

## Chapitre 5

# Moyens de la sûreté de fonctionnement

### 1 Introduction

Il s'agit dans ce court chapitre de faire un tour d'horizon des diverses techniques et des outils développés en Sciences pour l'Ingénieur pour permettre d'assurer une certaine sûreté de fonctionnement. Nous n'aborderons pas les aspects concernant la gestion des projets et de l'innovation qui ont été présentés au chapitre précédent.

Dans ce cadre, les concepts de surveillance et de supervision sont essentiels. Parler de supervision implique une organisation hiérarchique du système de commande du système de production. Cette fonction se positionne entre la gestion prévisionnelle et la commande en temps réel. Elle implique que l'on effectue des actions sur le système, à la fois dans le but d'optimiser son fonctionnement (mise en œuvre d'un plan optimal de fabrication par exemple) et d'assurer la sûreté (reconfiguration ou arrêt d'urgence en cas d'anomalie). La surveillance peut, quant à elle, être mise en œuvre à tous les niveaux de la commande et elle a une connotation plus passive. On peut surveiller pour détecter des anomalies sans nécessairement agir directement sur le système. Elle constitue dans ce cas une aide à l'opérateur humain, un outil qu'il peut utiliser pour mieux remplir sa tâche.

Pour organiser les différentes techniques et outils développés pour assurer la sûreté de fonctionnement des systèmes de production, il est possible de se fonder sur le vocabulaire utilisé dans le cycle de vie du logiciel. En effet, la supervision et la surveillance vont bien se traduire par du logiciel associé au logiciel de commande. Nous allons alors rencontrer des techniques, des méthodes et des outils concernant:

- la spécification des besoins,

- la conception de solutions répondant aux besoins,
- l'analyse et l'évaluation des performances des solutions retenues de façon à choisir entre plusieurs solutions possibles,
- la programmation, d'une façon plus générale la mise en œuvre des solutions retenues et leur vérification (y compris le test pour la certification du logiciel final),
- la maintenance et l'évolution au cours du temps.

Schématiquement, il est également possible de regrouper techniques et outils suivant deux points de vue : en différé (off-line) et en direct (on-line). Par exemple le développement de méthodes de spécification et de conception formelles du logiciel de supervision et sa certification (montrer qu'il est sans faute et qu'il correspond bien au problème posé) relève du point de vue "off-line". De même, ce qui concerne les politiques de maintenance préventive est "off-line". Par contre, tout ce qui concerne la mise en œuvre informatique des politiques d'actions de compensation et de reprise en cas de défaillance relève du point de vue "on-line".

Enfin, un troisième point de vue peut être adopté, celui des fonctions remplies par un système de supervision et de surveillance. Nous avons alors les fonctions suivantes :

- le pilotage en temps réel du flux de produit,
- le pilotage en temps réel des ressources,
- la sécurité opérationnelle.

Le pilotage en temps réel du flux de produit met en œuvre le plan de fabrication prévisionnel. Il correspond donc à une fonction de surveillance et de réactivité vis-à-vis de la politique prévisionnelle de gestion de l'atelier. Il définit, en fonction de l'état courant du système de production, ce qu'il est souhaitable de faire. Il donne ainsi des objectifs pour les fonctions de commande de plus bas niveau dans des architectures hiérarchiques.

Le pilotage en temps réel des ressources correspond à une fonction de surveillance et de réactivité vis-à-vis des ressources de l'atelier c'est-à-dire des machines, des outils, des opérateurs, *etc.* Il est chargé de mettre à jour l'état courant du système de production, état nécessaire au pilotage en temps réel du flux de produit puisqu'il définit ce que le système peut faire. Pendant cette mise à jour, il détecte les défaillances des ressources et peut offrir des fonctions de diagnostic et de reconfiguration.

La sécurité opérationnelle [Niel 92] est assurée par un ensemble de contraintes à vérifier. Ces contraintes dépendent de l'état courant. Ces contraintes ont pour but d'empêcher le système d'atteindre les états dangereux, soit en interdisant certains changements d'état, soit en forçant des changements d'état pour mettre en œuvre des

procédures d'arrêt d'urgence. On définit, en fonction de son état courant, ce que le système ne doit pas faire.

Ces trois fonctions sont étroitement imbriquées puisqu'elles partagent toutes les trois l'information sur l'état du système. Le principe utilisé pour assurer la sûreté de fonctionnement est le même : on vérifie, lors de chaque événement, que le comportement du système physique est compatible avec celui d'un modèle. La représentation de l'état courant est au cœur des modèles utilisés dans les trois fonctions, mais cette représentation est complétée chaque fois par des informations différentes. Nous allons les détailler, les unes après les autres en prenant en considération leur cycle de vie, puis nous introduirons les outils mathématiques, les concepts et les méthodes couramment utilisés.

## 2 Les fonctions à remplir

### 2.1. *Pilotage en temps réel du flux de produit*

Il s'agit donc de mettre en œuvre une politique de conduite définie par la gestion prévisionnelle. Cette politique peut être extrêmement précise et être donnée sous la forme d'un plan de fabrication pour lequel une date de début d'opération est associée à chaque opération ou bien sous la forme d'un ensemble de règles de décision à suivre. On peut, par exemple, décider que dans une file d'attente à l'entrée d'une machine, on prend toujours l'opération dont la durée est la plus courte, ou bien celle de plus grande priorité, ou bien celle qui correspond à un produit dont la date de livraison est la plus proche. Les contraintes de sûreté sont, à ce niveau, de nature plutôt économique. Les risques concernent le non respect des délais de livraison et la possibilité de perdre des parts de marché. Le modèle permettant d'assurer la sûreté est formé de l'état courant de l'atelier (produits en cours et état des ressources, c'est-à-dire des machines) complété par le plan de fabrication et les règles de décision, ainsi que par des informations sur les stocks de matières premières et sur les demandes et les caractéristiques des clients. Voyons les diverses étapes du cycle de vie de cette fonction.

A l'étape de spécification des besoins il faut analyser les besoins de flexibilité et de réactivité. Il est clair qu'un plan de fabrication spécifié de façon extrêmement précise permet de vérifier exactement la tenue des délais de livraison. Mais il sera extrêmement difficile de le respecter. Si les défaillances sont fréquentes, on peut se contenter d'un ensemble de règles et de travailler de façon réactive. Les règles seront toujours respectées, mais, en général, les délais de livraisons ne le seront pas. Ce qui est peut-être plus grave, c'est qu'il ne sera pas possible de détecter le retard pris, ni de fournir aux clients une nouvelle date de livraison. C'est pourquoi une combinaison des deux approches (date de livraison et règles de décision) est souvent

retenue. On voit bien que la manière de générer un plan puis de contrôler son exécution pas à pas et de réagir aux incidents est essentielle vis-à-vis de la qualité du service rendu aux clients.

A l'étape de conception, il faut élaborer effectivement les politiques de décision en temps réel. Ces politiques vont utiliser la flexibilité, c'est-à-dire l'indéterminisme résiduel du plan de fabrication, pour permettre de réagir aux imprévus. Il faudra une étape de détection et éventuellement de diagnostic pour déterminer les situations où il est possible de rester dans le cadre du plan prévisionnel des situations dans lesquels la réactivité implique un appel à la gestion de production pour élaborer un nouveau plan.

L'analyse des performances doit permettre de comparer diverses politiques de décision vis-à-vis d'un ensemble de critères: minimiser le nombre de remises en question de la politique prévisionnelle (robustesse), coût (efficacité), simplicité de mise en œuvre, adéquation avec les modèles cognitifs des opérateurs humains. Dans le cas où le système n'est pas entièrement discret (lots de matières premières continues par exemple), il peut être nécessaire de valider la politique prévisionnelle par simulation hybride pour vérifier que l'évolution des variables continues est compatible avec les commandes discrètes et la qualité requise pour le produit.

La programmation va poser des problèmes de nature informatique : interaction avec des bases de données techniques, avec la traçabilité des produits (l'histoire individuelle de chaque produit nécessaire au service qualité), *etc.*

La maintenance et l'évolution au cours du temps posent le problème de savoir comment la politique prévisionnelle, la politique de décision en temps réel, les gammes de fabrication *etc.*, pourront être modifiées.

## **2.2. Pilotage en temps réel des ressources**

Comme nous l'avons vu, il s'agit de travaux s'attachant à l'interaction avec la commande locale du système de production et qui visent à surveiller le comportement des ressources (machines, outils, systèmes de transport et de manutention *etc.*) et éventuellement de certaines parties de leurs commandes. En plus de la représentation de l'état courant de l'atelier, le modèle comprend des informations sur les ressources. Par exemple, le comportement des machines indépendamment de la commande en cours, la disposition des machines dans l'atelier, leurs caractéristiques *etc.* Les contraintes de sûreté sont, à ce niveau, de nature plutôt mécanique. Il faut éviter toute détérioration des machines, assurer correctement leur maintenance *etc.*

La spécification des besoins implique une analyse des défaillances et de leurs effets, c'est-à-dire en particulier de leur propagation. Leur criticité englobant la durée d'indisponibilité, la probabilité d'occurrence et la probabilité de non détection est évidemment un paramètre important. L'analyse des erreurs humaines est également nécessaire car elle peut être la cause première d'un mauvais fonctionnement du système. Il faut aussi étudier les divers modes de marche et les diverses façons de reprendre un fonctionnement nominal après une défaillance. Enfin, l'analyse des politiques de maintenance préventives ou uniquement curatives a également un impact très important.

La conception des solutions doit répondre aux trois points essentiels de la surveillance : détecter la défaillance, faire un diagnostic, détailler les mécanismes de reprise ou de reconfiguration. Ceci nécessite l'élaboration de modèles de bon fonctionnement et, également de modèles de propagation de défaillances dans la majorité des cas. La détection est pratiquement toujours fondée sur une comparaison, en temps réel, du comportement effectif du système surveillé avec celui d'un modèle de bon comportement. A un niveau très bas (traitement du signal), ce modèle peut être simplement un ensemble d'hypothèses sur le comportement stochastique d'un signal. Dans le cadre de la productique, on se place en général à un niveau plus agrégé et l'on surveille l'apparition d'événements. La détection consiste à remarquer une incohérence entre le comportement du système physique et celui du modèle. Une défaillance se traduit soit par l'apparition d'un événement non attendu, soit par la non apparition (avant la date au plus tard) d'un événement attendu [Combacau 91, Toguyeni 92].

Suivant l'instant auquel est effectuée la comparaison, diverses approches peuvent être proposées. Si la comparaison est faite au moment où une commande va être envoyée, la notion de filtre de commande est utilisée. Si elle est faite au moment de la réception d'une nouvelle information, c'est la notion de filtrage-perception (mise à jour d'une base de connaissance en IA). Si elle est faite dans les deux cas c'est la notion de modèle de référence.

Le diagnostic, lui, est souvent fondé sur la recherche d'une cohérence entre le comportement (ou l'état) du système physique et celui d'un modèle de défaillance correspondant à des connaissances profondes. Une autre approche du diagnostic consiste à formaliser la connaissance superficielle des opérateurs sous la forme d'un ensemble de règles. Il est enfin possible de travailler en utilisant des techniques d'apprentissage, soit sous la forme de réseaux de neurones, soit en synthétisant un automate reconnaissant les scénarios significatifs (chroniques).

Les modèles de propagation de fautes et de défaillances peuvent correspondre à diverses visions. Il peut s'agir de modèles structurels [Chaillet 95] décrivant l'architecture des systèmes (vision statique), de modèles comportementaux décrivant

des enchaînements d'événements et d'activités (sous la forme d'automates par exemple), de modèles fonctionnels décrivant des transformations de valeurs ou d'informations effectuées par les éléments du système (fonctions algébro-différentielles par exemple), ou bien de modèles causaux décrivant les événements observés qui sont des conséquences directes ou indirectes des fautes (les arcs d'un graphe causal peuvent être interprétés comme des implications logiques).

La conception des stratégies de reprise [Berruet 98] (actions de maintenance curative, de compensation ou de reconfiguration) est un vaste problème. Comme pour la phase de diagnostic il faut travailler à partir de modèles du système en présence de défaillances ce qui provoque une explosion combinatoire difficile à maîtriser. Ces modèles permettent de connaître les services que le système peut produire dans une situation donnée, mais il est absolument nécessaire de faire également intervenir les nouveaux objectifs de production qui ont pu être modifiés après la défaillance (paragraphe 2.1) et les nouvelles contraintes de sécurité opérationnelle (paragraphe 2.3).

L'évaluation des performances consiste à vérifier si les politiques de détection, de diagnostic, de fonctionnement dégradé, de reprise et de maintenance curative sont efficaces. Par exemple, il faut vérifier si les défaillances de courte durée (les micro pannes d'un système de manutention de canettes par exemple) auront un impact important ou non sur la production. En ajoutant des considérations de coût, elle permet de choisir entre différentes solutions qui assurent toutes le comportement désiré.

La programmation va impliquer au préalable une définition de l'architecture du système informatique (celui de l'application, pas le système d'exploitation). Comment va s'organiser l'architecture de la commande et celle des fonctions de diagnostic et de reprise? Comment et par l'intermédiaire de quels réseaux locaux, les communications entre les diverses fonctions vont-elles s'effectuer? Quelles sont les informations nécessaires et quand? Comment propager les détections d'anomalies dans une architecture hiérarchisée et distribuée? Comme les modèles et les informations nécessaires pour la commande normale diffèrent de ceux nécessaires au diagnostic et à la reprise, la définition d'une architecture informatique adéquate est particulièrement ardue.

La vérification et la validation (ce peut être une véritable certification dans certains cas) de tout le logiciel utilisé pour la commande est également une tâche essentielle. Il est clair qu'une partie de cette problématique relève de l'informatique. Toutefois, la façon de spécifier les fonctions réalisées par les logiciels dépend fortement de l'application et cela entraîne des spécificités. Par exemple la vérification qu'un programme d'automate donné sous la forme d'un Grafcet correspond bien à la spécification des besoins trouve naturellement sa place en

productique. Dans le cycle de vie en "V", chaque étape (analyse, conception architecturale, *etc.*) comprend également la définition des procédures de test (systèmes, sous-systèmes, unités, *etc.*) et de vérification.

La maintenance du pilotage en temps réel des ressources et la maîtrise des évolutions des modules mettant en œuvre la commande est en forte relation avec ce qui vient d'être présenté. Un logiciel bien spécifié, bien documenté et vérifié formellement sera plus facilement maintenu et modifié. Le choix de politiques de maintenance préventive des ressources (c'est-à-dire des machines) est à prendre en compte lors de l'évaluation des performances.

### ***2.3. La sécurité opérationnelle***

La sécurité opérationnelle [Niel 89] se rapproche du pilotage en temps réel des ressources en de nombreux points. Toutefois, au lieu de chercher à piloter le système physique (l'ensemble des ressources) pour qu'il suive une trajectoire donnée afin d'assurer une production nominale, il faut veiller à ce qu'il ne passe jamais dans certains états jugés extrêmement dangereux, éventuellement pour le système lui-même, mais surtout pour les opérateurs humains et pour l'environnement. Par exemple, la surveillance de la température dans un réacteur peut ne relever que du pilotage des ressources s'il s'agit d'éviter une détérioration légère du réacteur, alors qu'elle relèvera de la sécurité opérationnelle si le contenu du réacteur présente un risque d'explosion mettant en danger les opérateurs humains et l'environnement. Le modèle utilisé complète la représentation de l'état courant par un ensemble de contraintes de sécurité à vérifier. Ces contraintes dépendent de l'état courant (des produits et des ressources). Elles sont de nature variée et elles découlent souvent de considérations juridiques.

Prenons par exemple le cas de l'analyse des besoins, dans le cas de la sécurité opérationnelle il ne sera pas possible d'avoir une approche purement statistique. Les accidents graves sont rares et ne peuvent être comptés comme les défaillances bénignes. Souvent, ce qui mène à un accident, est une succession d'événements redoutés pour lesquels aucune compensation adéquate n'a pu être générée. Les études probabilistes seront également difficiles à mener car il faudrait traiter de probabilités extrêmement faibles (problème des événements rares). Enfin l'impact des erreurs humaines est très important. Des démarches d'analyse sur des modèles qualitatifs, en général graphique, sont souvent nécessaires pour mettre en évidence les scénarios critiques.

La conception de stratégies de compensation après détection de situations dangereuses se pose également différemment. Au lieu de chercher à provoquer certains comportements permettant une reprise d'activité normale, on va se contenter

de chercher à empêcher certaines transitions d'état qui feraient passer le système d'un état dangereux à un état catastrophique. Pour cela, la connaissance des événements contrôlables est indispensable. Un point important à souligner est que le même événement peut avoir des conséquences tout à fait différentes suivant l'état courant du système. Par exemple le franchissement du seuil "température trop élevée" dans un échangeur de chaleur ne sera pas trop dangereux s'il s'agit d'eau, alors que s'il s'agit de lait, il peut se créer un dépôt bouchant l'échangeur très rapidement avec un risque d'explosion. C'est pourquoi la simple utilisation de capteurs d'alarmes pilotant des procédures d'arrêt d'urgence est tout à fait insuffisante. Comme dans le cas du pilotage des ressources, il faut avoir des modèles, détecter les comportements non attendus, analyser leur conséquence vis-à-vis de l'état courant et réagir en conséquence pour interdire certaines progations.

L'explicitation des évolutions interdites est tout à fait indispensable avant toute recherche de reprise. Elle délimite les degrés de liberté restant. S'ils sont suffisant, alors on cherchera une séquence forçant le système vers un mode de fonctionnement compatible avec la nouvelle politique de conduite des flux générée par le module de pilotage et de supervision des flux.

Par ailleurs, la vérification formelle des logiciels est encore plus essentielle que dans le cas précédent. La taille des modules de détection, de diagnostic et de compensation devra donc rester petite pour que leur vérification reste faisable.

### **3. Les aspects théoriques: concepts, modèles, méthodes**

D'une façon générale, les concepts et les méthodes utilisés en productique diffèrent notablement de ceux qui sont utilisés en automatique des systèmes continus. Très globalement, les fondements de nos modèles reposent sur les mathématiques discrètes. Nous manipulons plus souvent des ensembles et des graphes que des équations algébriques, et quand nous écrivons des équations algébriques elles portent plus souvent sur des entiers (ou des rationnels) que sur des réels. Le temps est parfois une variable continue mais il est, lui aussi, souvent discret (suite d'événements partiellement ordonnés, à ne pas confondre avec un temps échantillonné). Le concept de modèle à événements discret (états discrets et temps discret) est celui qui est le plus fréquemment utilisé. La dynamique de nos systèmes s'exprime sous la forme d'automates et non sous celle d'équations différentielles et cette différence est fondamentale. Ceci tendrait à nous rapprocher de l'informatique, mais vis-à-vis de cette discipline, la différence est la suivante: nous supposons que le système d'exploitation et les réseaux locaux sont corrects, et nous utilisons l'informatique pour surveiller le comportement du procédé. La sûreté du système d'exploitation n'est pas au centre de nos préoccupations ; nous la supposons acquise. Nous nous focalisons sur la partie du système informatique correspondant à l'application, c'est-

à-dire mettant en œuvre les fonctions à remplir par le système de commande et de surveillance.

D'autre part, les systèmes que nous représentons sont des systèmes artificiels construits par l'homme et leur surveillance se fait avec l'aide de l'homme et vis-à-vis de l'homme. On se pose des problèmes d'aide à la décision, on analyse les conséquences d'erreurs humaines, la sécurité est définie vis-à-vis de l'homme. La détection, le diagnostic et la compensation des défaillances s'inspirent largement des raisonnements (superficiels ou profonds) des opérateurs humains. La logique est l'aspect des mathématiques qui nous est le plus utile et ceci renforce encore l'importance des mathématiques discrètes dans nos travaux. Cet aspect rapproche nos préoccupations de celles que l'on peut rencontrer en Génie Industriel. Toutefois, bien que nous prenions souvent en compte la présence de l'homme dans le système, les méthodes que nous allons évoquer ici relèvent essentiellement des sciences de l'ingénieur et ne se placent pas dans une optique pluridisciplinaire (sciences humaines et sciences physiques) bien que le besoin en soit ressenti.

Une partie de ces outils de ces concepts et de ces méthodes étant détaillée dans la suite de cet ouvrage, nous allons simplement les lister rapidement.

### **3.1. Outils mathématiques**

Par outil mathématique, nous entendons un outil théorique abstrait, parfaitement formalisé et non spécifiquement dédié à un type de problème.

La théorie des ensembles est bien entendu souvent présente explicitement ou implicitement derrière beaucoup de formalismes utilisés. Toutefois, nous avons souvent besoin de considérer que certains éléments, s'ils peuvent être énumérés, sont néanmoins équivalents et non individualisés. Il faut alors utiliser des multi-ensembles et/ou des monoïdes. Par exemple au lieu de considérer un ensemble de machines (toutes différentes) on peut considérer un ensemble de pools de machines. On sait combien de machines il y a dans un pool donné, mais elles sont considérées comme étant toutes équivalentes au niveau d'abstraction considéré. Les éléments de l'ensemble sont les pools et si un pool  $p_i$  contient trois machines, l'atome  $p_i$  sera répété trois fois dans le multi-ensemble.

La théorie des graphes est également à la base de nombreuses approches. Les graphes servent en particulier à définir formellement des relations entre des éléments (modèles entités-associations par exemple) ou bien à définir des relations d'ordre entre des événements (une opération  $o_i$  doit précéder une opération  $o_j$ ) ou bien des relations de causalité (la défaillance de  $a$  entraîne celle de  $b$ ) ou encore des relations

d'appartenance ou de composition (l'élément  $a$  est formé de  $a_1$  et  $a_2$ ). Dans les deux derniers cas il s'agit de graphes acycliques et donc d'arbres.

Pour décrire la dynamique des systèmes à événements discrets, les outils utilisés sont les automates finis, et lorsqu'il est nécessaire d'exprimer clairement le parallélisme les réseaux de Petri et le Grafset. Enfin pour tout ce qui concerne les problèmes d'évaluation de performance, les processus Markoviens et les réseaux de Petri stochastiques sont d'une grande utilité.

Nous pouvons remarquer qu'il existe en fait une grande unité entre toutes ces approches puisque les automates et les réseaux de Petri sont des graphes, et que, par exemple, le marquage d'un réseau de Petri peut s'exprimer par un multi-ensemble (ou monoïde commutatif) sur l'ensemble des places.

Bien entendu, les logiques modales et non classiques sont nécessaires, en particulier pour les preuves formelles lors du processus de vérification et de validation du logiciel mettant en œuvre les mécanismes assurant la sûreté.

### **3.2 Concepts**

Les concepts sont moins formalisés et universels que les outils mathématiques, mais ils sont tout autant nécessaires qu'eux pour mettre en œuvre une démarche permettant d'assurer une bonne maîtrise de la sûreté.

Un premier groupe de concepts est absolument nécessaire pour aborder la complexité des systèmes industriels en permettant une décomposition et une structuration des modèles. Il s'agit de concepts provenant de l'approche systémique, mais aussi de méthodes de Génie Logiciel comme le concept d'objets, d'héritage, de composition, d'agrégation, de hiérarchie.

Un autre groupe de concepts importants est lié à la notion de contrainte. Une contrainte est une frontière entre un comportement normal et un comportement interdit car dangereux ou non efficace. Toutefois, ces contraintes ne sont pas toujours des barrières qu'il est absolument interdit de franchir. Des contraintes actives lors du fonctionnement normal, peuvent être relâchées dans des modes dégradés. Et inversement, lorsque le système est dans un mode anormal, certains seuils peuvent devenir très critiques car les équipements permettant de réagir en cas de dépassement sont momentanément indisponibles. Une bonne maîtrise des modes dégradés implique donc une remise en question de l'ensemble des contraintes actives lors de chaque événement significatif. Par le concept de contraintes mal formalisables ou non formalisables, il est possible de prendre en compte partiellement les facteurs humains dans des approches exprimées sous la forme d'un problème de satisfaction

de contraintes (CSP). On caractérise un ensemble de décisions compatibles avec les contraintes formalisables et on laisse l'opérateur humain prendre la décision finale en tenant compte des contraintes non formalisables.

Un troisième groupe de concepts provient de l'automatique. Il s'agit par exemple de la notion de système en boucle ouverte (un système de supervision ne faisant qu'apporter une aide à la conduite à un opérateur humain) et de celle de boucle fermée (réactivité automatique). On trouve également des notions comme celles d'événement observable, d'événement commandable *etc.*

### 3.3. Méthodes

Une méthode consiste à associer des outils mathématiques et des concepts pour aboutir à un objectif précis. Par exemple, pour évaluer la disponibilité ou la sécurité d'un système, on peut effectuer une simulation de Monte-Carlo à partir d'un modèle comportemental basé sur les réseaux de Petri où les événements non contrôlables sont des défaillances et les événements contrôlables les reconfigurations (voir la section IV.1.4). Un diagramme de fiabilité formalise par un graphe des relations de composition (ce qui est nécessaire pour la réalisation d'une fonction) et offre une méthode calcul de la disponibilité de cette fonction. Des approches fondées sur ce type de méthodes sont présentées en IV.1.2. Des concepts provenant de l'automatique associés à des réseaux de Petri sont à la base des méthodes présentées en IV.2.3, et quand les réseaux de Petri sont associés à des concepts de hiérarchie ou d'objets, on trouve les travaux présentés en IV.1.1 ou IV.2.2.

[Berruet 98] P. Berruet, Contribution au recouvrement des systèmes flexibles de production manufacturière : analyse de la tolérance et reconfiguration, thèse de Doctorat de l'Université des Sciences et Techniques de Lille, décembre 1998.

[Chaillet 95] A. Chaillet, Approche multi-modèles pour la commande et la surveillance en temps réel des systèmes à événements discrets, thèse de Doctorat de l'Université Paul Sabatier de Toulouse, décembre 1995.

[Combacau 91] M. Combacau, Commande et surveillance des systèmes à événements discrets complexes : application aux ateliers flexibles, thèse de l'Université Paul Sabatier de Toulouse, décembre 1991.

[Niel 89] E. Niel, J.P. Simon, La sécurité des installations robotisées, Editions Hermès, Collection Technologie de pointe, N. 20, mars 1989.

[Niel 92] E. Niel, A. Jutard, Contribution à la formalisation de la sécurité opérationnelle, Revue d'Automatique et de Productique Appliquée (RAPA), Vol.5, N.2, 1992, p.57-64.

[Toguyeni 92] A.K.A. Toguyeni, Surveillance et diagnostic en ligne dans les ateliers flexibles de l'industrie manufacturière, thèse de Doctorat de l'Université des Sciences et Techniques de Lille, novembre 1992.