé sans référence explicite à la logique Linéaire. Ceci est un avantage lorsqu'il s'agira de l'utiliser dans un contexte industriel par des non spécialistes de la logique formelle.

Avant de penser à utiliser l'algorithme de recherche de scénario dans un contexte industriel, il reste du travail à faire. Le premier travail va consister à le mettre en œuvre sous la forme d'un logiciel afin de pouvoir traiter des exemples plus conséquents. Ce n'est qu'ainsi que nous allons éclaircir des points restés encore imprécis dans l'algorithme. Parmi ces points, il y a le critère d'arrêt qui est basé sur la notion d'état normal. Qu'est ce qu'un état normal ? Nous pouvons avoir un ensemble de places qui, individuellement, sont fréquemment marquées sans qu'elles correspondent effectivement à un marquage associé à un état normal. En effet, la situation correspondant au marquage simultané de toutes ces places peut être très rare. Dans une telle situation, le concepteur doit refaire une simulation de Monte Carlo pour chercher la probabilité du marquage puis, si nécessaire, choisir ce marquage comme état cible et relancer un raisonnement arrière. Ce problème est un cas particulier d'un problème plus général, celui de la granularité des scénarios recherchés automatiquement par l'algorithme. Plus le critère d'arrêt est fort, plus le scénario généré par l'algorithme sera court et plus le concepteur sera obligé de réitérer la méthode à partir d'états cibles intermédiaires pour obtenir un scénario réellement significatif et complet. Nous avons vu un exemple de ce problème au chapitre 4 lorsque nous avons traité le cas des réservoirs et de la pompe de secours. Très liée à ce problème, il y a la stratégie d'utilisation des jetons enrichis. Devrons nous ou non franchir des transitions qui sont telles que tous les jetons les sensibilisant sont des jetons enrichis ? Si la réponse est oui, nous aurons plus de scénarios pour une seule exécution de l'algorithme mais, en contrepartie, cet ensemble de scénarios sera plus difficilement compréhensible tout en courant le risque d'énumérer un très grand nombre de marquages. Le concepteur peut être noyé sous un flot d'informations. Si nous répondons par la négative, le risque est d'avoir des scénarios parcellaires et pauvres en information. Ce sera au concepteur d'y remédier en relançant de nombreuses fois l'algorithme sur divers états cibles. Nous pensons qu'il sera nécessaire de programmer et de tester ces différentes stratégies sur des exemples de systèmes mécatroniques complexes.

Ces systèmes vont également nous poser des problèmes de modélisation. En effet, un même système peut être modélisé de diverses façons avec des réseaux de Petri Prédicats Transitions Différentiels et Stochastiques. Entre ces divers modèles, la délimitation entre les parties continue et discrète peut varier et comme l'algorithme n'exploite, à ce stade, que la partie discrète, cela aura un impact sur les scénarios trouvés. Par exemple, si deux composants sont interconnectés par une variable continue partagée et qu'aucune place ne met explicitement en évidence cette interdépendance, elle échappera à notre recherche. Cela pose

le problème de l'exhaustivité de la recherche et de la complétude. De toute façon, il est clair que seuls les scénarios induits par le modèle sont trouvés et que l'exhaustivité par rapport au système réel dépend de la modélisation qui reste et restera longtemps un problème ouvert.

Pour en revenir au problème précédent, l'élimination des variables continues partagées peut amener à définir des contraintes de modélisation assez strictes, comme le fait par exemple d'interdire qu'une variable donnée apparaisse comme attribut de deux jetons différents. Cela doit amener aussi à faire intervenir la partie continue dans le processus de recherche de scénarios. Une manière simple est, par exemple, de faire intervenir les fonctions de sensibilisation associées aux transitions en conflits. Ces fonctions correspondent normalement à des franchissements de seuils et lorsque la même variable est impliquée dans ces seuils, nous pouvons savoir quelle transition est d'abord franchie. Cela doit permettre d'éliminer des scénarios non significatifs. D'une façon générale, l'algorithme que nous avons développé ne s'appuyant que sur l'aspect discret, il faut a posteriori vérifier la cohérence des scénarios obtenus vis-à-vis de la dynamique continue.

Pour des applications réellement complexes, il sera de toute façon nécessaire de mettre en œuvre une approche de modélisation structurée et modulaire. Des travaux sont en cours [Villani 02] à propos de l'intégration des réseaux de Petri Prédicats Transitions Différentiels à une démarche à objets basée sur UML pour la modélisation des systèmes dynamiques hybrides complexes. Leur extension aux réseaux de Petri Prédicats Transitions Différentiels et Stochastiques et aux systèmes mécatroniques ne devrait pas poser de problèmes.

Il reste un point important : l'étude quantitative. Il s'agit de l'estimation des probabilités d'occurrence des scénarios redoutés. Ce problème a déjà été partiellement abordé par Gilles Moncelet dans sa thèse. Il a montré que la difficulté principale était la reconstitution d'un état initial cohérent au début du déroulement du scénario et l'évaluation de sa probabilité d'occurrence. Comme nous avons à faire à des systèmes hybrides, une partie de l'état est continue et la reconstitution de l'état initial consiste à associer à un marquage (la partie discrète de l'état produite par notre algorithme de recherche) un état continu et sa probabilité d'occurrence. Nous pouvons utiliser la méthode de Monte Carlo mais il faudra sans doute discrétiser cette partie continue. Dans des cas simples où les variables continues sont équiprobables dans leur domaines et où l'on est capable d'évaluer la probabilité d'occurrence de chaque branche des bifurcations, l'évaluation se fera de façon analytique de manière similaire à ce qui se fait avec les arbres de défaillance : ce sera un simple produit de probabilités.

Bibliographie

[Abouïassa 95] Hassane Abouïassa : " Contribution à l'unification des méthodes de modélisation et de conception de la commande des systèmes complexes, discrets et continus ", thèse présentée à l'université de Franche-Comté en octobre 1995.

[Acosta 93] Acosta C. et Siu N., « Dynamic event trees in accident sequence analysis : application to steam generator tube rupture », Reliability Engineering and System Safety 41, pp. 135-154, 1993.

[Aldemir 91] T. Aldemir : " Utilization of the cell-to-cell mapping technique to construct Markov failure models for process control systems ", Probabiltic Safety Assessment and Management, G. Apostolakis, Editor, Elsevier Science Publishing CO., 1991

[Alla 98] H. Alla, J.M. Flaus : « Modélisation d'une unité de sotockage de gaz par réseaux de Petri hybrides », ADPM'98, Reims, mars 1998.

[Allam 98] Mohamed Allam, "Sur l'analyse qualitative des réseaux de Petri hybrides Une approche basée sur les automates hybrides", thède de l'Institut Natianal Polytechnique de Grenoble, 7 décembre 1998.

[Alur 95] R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.H. Ho, X. Nocollin, A. Olivero, J. Sifakis, S. Yovine, "The lgorithmic analysis of hybrid systems", Theoritical Computer science, n°138, 1995.

[Andreu 96] David Andreu : " commande et supervision des procédés discontinus ", thèse présentée au LAAS le 15 novembre 1996.

[Atamna 94] Y. Atamna : " Réseaux de Petri temporisés stochastiques classiques et bien formés - définition, analyse et application aux systèmes distribués temps réel ", thèse présentée au LAAS, 1994.

[Aubry 87] J.F. Aubry: « Conception des systèmes de commande numériques des convertisseurs : vers une méthodologie intégrant la sûreté de fonctionnement », Thèse de doctorat d'état.

[Aubry 91] J.F. Aubry et C. Zanne : "Intégration de la sûreté de fonctionnement dans la conception des systèmes de commande numérique des processus électromécaniques", Sûreté de fonctionnement, Vol n°25, n°4, pp 297-324.

[Benasser 00] A. Benasser, « L'accessibilité dans les réseaux de Petri : une approche basée sur la programmation par contraintes », Thèse préparée au LAIL, Lille, 20 janvier 2000.

[Brettschneider 96] H. Brettschneider, H.J. Genrich, H.M. Hanisch, « Verification and performance analysis of recipe-based controllers by means of dynamic plant models », Proceedings de Second International Conference on Computer integrated Manufacturing in the Process Industries, Eindhoven, Hollande, juin 1996.

[Buisson 93] Buisson J., « Analysis of switching devices with Bond Graphs », journal of the Franklin Institute, vol. 330, n°6, p. 1165-175, 1993]

[Buisson 93] J. Buisson, "Analysis oh Switching Devaices with Bond Graphs", Journal of the Franklin Institute, vol. 32, n°6, p. 1165-1175, 1993.

[Chabot 01] Chabot J.L., Dutuit Y. et Rauzy A., « A Petri net approach to dynamic reliability problems », Proceedings of Esrel 2001, Volume 2, pp. 1387-1394, 2001.

[Chabot 01] J.L. Chabot, Y. Dutuit et A. Rauzy, « A petri net approach to dynamic reliability », Esrel 2001.

[Chabot 98] Jean-Luc Chabot, « Approche probabiliste relative à l'étude des scénarios d'incendie », thèse de l'Université de Poitiers-ENSMA, 16 octobre 1998.

[Champagnat 98a] Ronan Champagnat, « Supervision des systèmes discontinus : définition d'un modèle hybride et pilotage en temps-réel », Thèse de Doctorat de l'Université Paul Sabatier de Toulouse, soutenue le 1er octobre 1998.

[Champagnat 98b] R. Champagnat, H. Pingaud, H. Alla, C. Valentin-Roubinet, J.M. Flaus : « A gazstorage example as a benchmark for hybrid modelling », ADPM'98, Reims, mars 1998.

[Champagnat 98c] R. Champagnat, P. Esteban, H. Pingaud, R. Valette : « Modelling and simulation of a hybrid system through Pr/Tr PN-DAE model », ADPM'98, Reims, mars 1998.

[David 89] René David & Hassane Alla : " Du Grafcet aux réseaux de Petri ", Hermes, 1989.

[David 94] R. David, H. Alla, « Petri nets for modelling dynamics systems : a survey », Automatica, vol.30, n°2, p. 175-202, 1994.

[Demmou 02] H. Demmou, S. Khalfaoui, N. Rivière, E. Guilhem, « A method for deriving critical scenarios from mechatronic systems », Journal Européen des Systèmes Automatisés, volume 36 – n°7/2002, pages 987 à 999. Rapport LAAS No02177.

[Devooght 92a] J. Devooght & C. Smidts : " Probabilistic Reactor Dynamics - I: The Theory of Continuous Event Trees ", Nuclear Science and Engineering, Vol. 111, p229-240, 1992

[Devooght 92b] J. Devooght & C. Smidts : " Probabilistic Reactor Dynamics - II: A Monte-Carlo Study of a Fast Reactor Transcient ", Nuclear Science and Engineering, Vol. 111, p241-256, 1992

[Devooght 94] Devooght J. et Smidts C., « A theoretical analysis of DYLAM-type event tree sequences », Proceedings of PSAM-II, Volume 1, pp. 011.1-011.6, 1994.

[Devooght 97] J. Devooght, « Dynamic reliability », Advances in Nuclear Science and Technology 25, pp. 215-278, 1997.

[Dietsche 01] Karl-Heinz Dietsche, Jürgen Crepin, Folkhart Dinkler, « Mémento de technologie automobile », Bosch, ISBN 3934584195.

[Dubi 00] A. Dubi, "Monte Carlo Applications in Systems Engineering", John Wiley&Sons Ltd, ISBN 0-471-981-72-9.

[Dubois 90] D. Dubois et S. Gentil, "Intelligence Artificielle: un outil pour l'informatique", Journée annuelles du GR automatique, octobre 1990.

[Dubois 93] E. Dubois, H. Alla, "Hybrid Petri Nets with a stochastic discrete part", European Control Conference, Groningen, Hollande, juin 1993.

[Dufour 02] F. Dufour et Y. Dutuit, « Dynamic reliability : a new model », Lambda-Mu 13-Esrel 2002 European Conference.

[Dugan 84] J.B. Dugan, K.S. Trivedi, R.M. Geist et V.F. Nicola, "Extende Stochastic Petri Nets : Applications and Analysis", 10th International Symposium on Computer Performance, p. 507-519.

[Dutuit 96] Yves Dutuit, Antoine Rauzy, Jean-Pierre Signoret, Philippe Thomas : " Modélisation d'un système dynamique simple et évaluation de sa fiabilité par Réseaux de Petri Stochastiques ", λμ10, Saint-Malo, octobre 1996

[Dutuit 97] Dutuit Y., Châtelet E., Signoret J.P. et Thomas P., « Dependability modeling and evaluation by using stochastic Petri nets : application to two test cases », Reliability Engineering and System Safety 55, pp. 117-124, 1997.

[Dutuit 97] Yves Dutuit, Antoine Rauzy, Jean-Pierre Signoret, Philippe Thomas : " Analyse qualitative et quantitative de la fiabilité d'un système dynamique simple ", Congrès Qualité et Sûreté de Fonctionnement, Angers, mars 1997

[Ereau 94] J.F. Ereau, M. Saleman, R. Valette, H. Demmou, " Réseaux de Petri pour l'évaluation des systèmes redondants ", ESREL/λμ9, La Baule 94.

[Ereau 95] Jean-François Ereau et Malecka Saleman: " Dynamic fault trees based on synchronised Petri nets ", ESS'95, Erlangen.

[Ereau 96] Jean-François Ereau et Malecka Saleman: " Modelling and Simulation of a Satellite Constellation based on Petri Nets ", Annual Reliability and Maintenability Symposium, Proceedings 1996.

[Ereau 97] Jean-François Ereau : " Réseaux de Petri pour l'étude de la disponibilité opérationnelle des systèmes spatiaux en phases d'avant projet ", thèse présentée au LAAS le 28 novembre 1997.

[Florin 85] G. Florin et S. Natkin: “ Les réseaux de Petri stochastiques ”, Techniques et Sciences Informatiques, vol. 4, n°1, 1985.

[Fota 97] Nicolae Fota : “ Spécification et Construction Incrémentale de Modèles de Sûreté de Fonctionnement - Application au CAUTRA ”, thèse présentée au LAAS, 1997.

[Gardiner 85] C.W. Gardiner : “Handbook of Stochastic Methods”, Springer Verlag, Berlin, 1985

[Garnier] Robert Garnier, « Une méthode efficace d’acceleration de la simulation des réseaux de Petri stochastiques », thèse de l’Université de Bordeaux 1, soutenue le 29 juin 1998.

[Garret 93] C.J. Garret, S.B. Guarro and G.E. Apostolakis : "Development of a methodology for assessing the safety of embedded software systems", American institute of Aeronautics and Astronautics, 1993.

[Gehlot 92] V. Gehlot, “Aproof theoric approach to semantics of concurrency”, PhD thesis, University of Pennsylvania, 1992.

[Genrich 98] Hartmann J. Genrich, Ines Schuart : « Modelling and verification of Hybrid systems using hierarchical coloured Petri nets », ADPM’98, Reims, mars 1998.

[Girard 87] J.Y. Girard, « Linear Logic », Theoretical Computer Science, 50, 1987, p.1-102.

[Girault 97] F. Girault, « Formalisation en Logique Linéaire du fonctionnement des réseaux de Petri », Thèse de Doctorat, N°2870, Université Paul Sabatier, Toulouse.

[Guarro 84] Sergio Guarro and David Okrent : "The logic Flowgraph : a new approach to process failure modelling and diagnosis for disturbance analysis applications", Nuclear Technology, vol 67, 1984.

[Harel 87] D. Harel, "Statecharts: A visual Formalism for complex systems", The Science of Computer Programming, 1987, 8, pp.231-274.

[Hénault 96] Valéry Hénault: “ Méthodologie de développement des systèmes électroniques embarqués automobiles, matériels et logiciels, sûrs de fonctionnement ”, thèse présentée à l’IRESTE, septembre 1996.

[Hochon 98] Jean-Claude Hochon, Ronan Champagnat, Robert Valette, “Modélisation et simulation hybride à l'aide des réseaux de Petri Prédicats-Transitions couplés à des équations algébro-différentielles”, Actes du 4e Colloque Africain sur la Recherche en Informatique, CARI'98, Dakar (Sénégal), 12-15 Oct. 1998, p.737-749.

[Jampi 01] Jampi D., Détermination d’une méthodologie d’aide à la conception d’un système de contrôle commande numériques sûrs de fonctionnement, Thèse de doctorat, Institut National Polytechnique de Lorraine et au Centre de Recherche en Automatique de Nancy, 2001.

[Jensen 92] Kurt Jensen : "Coloured Petri Nets - Basic Concepts, Analysis Methods and Practical Use". Vol. 1, Basic Concepts. EATCS Monographs on Theoretical Computer Science, Springer-Verlag, 1992.

[Jensen 94] Kurt Jensen : "Coloured Petri Nets - Basic Concepts, Analysis Methods and Practical Use". Vol. 2, Analysis Methods. Monographs in Theoretical Computer Science. Springer-Verlag,1994.

[Jensen 97a] Jensen, K.; Christensen, S.; Huber, P.; Holla, M. (1997) Design/CPN Reference Manual. Computer Science Department, University of Aarhus, Denmark. On-line http //www.daimi.aau.dk/designCPN/.

[Jensen 97b] Kurt Jensen : "Coloured Petri Nets - Basic Concepts, Analysis Methods and Practical Use". Vol. 3, Practical Use. Monographs in Theoretical Computer Science. Springer-Verlag,1997.

[Juanole 91] G.Juanole and Y. Atamna : "Dealing with Arbitrary time distributions with the stochastic and timed Petri net model- Application to queuing systems", Proceedings of PNPM, 1991.

[Kassaagi 01] M.O. Kassaagi, "Caractérisation expérimentale du comportements des conducteurs en situation d'urgence pour la specification de systèmes de sécurité active", thèse de l'Ecole Centrale Paris, 11 juillet 2001.

[Kehren 03] C. Kehren et C. Seguin, "Evaluation qualitative de systèmes physiques pour la sûreté de fonctionnement", Journées *FAC'2003* : *Formalisation des Activités Concurrentes*, les 12 et 13 mars 2003 à l'IRIT, Toulouse.

[Khalfaoui 00] Sarhane Khalfaoui, "Modélisation et validation par simulation des systèmes hybrides", rapport de DEA, INP-ENSEEIHT, septembre 2000.

[Khalfaoui 01b] S. Khalfaoui, E. Guilhem, H. Demmou, N. Rivières, " Extraction de scénarios critiques à partir d'un modèle RdP à l'aide de la logique Linéaire ", MSR'2001 Modélisation des systèmes réactifs, 17-19 Octobre 2001, Toulouse, France p. 409-424. Rapport LAAS No01126.

[Khalfaoui 01b] S. Khalfaoui, E. Guilhem, H. Demmou, R. Valette, " Modeling critical mechatronic systems with Petri Nets and feared scenarios derivation ", ECM2S5 5th Workshop on Electronics, Control, Modeling, Measurement and Signals, 30-31 May, 1er Juin, 2001, Toulouse, France p. 55-59. Rapport LAAS No01027.

[Khalfaoui 02] S. Khalfaoui, E. Guilhem E., H. Demmou, R. Valette, «A method for deriving critical scenarios in mechatronic systems», λμ13, Europeen Conference on System Dependability and Safety, March 18-20, 2002, Lyon, France.

[Khalfaoui 02a] S. Khalfaoui, H. Demmou, E. Guilhem, R. Valette, « An algorithm for deriving critical scenarios in mechatronic systems », 2002 IEEE International Conference on Systems Man and Cybernetics (SMC'02), Hammamet (Tunisie), 6-9 Octobre 2002. Rapport LAAS No02257.

[Khalfaoui 02b] S. Khalfaoui, E. Guilhem, H. Demmou, R. Valette, " Une méthode pour obtenir des scénarios critiques dans les systèmes mécatroniques ", Colloque Européen de Sûreté de Fonctionnement (lm13), Palais des Congrès - Lyon - France - 18 au 21 Mars 2002. Rapport LAAS No01614.

[Khalfaoui 02c] S. Khalfaoui, E. Guilhem, H. Demmou, R. Valette, " Une méthode pour obtenir des scénarios critiques dans les systèmes mécatroniques ", 3ème Congrès des Doctorants de l'Ecole Doctorale Systèmes, Toulouse (France), 22-23 Mai 2002.

[Kristensen 98] Lars Michael Kristensen and Antti Valmari : « Finding stubborn sets of coloured Petri nets without unfolfing », ICATPN'98, Lisbonne, Portugal, 22-26 juin 1998

[Kunzle 97] Luis Allan Künzle, "Raisonnement temporel basé sur les réseaux de Petri pour des systèmes manipulant des resources", Thèse de Doctorat de l'Université Paul Sabatier de Toulouse, soutenue le 29 septembre 1997.

[Labeau 00] P.E. Labeau, C. Smidts et S. Swaminathan, « Dynamic reliability : towards an integrated platform for probabilistic risk assessment », Reliability Engineering and System Safety 68, pp. 219-254, 2000.

[Labeau 02a] P.E. Labeau et C. Kermisch, "Approche dynamique de la fiabilité des systèmes", Rapport MNFD 2002-10, Service de Métrologie Nucléaire, Université Libre de Bruxelles (Belgique), novembre 2002.

[Labeau 02b] P.E. Labeau et C. Kermisch, "Approche dynamique de la fiabilité des systèmes", Rapport MNFD 2002-07, Service de Métrologie Nucléaire, Université Libre de Bruxelles (Belgique), juillet 2002.

[Labeau 03] P.E. Labeau et C. Kermisch, "Approche dynamique de la fiabilité des systèmes", Rapport MNFD 2002-03, Service de Métrologie Nucléaire, Université Libre de Bruxelles (Belgique), mars 2002.

[Laprie 96] J. C. Laprie, J. Arlat, J. P. Blanquart, A. Costes, Y. Crouzet, Y. Deswarte, J. C. Fabre, H. Guillermain, M. Kaâniche, K. Kanoun, C. Mazet, D. Powell, C. Rabéjac et P. Thévenod : " Guide de la sûreté de fonctionnement ", ISBN 2-85428-382-1, Cépaduès-Editions, Janvier 1996.

[Lefort 98] Arnaud Lefort : " Les Hypernets : un outil de modélisation et de spécification ", thèse présentée au LAIL, le 10 juillet 1998.

[Leroy 92] Alain Leroy et Jean Pierre Signoret: " Le risque technologique ", Collection " Que sais-je ? ", 1992.

[Marquetty 95] S. Marquetty : " DEVYSE, développement d'un système électronique automobile ", manuel d'utilisation, PSA/DETA, septembre 1995.

[Marsan 84] M.A. Marsan, G. Balbo et G. Conte, "A Class of Generalised Stochastic Petri Nets for the Performance Evaluation of Multiprocessor Systems", Transactions on Computer Systems, vol. 2, n°2, p. 93-122, Association for Computing Machinery, may 1984.

[Marsan 86] M.A. Marsan et G. Chiola, "On Petri Nets with Deterministic and Exponentially Distributed Firing Times", 7th European Workshop on Applications and Theory of Petri Nets, p. 132-145, Springer-Verlag, Lecture Notes in Computer Science 266, "Advances in Petri Nets, 1987", june 1986.

[Marseguerra 01] M. Marseguerra, E. Zio, "Principles on Monte Carlo simulation for application to reliability and availability analysis", tutorials of Esrel 2001.

[Marseguerra 93] Marseguerra M. et Zio E., « Towards dynamic PSA via Monte Carlo methods », Proceedings of Esrel'93, pp. 415-427, 1993.

[Marseguerra 94a] Marseguerra M. et Zio E., « Approaching dynamic reliability by Monte Carlo simulation », Reliability and safety assessment of dynamic process systems, NATO ASI Series 120, Springer Verlag, Berlin, 1994.

[Marseguerra 94b] Marseguerra M. et Zio E., « Improving the efficiency of Monte Carlo methods in PSA by using neural networks », Proceedings of PSAM-II, 025.1-025.8, 1994.

[Marseguerra 94c] Marseguerra M., Nutini M. et Zio E., « Approximate physical modelling in dynamic PSA using artificial neural networks », Reliability Engineering and System Safety 45, pp. 47-56, 1994.

[Marseguerra 95] M. Marseguerra & E. Zio: " The cell-to-boundary method in Monte Carlo based dynamic PSA ", Reliability Engineering & System Safety, Vol. 48, 1995

[Marseguerra 96] M. Marseguerra & E. Zio: " Monte Carlo approach to PSA for dynamic process systems ", Reliability Engineering & System Safety, Vol. 52, 1996

[Moncelet 97] Gilles Moncelet, Hamid Demmou, Mario Paludetto, José Porras : " Modélisation par réseaux de Petri des systèmes mécatroniques du produit automobile pour l'évaluation de la sûreté de fonctionnement ", 2ème congrès Qualité et Sûreté de Fonctionnement, Angers, mars 1997.

[Moncelet 98a] Gilles Moncelet, « Application des réseaux de Petri à l'évaluation de la sûreté de fonctionnement des systèmes mécatroniques du monde automobile », Thèse de Doctorat, N°3076, Université Paul Sabatier, Toulouse, 9 octobre 1998.

[Moncelet 98b] Gilles Moncelet, Søren Christensen, Hamid Demmou, Mario Paludetto, José Porras : " Estimation de la sûreté de fonctionnement d'un système mecatronique simple à l'aide des Réseaux de Petri colorés », ADPM'98, Reims, mars 1998.

[Moncelet 98c] Gilles Moncelet, Søren Christensen, Hamid Demmou, Mario Paludetto, José Porras : " Qualitative an quantitative dependability evaluation of a simple mechatronic system using Coloured Petri Nets », Workshop on practical use of coloured Petri nets and DesignCPN, Aarhus, Danemark, Juin 98.

[Moncelet 98d] Gilles Moncelet, Søren Christensen, Hamid Demmou, Mario Paludetto, José Porras : " Application des Réseaux de Petri colorés à l'étude de la sûreté de fonctionnement d'un système mecatronique de l'industrie automobile », λμ11, Arcachon, Octobre 98.

[Murata 89] Tadao Murata: " Petri nets : properties, analysis and applications ", Proceedings of the IEEE, vol 77, n°4, avril 1989.

[Natkin 80] S. Natkin : "Les réseaux de Petri stochastiques", Thèse de Docteur Ingénieur CNAM, Paris, Juin 1980.

[Nouillant 02] C. Nouillant, D. Jampi, F. Assadian, X. Moreau et A. Oustaloup : « Sûreté de Fonctionnement de la Commande Hybride d'un ABS » ; Congrès CIFA 2002, Nantes, France.

[Pagès 80] A. Pagès et M.Gondran: " Fiabilité des systèmes ", collection de la Direction des Etudes et Recherche d'Electricité de France, 1980.

[Perez 96] Daniel Perez, Robert Garnier, Marcel Chevalier : " Une méthode complète d'accélération de la simulation de Monte-Carlo d'un réseau de Petri stochastique généralisé pour des systèmes non markoviens ", ESREL'96.

[Pnuelli 84] A. Pnuelli, "In transition for global to modular temporal reasoning about programs", In K.R. Apt, ed., *Logics and Models of Concurrent Systems*, NATO ASI 13. Springer, 1984.

[Pradin 03] B. Pradin-Chézalviel, R. Valette, "Réseaux de Petri et Logique Linéaire", Chapitre 6 de l'ouvrage "Vérification et mise en oeuvre des réseaux de Petri" (sous la direction de Michel Diaz), Editions Hermès, Traité IC2 (Information-Commande-Communication) ISBN 2-7462-0445-2, 2003, p.209-229.

[Pradin 99] B. Pradin-Chézalviel, R. Valette, L.A Künzle, « Scenario duration characterization of t-timed Petri nets using linear logic », IEEE PNPM'99, 8th International Workshop on Petri Nets and Performance Models, Zaragoza, Spain, September 6-10, 1999, p.208-217.

[Pradin 99a] Brigitte Pradin-Chézalviel, Luis Allan Künzle, François Girault, Robert Valette, "Calculating duration of concurrent scenarios in time Petri nets", APII-JESA Journal Européen des Systèmes Automatisés, Volume 33, n 8-9/1999, p.943-958.

[Pradin 99b] B. Pradin-Chézalviel, R. Valette, L.A. Künzle, "Scenario duration characterization of t-timed Petri nets using linear logic", IEEE PNPM'99, 8th International Workshop on Petri Nets and Performance Models, Zaragoza, Spain, September 6-10, 1999, p.208-217.

[Pradin 99c] B. Pradin-Chézalviel, L. A. Künzle, F. Girault, R. Valette, "Evaluation temporelle de scénario de réseau de Petri incluant du parallélisme", 2e Conférence MSR'99, Modélisation des Systèmes Réactifs, 24-25 mars 1999, Cachan, Edition Hermès, p.131-140.

[Rauzy 97] Antoine Rauzy et Yves Dutuit : " Exact and truncated computations of prime implicants of coherent and non coherent fault trees within Aralia ", Reliability Engineering & System Safety, Vol. 58, 1997

[Rivière 00] N. Riviere, « Modélisation des systèmes coopératifs », Rapport de DEA, LAAS, Toulouse.

[Schnoebelen 99] "Vérification de Logiciels: Techniques et outils du model-checking", ouvrage collectif, coordination P. Schnoebelen, Vuibert, Paris, 1999, ISBN 2-7117-8646-3

[Signoret 96] Jean-Pierre Signoret : " Modélisation et simulation dans le domaine de la sûreté de fonctionnement ", REE, n°8, septembre 1996.

[Sinnamon 97] R.M. Sinnamon & J.D. Andrews : " New approaches to evaluating fault trees ", Reliability Engineering & System Safety, Vol. 58, 1997

[Siu 94] N. Siu : " Risk assessement for dynamic systems : An overview ", Reliability Engineering & System Safety, Vol. 43, 1994

[Swaminathan 99a] Swaminathan S. et Smidts C., « The event sequence diagram framework for dynamic PRA », Reliability Engineering and System Safety 63, pp. 73–90, 1999.

[Swaminathan 99b] Swaminathan S. et Smidts C., « The mathematical formulation for the event sequence diagram framework », Reliability Engineering and System Safety 65, pp. 103-118, 1999.

[Valantin 98] C. Valantin-Roubinet : « DAE supervised by Petri nets. The example of a gaz storage », ADPM'98, Reims, mars 1998.

[Valentin 93] C. Valentin, "Contribution à la modélisation et à la conduite des procédés mixtes (continues-discrets) : application à l'industrie papetière", Thèse de l'Institut Polytechnique de Grenoble, soutenue au Laboratoire d'Automatique de Grenoble, France, février 1993.

[Valette 92] R. Valette : "Les réseaux de Petri", support de cours, France, 1992.

[Villani 02] E. Villani, J.C. Pascal, P.E. Miyagi, R. Valette, "Apport d'une approche à objets fondée sur les réseaux de Petri à l'analyse des systèmes hybrides", CIFA 2002, Nantes, France, 8-10 juillet 2002, p. 659-664.

[Villani 03] E. Villani, J.C. Pascal, P.E. Miyagi, R. Valette, "Differential predicate transition Petri nets and objects, an aid for proving properties in hybrid systems", ADHS 03, IFAC Conference on Analysis and Design of Hybrid Systems, Saint-Malo, France, June 16-18, 2003, p.117-122.

[Villemeur 88] Alain Villemeur : " Sureté de fonctionnement des systèmes industriels ", collection de la Direction des Etudes et Recherche d'Electricité de France, 1988.

[Zanne 90] C. Zanne et J.F. Aubry : "Une méthode de conception des systèmes de commande numérique pour une classe de processus rapides", Sûreté de fonctionnement, Vol n°24, n°5, pp 435-456.

[Zanne 95] Christian Zanne : " Contribution à la conception des dispositifs de commande pour les systèmes dynamiques hybrides ", rapport pour l'habilitation à diriger des recherches, mars 1995.

[Zaytoon 01] "Systèmes dynamiques hybrides", ouvrage collectif sous la direction de Janan Zaytoon, collection Hermes Science, ISBN 2-7462-0247-6.

[Ziegler 96] Christian Ziegler : " Sûreté de fonctionnement d'architectures informatiques embarquées sur automobile ", thèse présentée au LAAS le 12 juillet 1996.

Annexe A1. Conflit de jetons et de transitions : duplication de l'arbre de preuve

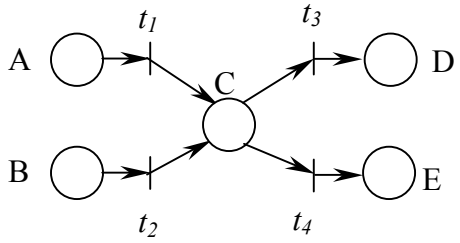


Figure 2.9. **Exemple de réseau de Petri avec conflits**

Preuve du séquent $A \otimes B, t_1, t_2, t_3, t_4 \vdash D \otimes E$ **:**

Il y a un conflit de transitions associé à un conflit de jetons.

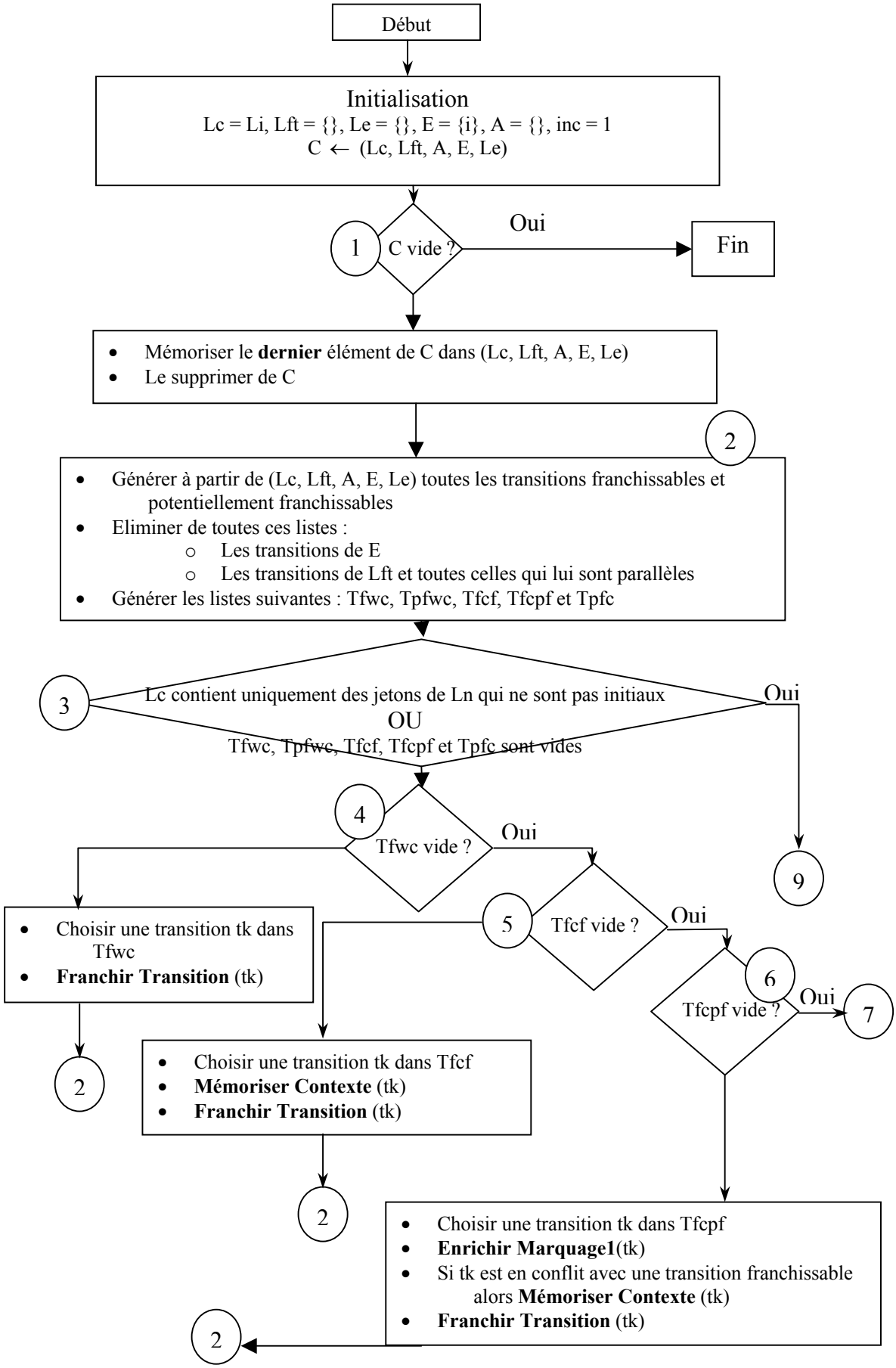
$$\frac{\dfrac{\vdots}{\textit{Séquent } 1} \quad \dfrac{\vdots}{\textit{Séquent } 2}}{C(t_1^{\,1}), C(t_2^{\,1}), C \multimap D, C \multimap E \vdash D(F^1) \otimes E(F^2)}\ \textit{Conflits}$$

$$\frac{\dfrac{\dfrac{}{A(I^1) \vdash A(t_1^{\,1})}\ id \quad \dfrac{\dfrac{}{B(I^2) \vdash B(t_2^{\,1})}\ id \quad \dfrac{\vdots}{C(t_1^{\,1}), C(t_2^{\,1}), t_3, t_4 \vdash D(F^1) \otimes E(F^2)}}{B(I^2), C(t_1^{\,1}), B \multimap C, t_3, t_4 \vdash D(F^1) \otimes E(F^2)} \multimap L(t_2^{\,1})}{A(I^1), B(I^2), A \multimap C, t_2, t_3, t_4 \vdash D(F^1) \otimes E(F^2)} \multimap L(t_1^{\,1})}{A(I^1) \otimes B(I^2), t_1, t_2, t_3, t_4 \vdash D(F^1) \otimes E(F^2)} \otimes L$$

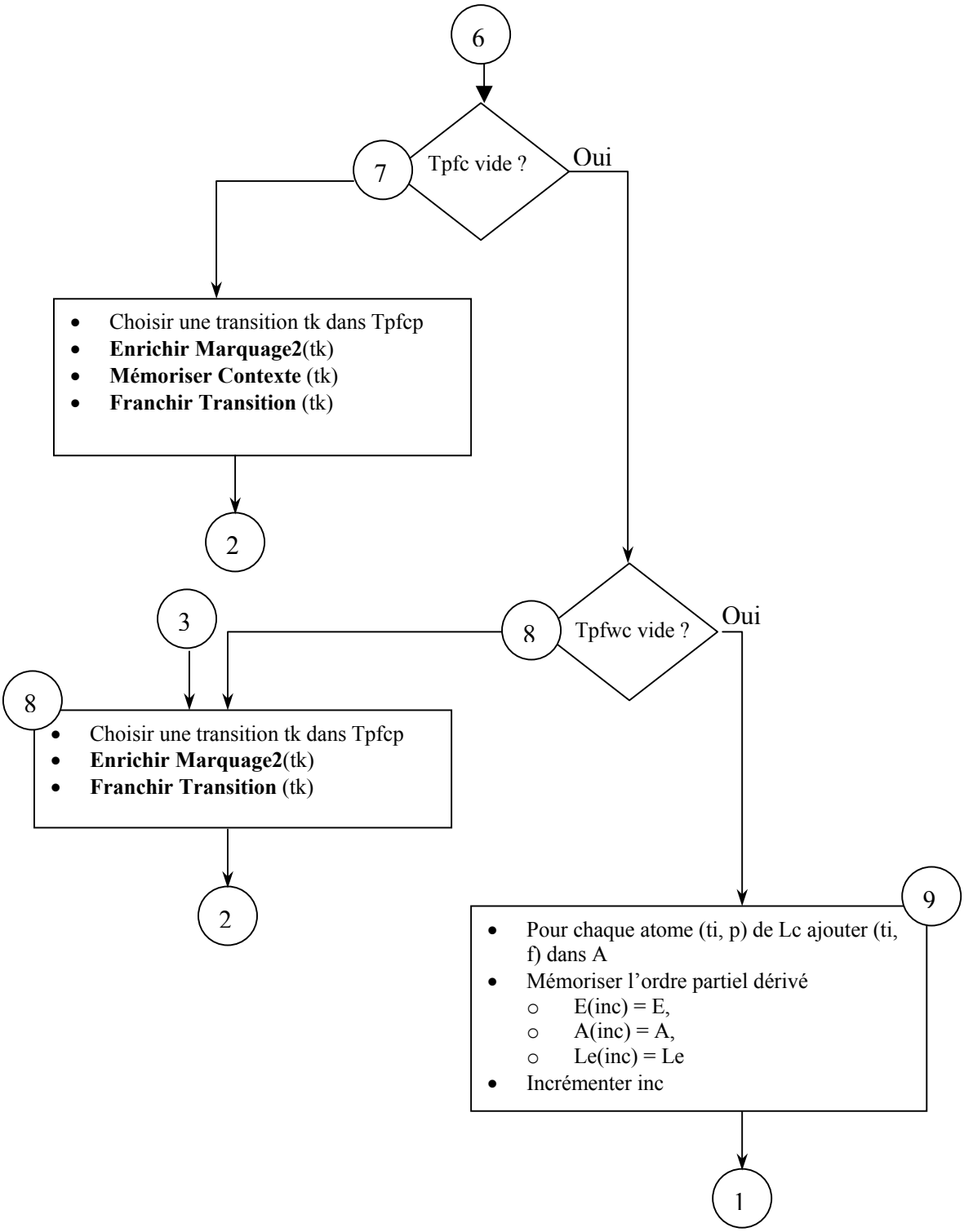
$$\frac{\dfrac{}{C(t_1^{\,1}) \vdash C(t_3^{\,1})}\ id \quad \dfrac{\vdots}{C(t_2^{\,1}), D(t_3^{\,1}), t_4 \vdash D(F^1) \otimes E(F^2)}}{\textit{Séquent } 1} \multimap L(t_3^{\,1})$$

$$\frac{\dfrac{}{C(t_1^{\,1}) \vdash C(t_4^{\,1})}\ id \quad \dfrac{\vdots}{C(t_2^{\,1}), E(t_4^{\,1}), t_3 \vdash D(F^1) \otimes E(F^2)}}{\textit{Séquent } 2} \multimap L(t_4^{\,1})$$

Annexe A2. Organigramme ½



Annexe A2. Organigramme 2/2



Méthode de recherche des scénarios redoutés pour l'évaluation de la Sûreté de Fonctionnement des systèmes mécatroniques du monde automobile

Le nombre croissant des systèmes électroniques embarqués dans le secteur automobile a considérablement amélioré et diversifié les services rendus par le véhicule. Ces systèmes sont appelés systèmes mécatroniques. Ils intègrent une partie énergétique (mécanique, hydraulique ou électrique) commandée et contrôlée par un calculateur. Leur principal atout est la flexibilité logicielle dont dispose le concepteur pour implémenter de nouvelles fonctions. Toutefois, ceci a contribué à accroître leur complexité et à en diminuer la maîtrise, d'où la nécessité d'effectuer des études de Sûreté de Fonctionnement afin de garantir un bon niveau de sécurité. Par ailleurs, mener de telles études dès la phase de conception permet de diminuer les délais et les coûts de conception en détectant et en corrigeant au plus tôt les erreurs de conception. Actuellement, les études de sécurité prévisionnelle des systèmes automobiles sont réalisées par la méthode des Arbres de Défaillance. Or cette méthode est statique et ne permet pas de prendre en compte les phénomènes temporels liés à leur dynamique de fonctionnement et à leur aspect hybride. C'est dans ce contexte que des recherches sont menées en collaboration entre le groupe PSA Peugeot Citroën et le LAAS visant à développer une méthodologie d'aide à la conception de systèmes mécatroniques sûrs de fonctionnement.

Mon projet de thèse se focalise sur l'analyse qualitative de la sécurité des systèmes mécatroniques en vue de l'obtention des scénarios redoutés. La connaissance de ces scénarios permet d'évaluer leurs probabilités d'occurrence et de valider les lois de reconfiguration pour orienter le choix des concepteurs quant aux différents types d'architectures possibles proposés pour le système. Nous avons développé une méthode de recherche des scénarios redoutés basée sur la modélisation préalable d'un système mécatronique sous la forme d'un Réseau de Petri et d'un ensemble d'équations différentielles. Cette modélisation hybride présente l'avantage de séparer clairement les aspects discrets et continus. Ceci nous permet une analyse logique (fondée sur la logique Linéaire) des causalités résultant des changements d'états. Grâce à cette analyse, il est possible à partir d'un état redouté de remonter les chaînes de causalité et de mettre ainsi en évidence tous les scénarios possibles conduisant à une situation critique. Chaque scénario est donné sous la forme d'un ordre partiel entre les événements nécessaires à l'apparition de l'état redouté. L'originalité de notre approche est qu'elle n'implique pas une énumération brutale et globale de tous les états accessibles du système. Au contraire elle permet de se focaliser sur le voisinage de l'état redouté en faisant une énumération locale d'états partiels. Autrement dit, nous ne considérons que les états des composants directement impliqués dans l'apparition de l'état redouté. Nous avons enfin élaboré un algorithme automatisant la recherche des scénarios redoutés et nous l'avons appliqué sur deux exemples simples de systèmes mécatroniques.

Mots clés : Sûreté de Fonctionnement, systèmes mécatroniques, systèmes dynamiques hybrides, scénarios redoutés, réseaux de Petri, logique Linéaire.

A method for deriving feared scenarios for dependability evaluation of automotive mechatronic systems

New cars include more and more electronic embedded systems that enhance considerably their performances. These systems are composed of mechanic, hydraulic, electronic and computing parts, and called mechatronic systems. For the designer, their benefit lies in the large software flexibility to implement new functions. However, this flexibility contributes to increase their complexity and may add safety problems. That is why it is necessary to make reliability studies in order to guaranty a high safety level. In order to reduce costs and duration of the development phase, these studies must be done as soon as possible. In fact, we detect and correct conception errors in the early design stage. Classical methods of safety, as fault trees, are not sufficient to deal with this kind of complex and hybrid systems because they are inherently static. This motivates the car maker PSA Peugeot Citroën and the Laboratory LAAS to make together research about a new methodology for designing safe mechatronic systems.

My thesis focuses on qualitative analysis of mechatronic systems safety in order to derive feared scenarios. Identifying these scenarios allows us to evaluate their occurrence probabilities and helps then designers to select the safe architecture. The hybrid aspect of mechatronic systems leads us to choose a model that associates Petri nets and differential equations. The Petri net model describes the operation modes, the failures and the reconfiguration mechanisms. The differential equations represent the evolution of continuous variables of the energetic part of the system. Based on a clear separation between continuous and discrete parts, this model allows us to make a causality based analysis (thanks to Linear logic) to point out the sequences of actions and state changes that lead to a feared situation. The advantage of our approach is that we can express partial order of transition firings and focus the search on the parts of the model that are interesting for safety analysis, without generating the reachability graph. We avoid then the combinatorial explosion problem. Finally, we developed an algorithm which makes automatic derivation of feared scenarios.

Keywords : dependability, mechatronic systems, dynamic hybrid systems, feared scenarios, Petri nets, Linear logic.