Utilisation des séquences de pannes pour la conception de systèmes de commande sûrs. Application au ferroutage

Joffrey Clarhaut, Saïd Hayat, Blaise Conrard, Vincent Cocquempot

Utilisation des séquences de pannes pour la conception de systèmes de commande sûrs. Application au ferroutage

Joffrey Clarhaut*, — Saïd Hayat* — Blaise Conrard** — Vincent Cocquempot****

** Institut national de recherche sur les transports et leur sécurité (INRETS)*
20, Rue Elisée Reclus BP317, F-59666 Villeneuve d'Ascq Cedex
joffrey.clarhaut@inrets.fr ; said.hayat@inrets.fr
*** Laboratoire d'automatique, génie informatique et signal (LAGIS) UMR –CNRS 8146 , Université de Lille 1 : Sciences et Technologies, F-59655 Villeneuve d'Ascq Cedex blaise.conrard@polytech-lille.fr ; vincent.cocquempot@univ-lille1.fr*

RÉSUMÉ. *Cet article propose une méthode originale pour la conception à moindre coût de systèmes d'automatisation tolérants aux fautes. La sûreté de fonctionnement est évaluée de manière semi-quantitative en tenant compte de l'ordre d'apparition des défaillances. Cette évaluation tient compte du nombre, de l'ordre et de la longueur des séquences menant aux événements redoutés. Cette méthode présente l'intérêt d'être suffisamment simple pour permettre d'étudier un grand nombre de solutions potentielles d'architectures matérielles et suffisamment précise pour permettre leur comparaison. Parmi les méthodes d'évaluation classiques, certaines permettent de traiter cet aspect dynamique (graphe de Markov, réseau de Petri), mais elles ne sont pas ou peu appropriées pour la comparaison de différentes solutions. La méthode de conception présentée y remedie par l'emploi d'alternatives de réalisation. Une phase d'optimisation vient sélectionner les meilleures alternatives en maximisant la sûreté de fonctionnement pour un coût donné. La méthode est illustrée par la conception d'un système de détection des incendies pour le ferroutage.*

ABSTRACT. *This article proposes an original method for the design of faulty tolerant automated systems at lower cost. Dependability is evaluated semi-quantitatively by taking into account the order of failures. This evaluation takes into account the number, the order and the length of failure sequences leading to a dreaded event. This method has the advantage to be enough simple to allow the study of a large number of potential architectures and to be sufficiently precise to enable comparison between them. Among the classical evaluation methods, some of them may consider this dynamic aspect (Markov graphs, Petri nets), but they are inappropriate for comparing different solutions. The design method allows the use of alternative achievement. An optimization phase just selects the best alternatives maximizing dependability at a given cost. The method is illustrated by designing a fire detection system for a railroad system.*

MOTS-CLÉS : *Sûreté de fonctionnement, conception de systèmes, séquences de défaillances, ferroutage.*

KEYWORDS: *Dependability, system design, sequences of failures, railroad systems.*

[e] soumission à *Journal Européen des Systèmes Automatisés*, le 23 juillet 2009.

1. Introduction

Un système d'automatisation est composé de composants physiques, capteurs, actionneurs et unités de traitement organisés afin d'accomplir un ensemble de missions (Simon et al., 2005). Un tel système est sûr si malgré la défaillance de certains de ses composants, il reste en état d'accomplir l'ensemble ou une partie de ces missions et surtout s'il reste dans des états ne risquant pas d'amener le système vers une situation jugée critique ou dangereuse. Lors du cycle de conception, l'objectif du concepteur est donc de déterminer un système qui garantit un niveau acceptable de sûreté de fonctionnement (Laprie., 1995). Ce niveau de sûreté de fonctionnement caractérise le risque de ne plus pouvoir rendre les missions attendues ou d'atteindre une situation redoutée. Il peut être évalué soit de manière quantitative et probabiliste sur la base des taux de défaillance de chacun des composants (Rausand et al., 2004), (Villemeur., 1992), soit de manière qualitative par l'étude des ensembles de combinaisons de défaillances amenant le système vers les états redoutés (Dutuit et al., 1997), (Rauzy., 2001), (Conrard et al., 2006). Cet article propose une méthode originale pour la conception à moindre coût de systèmes d'automatisation tolérants aux fautes. La sûreté est évaluée de manière semi-quantitative en tenant compte de l'ordre d'apparition des défaillances. Cette évaluation tient compte du nombre, de l'ordre et de la longueur des séquences menant aux événements redoutés. Cette méthode présente l'intérêt d'être suffisamment simple pour permettre d'étudier un grand nombre de solutions potentielles d'architectures matérielles et suffisamment précise pour permettre leur comparaison. Parmi les méthodes d'évaluation classiques, certaines permettent de traiter cet aspect dynamique (graphe de Markov, réseau de Petri), mais elles ne sont pas ou peu appropriées pour la comparaison de différentes solutions. La méthode de conception présentée y remedie par l'emploi d'alternatives de réalisation.
Cet article est divisé en trois parties. Dans la première partie, les concepts généraux relatifs aux méthodes de conception et d'évaluation de systèmes sûrs sont présentés puis la contribution est clairement mise en avant. Dans la deuxième partie, la méthodologie de conception de systèmes sûrs proposée est expliquée. Dans la troisième partie, la démarche est appliquée pour concevoir un système de détection des incendies pour le wagon de ferroutage. Une comparaison avec la méthode des arbres de défaillances classiques est réalisée afin de montrer les contributions de cette nouvelle approche. En conclusion, les notions importantes sont rappelées et quelques perspectives de ces travaux de recherche sont présentées.

2. Concepts et problèmes relatifs à la conception de systèmes sûrs de fonctionnement

2.1. *Méthodologie de conception de systèmes d'automatisation*

La conception d'un système d'automatisation impose la détermination successive de trois types d'architectures (Simonot-Lion et al., 1995), (Cauffriez et al., 2004) : l'architecture fonctionnelle, l'architecture matérielle et l'architecture opérationnelle.

L'architecture fonctionnelle est construite suivant des spécifications fonctionnelles et des contraintes. Cette architecture représente le lien entre différentes fonctions du système sous forme d'une décomposition hiérarchique en fonctions, sous-fonctions et fonctions élémentaires (Figure 1, Activité A1) (AFNOR, 2004) . Aux fonctions élémentaires de cette architecture fonctionnelle est ajoutée une association d'un ensemble de matériels reflétant l'ensemble des choix matériels possibles pour l'accomplissement de ces fonctions (Figure 1, Activité A2). Cette architecture fonctionnelle et matérielle peut prendre en compte des composants de base comme des composants matériels et logiciels ainsi que des systèmes de communication.

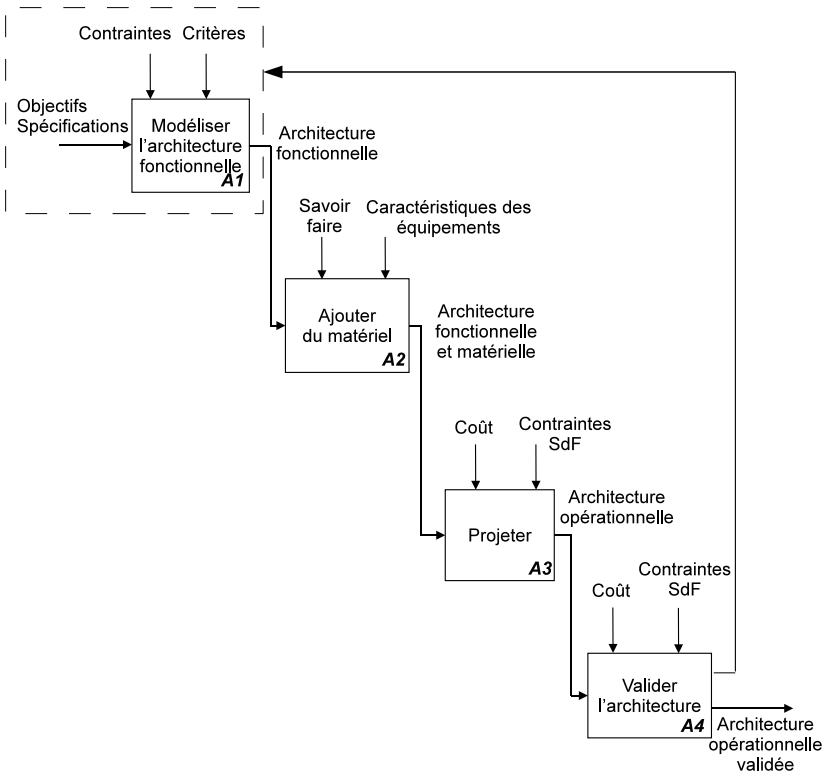

Figure 1. Méthodologie de conception

L'architecture opérationnelle est le résultat d'une projection, sous contraintes de coût et de sûreté de fonctionnement, de l'architecture fonctionnelle et matérielle (Figure 1, Activité A3). Cette architecture opérationnelle est ensuite validée suivant des objectifs de coût et de sûreté de fonctionnement fixés par le concepteur dans son cahier des charges (Figure 1, Activité A4). Si les objectifs ne sont pas atteints, la boucle de rétroaction entre les activités A1 et A4 fournit des informations qui seront utilisées pour améliorer les performances du système par la modification des objectifs/spécifications,

des contraintes ou des critères de base. L'évaluation du niveau de sûreté de fonctionnement peut être faite de façon statique ou dynamique. L'évaluation statique est une évaluation probabiliste des critères de la sûreté de fonctionnement réalisée par rapport à chaque mode de défaillance. L'évaluation dynamique, quant à elle, considère les séquences ordonnées de modes de défaillances ou scénarios. Cette dernière évaluation permet de déterminer les scénarios amenant le système vers une situation dangereuse précise.

2.2. *Notions de base de la sûreté de fonctionnement*

La sûreté de fonctionnement est définie comme la science des défaillances (Villemeur, 1988). Elle est caractérisée par l'analyse des défaillances du système et de leurs conséquences. Les quatre principales caractéristiques de la sûreté de fonctionnement sont : la fiabilité, la disponibilité, la maintenabilité et la sécurité (AFNOR, 1988), (AFNOR., 1991), (Zwingelstein., 1999).

Une approche unifiée pour considérer ces caractéristiques est d'utiliser le concept d'événement redouté (Conrard et al., 2007). Par exemple, les événements redoutés « impossibilité d'achever la mission », « arrêt inattendu » ou « comportement dangereux » sont relatifs respectivement aux paramètres de disponibilité, fiabilité et sécurité. L'événement redouté permet de considérer les caractéristiques de la sûreté de fonctionnement de façon plus qualitative et plus compréhensible.

De même, un système sûr est un système qui accomplit ce pourquoi il a été conçu, sans incident réduisant sa disponibilité et sans accident réduisant sa sécurité (Conrard et al., 2006). Comme les quatre critères de la sûreté de fonctionnement ne sont pas indépendants (Sourisse et al., 1997), la conception d'un système sûr nécessite de trouver le meilleur compromis disponibilité-fiabilité-sécurité-maintenabilité.

2.3. *Méthodes d'évaluation de la sûreté de fonctionnement*

L'évaluation des critères de sûreté de fonctionnement d'un système peut être effectuée par différentes méthodes. De nombreux travaux de recherches (Kumamoto et al., 1996),(Rausand et al., 2004) et la norme internationale dédiée à la sûreté de fonctionnement (IEC, 2003), décrivent ces méthodes.

Les arbres de défaillance et les diagrammes de fiabilité sont des méthodes statiques classiques qui permettent d'isoler les parties du système et les composants qui sont sensibles aux défaillances. Cependant, ces méthodes ne prennent pas ou difficilement en compte les séquences ordonnées de modes de défaillance et les dépendances temporelles entre fonctions (Kerhen et al., 2003). Comme cela sera vu dans les parties suivantes, ces aspects temporels ne peuvent pas être négligés pour une évaluation précise de la sûreté de fonctionnement d'une architecture matérielle donnée.

Les méthodes basées sur les réseaux de Petri ou les graphes de Markov incluent la prise en compte de ces aspects temporels. Ces méthodes permettent de considérer différents taux de défaillance des composants en fonction de l'état global du système. Cependant, l'étude de ces modèles est plus complexe et s'appuie généralement sur des méthodes de simulation qui requièrent des temps de calcul relativement longs, surtout dès que le nombre de composants et le nombre d'états à étudier deviennent suffisamment important. Ainsi, ces méthodes ne sont pas bien adaptées pour la conception et l'optimisation où un grand nombre de solutions potentielles doivent être évaluées et comparées (Moncelet, 1998), (Jampi et al., 2001),(Schoenig et al., 2006). L'idée de combiner les méthodes classiques et les méthodes de simulation afin d'évaluer de façon précise la sûreté de fonctionnement a fait l'objet de nombreux travaux (Dugan., 2001),(Cepin et al., 2002),(Bouissou et al., 2004), mais ces méthodes souffrent de la même limitation concernant les temps de traitement. Il s'agit donc de regrouper et d'adapter les outils d'évaluation rapides et les algorithmes d'optimisation au sein d'une méthodologie globale de conception de systèmes sûrs.

Ce besoin d'un outil global a motivé le développement d'un nouveau modèle statique/dynamique. Ce nouveau modèle, appelé « Arbre de défaillances multiples amélioré » est basé sur une représentation arborescente assez classique, proche des arbres de défaillance. Il prend en compte les séquences ordonnées de multiples modes de défaillances et leurs dépendances temporelles. Les dépendances temporelles entre les défaillances des composants ou des fonctions qu'ils supportent sont exprimées dans le modèle par des opérateurs temporels. Ces ajouts permettent d'évaluer plus précisément le niveau de sûreté de fonctionnement.

Par ailleurs, les événements considérés n'étant que des défaillances permanentes et soudaines, la méthode proposée ne s'interesse qu'à des systèmes non-réparables durant le temps d'exécution d'une mission. Enfin, dans l'arbre proposé, des noeuds particuliers offrant différentes alternatives de réalisation sont utilisés. Ils permettent de proposer plusieurs solutions de réalisation du système afin d'en déterminer la ou les meilleures.

3. Présentation de la méthodologie de conception

Les objectifs de la méthodologie proposée sont :

– de modéliser un système d'automatisation sous la forme d'un ensemble de d'alternatives de réalisation,

– de définir les conséquences et les effets des défaillances selon leur ordre d'apparition,

– d'évaluer avec suffisamment de précision le niveau de sûreté de fonctionnement d'un ensemble d'architectures matérielles,

– d'obtenir un ensemble d'architectures matérielles possibles, dont chaque solution est caractérisée par un niveau de sûreté de fonctionnement et un coût.

La méthodologie est décomposable en deux phases : la phase de modélisation et la phase d'optimisation. Ces deux phases sont définies dans les sections 3.2 et 3.3. Nous détaillons d'abord les définitions et propriétés utilisées.

3.1. *Définitions et propriétés de base*

Cette section définit les concepts et les notions utilisés par la méthodologie proposée. Les définitions d'un scénario et de ses valeurs caractéristiques sont enoncées. Les opérateurs et les propriétés (ou lois de composition) de l'arbre de défaillances multiples sont également présentés. Le coût, le niveau de sûreté de fonctionnement et l'ensemble des architectures matérielles sont également définis dans cette section.

3.1.1. *Défaillance, scénario et coefficient de fiabilité relatif*

Définition 1 : Une *défaillance* est un événement non désiré, d'occurrence aléatoire. Cet événement correspond à la transition d'un état normal vers un état non désiré d'un composant ou d'un ensemble de composants. Dans cet état non désiré, il est supposé que l'élément considéré ne peut plus accomplir correctement ou complètement sa mission.

Pour un même élément, plusieurs défaillances ou modes de défaillances peuvent être considérées s'il existe plusieurs états non désirés. Appliqué à des phases de fonctionnement où le système est supposé non réparable, un élément ne peut revenir à son état initial. Par la suite, on distinguera par événement redouté, la défaillance d'une fonction supportée par un ensemble de composant, tandis que le terme défaillance ne sera employé que pour un composant particulier.

Définition 2 : Un scénario correspond à une séquence de défaillances qui amène le système vers un événement redouté précis D. Un scénario est un ensemble ordonné dans le temps de modes de défaillances noté φ_D tel que :

$$\varphi_D = [F_i^1, \ldots, F_j^n]$$

où F_α^β est le mode de défaillance qui apparaît à la position β dans φ_D.

Notation 1 : (Ensemble de scénarios) Soit Φ_D l'ensemble des scénarios amenant le système vers un événement redouté D. Soit φ_D^iun élément de Φ_D.

$$\Phi_D = \left\{\varphi_D^i, \ldots, \varphi_D^m\right\}$$

où φ_D^i est le ième élément de Φ_D.

Notation 2 : (Longueur d'un scénario) Pour un scénario composé d'un ensemble de modes de défaillances notés F_i avec $i = 1, \ldots, n$ et $n = card(\varphi_D)$, la longueur $L(\varphi_D)$ est égale au cardinal de φ_D.

$$L(\varphi_D) = card(\varphi_D)$$

Notation 3 : (L_{min}Longueur minimale d'un ensemble de scénarios) La longueur minimale de tous les scénarios contenus dans Φ_D est noté .

$$L^D_{min} = min_{1 \leq i \leq card(\Phi_D)} L(\varphi^i_D)$$

Pour un événement redouté particulier, cette valeur correspond au nombre de modes de défaillances que le système peut tolérer avant l'occurrence de cet événement redouté.

Notation 4 : (Ensemble de scénarios ayant une longueur minimale) L'ensemble des scénarios de longueur minimale L_{min} amenant à l'événement redouté D est noté Δ_D.

$$\Delta_D = \{\varphi^i_D \in \Phi_D / L(\varphi^i_D) = L^D_{min}\}$$

Remarque. — Dans la littérature (Bouissou, 2006), un scénario qui amène le système vers un événement redouté est considéré comme minimal s'il n'est pas inclus dans un autre scénario qui amène vers le même événement redouté. L'ensemble des scénarios ayant une longueur minimale, défini dans la notation 4, est un sous ensemble de l'ensemble des scénarios minimaux où seules les séquences ayant une longueur minimale sont considérées.

Notation 5 : (Nombre de combinaisons d'un ensemble de scénarios) Le nombre de scénarios contenus dans l'ensemble Δ_D est noté N^D_{min}. Cette valeur, associée à L^D_{min}, est directement liée à la probabilité d'occurrence de l'événement redouté D.

$$N^D_{min} = card(\Delta_D)$$

L^D_{min} et N^D_{min} représentent les valeurs caractéristiques d'un événement redouté D.

3.1.2. *Opérateurs caractérisant les relations entre modes de défaillances*

Les arbres de défaillance classiques permettent de représenter graphiquement les combinaisons d'événements qui conduisent à la réalisation d'un événement redouté. Des opérateurs logiques, classiquement ET et OU, sont utilisés pour décrire les relations entre ces modes de défaillances. La prise en compte de dépendance temporelle entre événements et défaillances est réalisée par l'ajout de deux opérateurs notés p-ET et SEQ (Dugan et al., 1992, Coppit et al. 2000), classiquement utilisés sour leur forme anglaise (PAND et SEQ). Dans la phase de modélisation proposée, ces opérateurs sont utilisés dans les relations entre les différents modes de défaillances des fonctions, sous-fonctions et composants. Ces opérateurs possèdent également des propriétés mathématiques (ou lois de composition) utiles dans la phase d'optimisation lors du traitement de l'arborescence.

Considérons A, B et C, trois événements redoutés, tels que C est le résultat de l'association de A et de B avec l'un des opérateurs. Δ_A, Δ_B et Δ_C sont les ensembles de scénarios minimaux associés à A, B et C.

L'opérateur ET représente le cas où l'occurrence de C se produit après l'occurrence de A et de B quel que soit leur ordre d'apparition.

Propriété 1 : Avec $C = A\,ET\,B$, les valeurs caractéristiques de C peuvent être évaluées à l'aide des relations suivantes dont le détail des calculs est donné en Annexe :

$$L^C_{min} = L^A_{min} + L^B_{min}$$

$$N^C_{min} = \frac{(L^A_{min} + L^B_{min})!}{L^A_{min}! \times L^B_{min}!} \times N^A_{min} \times N^B_{min}$$

Cette dernière relation détermine le nombre des combinaisons différentes associant les événements des séquences menant à A et ceux menant à B formant l'ensemble des séquences menant à A et B.

L'opérateur OU représente le cas où l'occurrence de C se produit après l'occurrence de A ou de B.

Propriété 2 : Avec $C = A\,OU\,B$, les valeurs caractéristiques de C peuvent être évaluées à l'aide des relations suivantes :

$$si\ L^A_{min} < L^B_{min} \left\{ \begin{array}{c} L^C_{min} = L^A_{min} \\ N^C_{min} = N^A_{min} \end{array} \right.$$

$$si\ L^A_{min} = L^B_{min} \left\{ \begin{array}{c} L^C_{min} = L^A_{min} \\ N^C_{min} = N^A_{min} + N^B_{min} \end{array} \right.$$

$$si\ L^A_{min} > L^B_{min} \left\{ \begin{array}{c} L^C_{min} = L^B_{min} \\ N^C_{min} = N^B_{min} \end{array} \right.$$

Définition 3 : (Opérateur p-ET) L'opérateur *p-ET* est une extension de l'opérateur ET intégrant une priorité entre les événements. Il représente le cas où l'occurrence de C se produit après les occurrences successives de A puis de B. Plus spécifiquement, si $\varphi_A = [F^1_A, \ldots, F^n_A]$ et $\varphi_B = [F^1_B, \ldots, F^m_B]$ sont deux séquences entraînant l'occurence respective des événements A et B, les séquences φ_C entraînant l'occurence de C et déduites de φ_A et φ_B respectent la contrainte suivante :

$$\left\{ \begin{array}{c} \varphi_C = [F^1_C, \ldots, F^{m-n-1}_C, F^m_B] \\ \text{avec} \\ F^i_C \in \varphi_A \cup \varphi_B \end{array} \right.$$

$$- \; Si \; F_C^i = F_A^j \; et \; \forall \; k > i \; alors \; si \; F_C^k = F_A^l, \; on \; a \; l > j$$

$$- \; Si \; F_C^i = F_B^j \; et \; \forall \; k > i \; alors \; si \; F_C^k = F_B^l, \; on \; a \; l > j$$

Cet opérateur est utile lorsque les conséquences de la défaillance de fonctions ou composants diffèrent selon leur ordre d'apparition.

Propriété 3 : Avec $C = A$ p-ET B, les valeurs caractéristiques de C peuvent être évaluées à l'aide des relations suivantes :

$$L_{min}^C = L_{min}^A + L_{min}^B$$

$$N_{min}^C = \frac{(L_{min}^B + L_{min}^B - 1)!}{L_{min}^B! \times \left(L_{min}^A - 1\right)!} \times N_{min}^A \times N_{min}^B$$

Définition 4 : (Opérateur SEQ) L'opérateur SEQ est une restriction de l'opérateur *p-ET*. L'expression C = A *SEQ* B représente le cas où l'occurrence de C se produit après les occurrences successives de A puis de B et impose qu'aucune défaillance amenant le système vers l'événement redouté B ne se produise avant l'occurrence de l'ensemble des défaillances amenant à l'événement redouté A. Plus formellement, soit $\varphi_A = [F_A^1, \ldots, F_A^n]$ (resp. $\varphi_B = [F_B^1, \ldots, F_B^m]$ et φ_B) une séquence de défaillances menant à l'événement redouté A (resp. B), la concaténation des 2 séquences φ_A et φ_B : $\varphi_{A \; SEQ \; B} = [F_A^1, \ldots, F_A^n, F_B^1, \ldots, F_B^m]$ entraîne l'événement redouté C, on note C = A SEQ B.

Cet opérateur est utile pour modéliser des redondances passives de fonctions ou de composants. En effet, la fonction redondée est exécutée seulement quand la fonction principale est défaillante. De ce fait, l'occurrence de la défaillance de la fonction redondée ne peut se produire qu'après l'occurrence de la défaillance de la fonction principale.

Propriété 4 : Avec $C = A \, SEQ \, B$, les valeurs caractéristiques de C peuvent être évaluées à l'aide des relations suivantes :

$$L_{min}^C = L_{min}^A + L_{min}^B$$

$$N_{min}^C = N_{min}^A \times N_{min}^B$$

3.1.3. *Comparaisons entre niveau de sûreté de fonctionnement et entre systèmes*

Définition 5 : (*Systèmes équivalents*) Deux systèmes (ou composants) sont équivalents s'ils peuvent accomplir les mêmes fonctions et si les mêmes événements redoutés peuvent être définis pour ces deux systèmes.

Notation 6 : (*Niveau de sûreté de fonctionnement pour un événement redouté*) Pour un système S et pour un événement redouté D, le niveau de sûreté de fonctionnement est formé par le couple $\left(L_{min}^{D,S}, N_{min}^{D,S}\right)$. Il est noté Niv_D^S.

Pour un système particulier, ce couple caractérise la probabilité d'occurrence de l'événement redouté D. Ainsi, ce couple peut être utilisé pour comparer plusieurs systèmes entre eux. En annexe une démonstration de la relation entre ce couple et la probabilité d'occurrence de l'événement considéré est présentée et précise le domaine de validité de cette relation.

Définition 6 : (*Comparaison entre niveaux de sûreté de fonctionnement pour un même événement redouté*) Pour deux systèmes équivalents S_1 et S_2 et pour le même événement redouté D, le niveau de sûreté de fonctionnement de S_1 est dit meilleur que celui de S_2, si la relation suivante est vérifiée :

$$\begin{array}{c} L_{min}^{D,S_1} > L_{min}^{D,S_2} \\ ou \left\{ \begin{array}{l} L_{min}^{D,S_1} = L_{min}^{D,S_2} \\ N_{min}^{D,S_1} < N_{min}^{D,S_2} \end{array} \right. \end{array}$$

On notera $Niv_D^{S_1} > Niv_D^{S_2}$.

Définition 7 : Par extension de la définition 6, le niveau de sûreté de fonctionnement entre deux systèmes est dit identique, c'est à dire $Niv_D^{S_1} = Niv_D^{S_2}$, si $L_{min}^{D,S_1} = L_{min}^{D,S_2}$et si $N_{min}^{D,S_1} = N_{min}^{D,S_2}$

Notation 7 : (*Niveau de sûreté de fonctionnement d'un système*) Pour un système (ou un composant) composé de n événements redoutés D_i ($i \in [1, n]$), le niveau de sûreté de fonctionnement est donné par l'ensemble Niv^Sde tous les $Niv_{D_i}^S$.

$$Niv^S = \left\{ Niv_{D_1}^S, \ldots, Niv_{D_n}^S \right\}$$

Cet ensemble exprime la robustesse du système considéré à plusieurs modes de défaillances et pour différents événements redoutés.

Propriété 5 : (*Comparaison entre systèmes équivalents de niveau de sûreté de fonctionnement*) Soit deux systèmes équivalents S_1 et S_2 et un ensemble d'événements redoutés, le niveau de sûreté de fonctionnement Niv^{S_1} est supérieur à Niv^{S_2} si la relation suivante est vérifiée :

$$\forall i \; Niv_{D_i}^{S_1} \geq Niv_{D_i}^{S_1} \; et \; \exists j \; Niv_{D_j}^{S_1} > Niv_{D_j}^{S_1}$$

On notera $Niv^{S_1} > Niv^{S_2}$.

$Niv^{S_1} > Niv^{S_2}$signifie que le système S1 est plus tolérant aux fautes que le système S2.

Notation 8 : (*Coût d'un système*) A chaque composant est associée une valeur représentant son coût. Pour un système S, son coût est égal à la somme des coûts individuels de ses q composants.

$$Coût(S) = \sum_{i=1}^{q} Coût(i)$$

Le coût calculé pour l'architecture matérielle correspond au coût d'achat de chacun de ses composants, cependant, d'autres coûts peuvent être pris en compte : coût de maintenance, coût de mise en service, ...

Notation 9 : (*Caractérisation d'un système*) Un système S est caractérisé par le couple C_S formé du coût et du niveau de sûreté de fonctionnement.

$$C_S = \left\{Coût(S), Niv^S\right\}$$

Définition 8 : (*Comparaison de systèmes*) Le système S_1 est meilleur que le système S_2, c'est à dire $C_{S_1} > C_{S_2}$, si la relation suivante est vérifiée.

$$\begin{cases} Coût(S_1) = Coût(S_2) \\ Niv^{S_1} < Niv^{S_2} \end{cases}$$
$$ou \begin{cases} Coût(S_1) < Coût(S_2) \\ Niv^{S_1} \geq Niv^{S_2} \end{cases}$$

Définition 9 : (*Systèmes optimaux*) Pour un ensemble de systèmes équivalents Ω, l'ensemble des systèmes optimaux $\Omega_{optimal}$ est défini par la relation suivante :

$$\Omega_{optimal} = \{S \in \Omega, \nexists S_i \in \Omega \, avec \, C_{S_i} > C_S\}$$

3.2. *Cas particulier des composants de sécurité*

L'utilisation des scénarios et de leur comparaison sur la base de leur longueur implique que les modes de défaillances considérés sont équiprobables ou tout au moins ont une probabilité d'occurrence du même ordre de grandeur. En conséquence, la comparaison de composants ayant des fiabilités très dissemblables ne peut pas être effectuée avec la méthode proposée. C'est plus particulièrement le cas des composants de sécurité qui, par rapport à des composants standards, offrent une grande fiabilité vis-à-vis de modes de défaillance dangereux, c'est-à-dire ceux susceptibles d'amener le système dans une situation dangereuse ou critique.

La méthode peut être enrichie en remplaçant les longueurs (L^D_{min}) par un coefficient permettant de comparer des composants de fiabilités variées. Noté CFR (Coefficient de Fiabilité Relatif), ce coefficient rend compte de la probabilité qu'un événement ou qu'une séquence d'événements apparaisse. Les relations précédentes dans

lesquelles L^D_{min} est remplacé par CFR^D_{min} restent valides sous l'hypothèse d'indépendance des modes de défaillances et des événements redoutés. C'est à dire que les modes de défaillances ne peuvent pas apparaître simultanément dans deux scénarios et que les scénarios conduisant à des événements redoutés ne partagent pas de défaillances.

Définition 10 : (Coefficient de fiabilité relatif d'un mode de défaillance) Le coefficient de fiabilité relatif, noté CFR_{F_i}(Conrard et al., 2006), caractérise la probabilité d'occurrence d'une défaillance F_i par rapport à une probabilité de défaillance de référence. Appliquée à un mode de défaillance d'un composant, cette valeur correspond au nombre équivalent de défaillances de composants standards qu'il faut avoir pour obtenir la même probabilité d'occurrence. Le CFR d'un composant est lié à la probabilité d'occurrence de la défaillance F_i par la relation suivante :

$$Prob(F_i) = r^{CFR_{F_i}}\, avec\, 0 < r << 1$$

où r est la défiabilité (1 - R($t_{mission}$)) de référence, correspondant à celle d'un composant standard (CFR = 1).

Remarque. — Lorsque les composants ont des fiabilités homogènes, les coefficients correspondants sont identiques et sont fixés à 1. Dans ce cas particulier, le CFR^D_{min} est égal à la longueur L^D_{min}.

Le CFR d'un scénario est égal à la somme des CFR des événements qui le composent sous condition d'indépendance des événements.

$$CFR_{\varphi_D} = \sum_{F_i \in \varphi_D} CFR_{F_i}\, avec\, \varphi_D = \left\{F_i^1, \ldots, F_i^n\right\}$$

D'un point de vue pratique, si le concepteur veut distinguer certains composants ayant une fiabilité bien supérieure aux autres, il leur affecte un CFR supérieur à 1, tandis que les composants standards ont un CFR fixé à 1, valeur servant de référence. Ainsi, un CFR d'une valeur de 2 appliquée à un mode de défaillance particulier d'un composant indique que sa probabilité de défaillance est équivalente à la défaillance de deux composants standards.

Pour illustrer ceci, considérons que la probabilité de défaillance $Prob(F)$ d'un composant standard (pour lequel $CFR_F = 1$) durant une mission est de l'ordre de 10^{-3}. Le CFR_{F_R}d'un composant robuste pour une défaillance F_R avec la fiabilité à $Prob(F_R) = 10^{-5}$ est évalué à $1,66$ par la relation suivante.

$$CFR_{F_R} = \frac{\ln(Prob(F_R))}{\ln(Prob(F))}$$

En annexe, une démonstration est fournie et montre, pour deux scénarios φ_1 et φ_2, sous quelles conditions l'inégalité $CFR_{\varphi_1} < CFR_{\varphi_2}$ implique que la probabilité d'occurrence de φ_1 est supérieure à celle de φ_2.

Dans les deux sous-sections suivantes, la méthodologie de conception de systèmes d'automatisation est expliquée. Cette méthodologie utilise les séquences ordonnées de modes de défaillances afin d'évaluer le coût et le niveau de sûreté de fonctionnement d'architectures matérielles. Après avoir identifié les événements redoutés d'un système d'automatisation, l'ensemble des architectures matérielles possibles est déterminé ce qui permet de sélectionner l'ensemble des architectures opérationnelles optimales.

3.3. *Phase de modélisation*

La première phase de la méthodologie de conception a pour objectif de représenter dans un modèle hiérarchique toutes les architectures matérielles potentielles réalisables. Ce modèle est construit en deux étapes : dans la première étape, le système est décrit à l'aide d'une décomposition fonctionnelle et dans la seconde étape, les modes de défaillance et leurs relations entre niveaux hiérarchiques sont ajoutées à chaque noeud de cette décomposition.

3.3.1. *Modèle hiérarchique*

Le modèle du système est décrit sous forme d'arbre obtenu à partir d'une analyse fonctionnelle hiérarchique. Cette analyse est souvent utilisée pour modéliser des systèmes d'automatisation (Benard et al., 2008),(Conrard et al., 2006). Dans la méthodologie proposée, trois types de noeuds sont utilisés : le noeud associatif, le noeud alternatif et le noeud élémentaire.

– Le noeud associatif modélise une fonction complexe nécessitant, pour sa réalisation, un ensemble de sous-fonctions. Par exemple, sur la Figure 2a, une fonction de contrôle par rebouclage nécessite une fonction de mesure, une fonction de contrôle et une fonction de commande.

– Le noeud alternatif est utilisé pour modéliser différentes possibilités d'implantation d'une fonction. Par exemple, sur la Figure 2b, pour une fonction de mesure le concepteur peut choisir d'utiliser soit un unique capteur, soit une fonction d'estimation ou soit un ensemble de capteurs redondés.

– Le noeud élémentaire (Figure 2c) est utilisé pour modéliser une fonction de base, associée avec un unique composant.

3.3.2. *Arbre de défaillances multiples amélioré*

La décomposition hiérarchique précédente donne l'architecture de l'arbre de défaillances multiples. Contrairement aux arbres de défaillances classiques qui n'associent qu'à une fonction un mode de défaillance, l'arbre de défaillances multiples

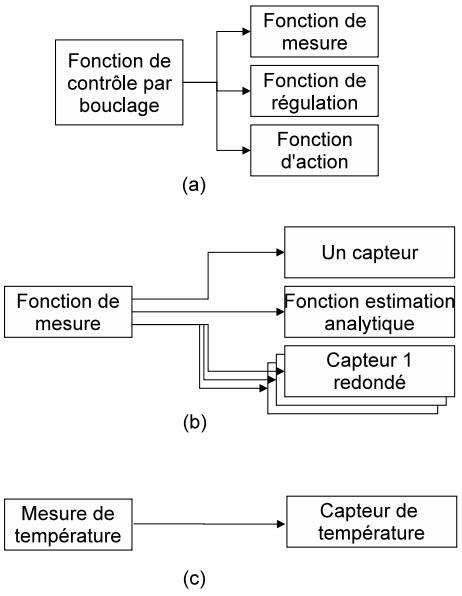

Figure 2. Exemples d'un noeud associatif (a), d'un noeud alternatif (b) et d'un noeud élémentaire (c)

contient pour chaque fonction de sa structure un à plusieurs modes de défaillances. Dans l'objectif de déterminer le comportement du système lors de l'occurrence d'une défaillance, l'arbre de défaillances doit être complété par une description des modes de défaillances possibles et de leurs effets sur le système. Cette phase associe à chaque noeud un ensemble de modes de défaillances décrivant comment peut être affecté l'accomplissement de la fonction correspondante. Pour les noeuds associatifs, les relations liant les modes de défaillances de la fonction associée à ceux de ses sous-fonctions doivent être ajoutées et utilisent les opérateurs ET, OU, p-ET et SEQ, décrits dans la section précédente. Par exemple Figure 3, la relation de défaillances peut être la suivante : (défaillance 1A = défaillance 2A p-ET défaillance 3A) ET (défaillance 1B = défaillance 2B ET défaillance 3B).

Pour les noeuds élémentaires, le CFR associé à chaque mode de défaillance définit la robustesse du composant associé vis-à-vis de ce mode. Indirectement, il spécifie s'il s'agit d'un composant standard, durci (ou éprouvé) ou de sécurité.

Pour les noeuds alternatifs, l'ensemble des modes de défaillances correspond à ceux des alternatives proposées et est l'union de ceux-ci. Plus particulièrement, suivant la technologie utilisée, certains modes de défaillances n'apparaissent pas pour certaines alternatives. Par exemple, un capteur intelligent peut posséder trois modes de défaillances : mesure trop basse, mesure trop haute et absence de mesure tandis qu'un capteur standard possède seulement deux modes de défaillances : mesure trop

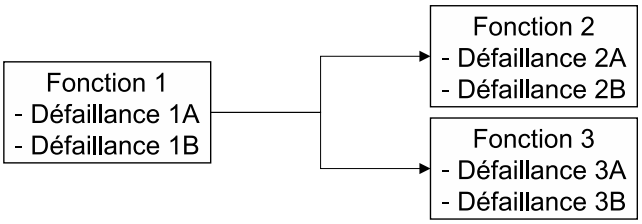

Figure 3. Exemples de modes de défaillances associés aux fonctions

basse et mesure trop haute car le capteur standard donne toujours une mesure. Dans ce cas spécifique, le CFR associé à un mode de défaillance n'apparaissant pas dans l'alternative sélectionnée, prend une valeur infinie (pratiquement très grande), indiquant que l'occurrence de ce mode ne peut apparaître.

L'arbre de défaillances multiples obtenu décrit différentes alternatives de réalisation du système, tout en modélisant le comportement dysfonctionnel, c'est-à-dire l'effet de combinaison de défaillances des composants sur la capacité du système à accomplir sa mission ou sur le risque de voir apparaître des situations critiques ou dangereuses.

3.4. *Phase d'optimisation*

L'objectif de la phase d'optimisation est de déterminer la meilleure architecture du système de contrôle commande parmi toutes les architectures potentielles décrites dans l'arbre de défaillances multiples. Une approche de type « bottom-up » est proposée et est comparable à la méthode du branch and bound (D'Ariano et al., 2006). Le principe général, détaillé en annexe, est de déterminer l'ensemble des solutions Pareto optimales pour chaque noeud du modèle hiérarchique jusqu'à obtenir celles du système.

Plus spécifiquement, pour les noeuds élémentaires, l'ensemble des solutions est composé d'une seule solution caractérisée par le coût de ses composants et par le CFR associé à chaque mode de défaillance. Pour les niveaux hiérarchiques supérieurs composés de noeuds associatifs et de noeuds alternatifs, cet ensemble de solutions est construit suivant les ensembles de niveaux hiérarchiques inférieurs. Dans le cas d'un noeud alternatif, cet ensemble est déterminé par l'union des différentes solutions possibles. Grâce à l'union de ces différents ensembles et grâce aux opérateurs de comparaison (cf. Définition 8), l'ensemble optimal est facilement construit. Dans le cas d'un noeud associatif, les solutions optimales sont déduites des différentes combinaisons des solutions offertes par chaque sous-fonction requise. Pour une combinaison particulière, le coût d'une solution obtenue est donnée par la somme des coûts des solutions de chaque sous-fonction requise. Le niveau de sûreté de fonctionnement est

évalué grâce aux valeurs caractéristiques de sûreté de fonctionnement (CFR et N) calculées pour chaque niveau de l'arbre à l'aide des relations entre défaillances et de leurs opérateurs associés. L'ensemble optimal est obtenu par l'union de toutes les solutions trouvées après l'application des critères de comparaison définis dans la section précédente.

D'un point de vue algorithmique, l'ensemble des solutions Pareto optimales est construit progressivement. Ainsi quel que soit le noeud considéré (associatif ou alternatif), chaque nouvelle solution potentielle est immédiatement comparée à celles précédemment trouvées. Plus précisément, une nouvelle solution potentielle est ajoutée à l'ensemble optimal seulement s'il n'existe pas de meilleure solution dans cet ensemble. Sur le même principe, quand une nouvelle solution est ajoutée, si d'autres solutions sont moins bonnes que la nouvelle, elles sont retirées de l'ensemble optimal. A la fin de la phase d'optimisation, la méthodologie fournit l'ensemble optimal pour le noeud traité. Pour l'optimisation de très grands systèmes, l'explosion combinatoire du nombre de solutions à envisager peut limiter l'emploi de cette méthode d'optimisation par des temps de calcul relativement longs. D'autres méthodes, comme les algorithmes génétiques, pourraient être utilisées (Sallak et al., 2006) pour déterminer des solutions de réalisations industriellement intéressantes, mais elles ne garantiront pas d'avoir nécessairement trouvé l'optimum.

4. Cas d'étude : conception d'un système d'automatisation pour un wagon de ferroutage

La méthodologie de conception est appliquée à un système de ferroutage. Dans une première section, les problèmes associés et les besoins liés à la conception de systèmes d'automatisation pour le wagon de ferroutage sont présentés. Dans une seconde section, la méthodologie est appliquée afin de concevoir un système de détection des incendies pour ce wagon. Enfin, un exemple d'architecture opérationnelle optimale est donné.

4.1. *Présentation du ferroutage et du cas d'étude*

Le ferroutage est un terme général du fret ferroviaire, correspondant à l'ensemble des systèmes qui transportent des camions. De nombreuses solutions sont proposées : Modalhor et Eurotunnel Fret (France), RoadRailer (USA), Expressway (Canada), Route Roulante (Suisse), Projet Sail (Allemagne), ... Ce type de transport présente de nombreux avantages comme la réduction de la congestion routière, la baisse de la pollution, la baisse de la consommation en carburants. Cependant, ce mode de transport présente de nombreux risques d'accidents pour lui-même (déraillement, incendie) et pour son environnement proche à cause des camions et du chargement transporté. Par conséquent, par rapport aux trains classiques, le ferroutage doit prendre en compte des exigences et des besoins de sûreté de fonctionnement supplémentaires : protection incendie, protection contre des agressions extérieures humaines et environ-

nementales, surveillance de la charge, ... La conception d'un wagon intelligent permet de répondre à ces exigences (Clarhaut et al., 2006), (Clarhaut et al., 2007). Ce wagon doit avoir des fonctionnalités supplémentaires qui augmentent le niveau de sûreté de fonctionnement global du système de ferroutage. La méthodologie proposée est illustrée pour la conception d'une nouvelle fonctionnalité de ce wagon intelligent : le système de détection d'incendie automatique. La conception de ce système a fait l'objet de nombreux travaux de recherche (Eisinger et al., 2001),(Amer et al., 2005),(Elsayed et al., 2005),(Jiang et al., 2006). Notre objectif est d'obtenir un système d'automatisation pour ce système de détection incendie qui présente un bon compromis entre son coût et son niveau de sûreté de fonctionnement.

4.2. *Phase de modélisation*

4.2.1. *Modèle hiérarchique*

La Figure 4 présente le modèle hiérarchique du système de détection incendie automatique. Ce système a trois missions en phase de convoi : détecter les incendies, alerter les opérateurs du système (conducteur du train) et éteindre les incendies en utilisant un système d'extinction du feu. La partie détection du système est composée de détecteurs de fumées et de détecteurs de chaleur. La partie signalisation est constituée d'un système de traitement utilisant des automates programmables (API) et un système d'alimentation. Le système d'extinction est activé par un système de relais. Les composants de base sont décris dans la Table 1. Deux types de composants (standards et sécuritaires) ayant des coûts différents peuvent être utilisés. Ces différents types de composants possèdent des niveaux de robustesse différents (cf. Définition 10). De plus, quatre types de redondances peuvent être choisis : active, passive, série ou parallèle. Les redondances séries ou parallèles font référence à l'organisation des composants. Dans une redondance active, les composants remplissent leur mission au même moment tandis que pour une redondance passive, le second composant est utilisé dès que le premier composant est défaillant.

4.2.2. *Arbre de défaillances multiples amélioré*

Afin de limiter le nombre de pages, seules les missions détection et signalisation sont considérées dans cet article. La première étape dans la construction de l'arbre de défaillances amélioré est de définir les événements redoutés. Deux événements redoutés sont considérés :

– Pas d'alarme incendie provenant du système de traitement quand un feu est détecté. Cet événement redouté est associé au niveau de sécurité du système et est noté ER_1.

– Fausse alarme, c'est-à-dire que le système d'automatisation active de façon intempestive une alarme en l'absence d'incendie. Cet événement redouté est associé au niveau de disponibilité du système et est noté ER_2.

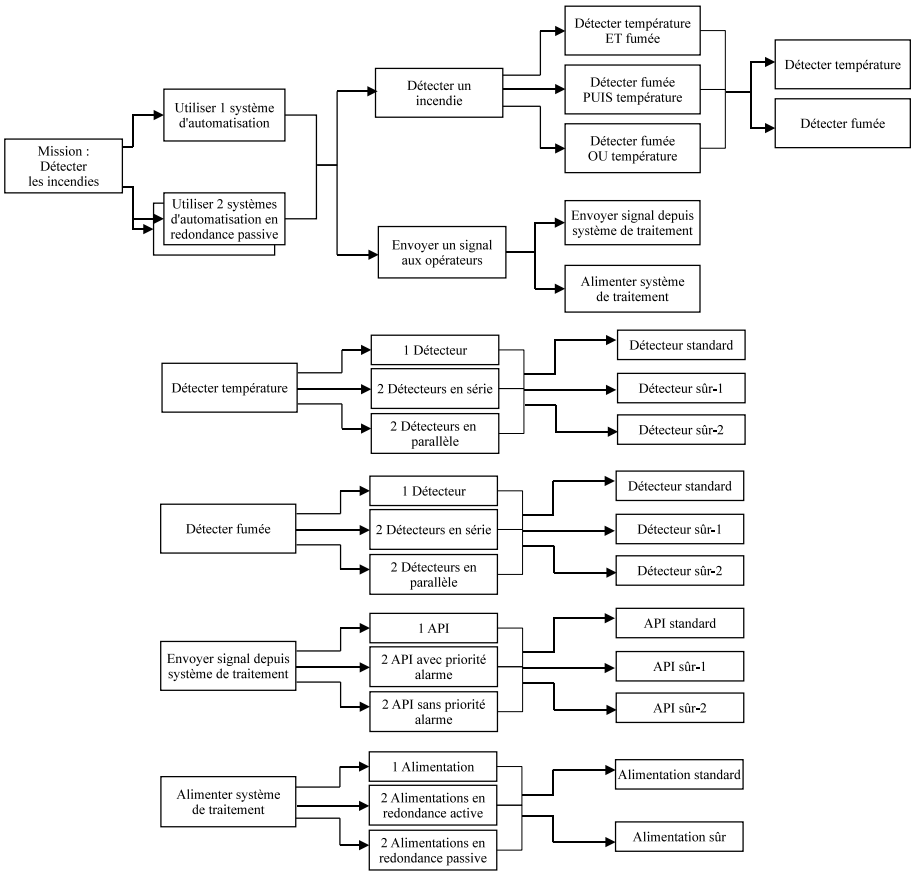

Figure 4. Modèle hiérarchique du système de détection incendie automatique

La seconde étape dans la construction de l'arbre est d'associer à chaque noeud de l'arbre hiérarchique l'ensemble des relations de modes de défaillances décrivant le comportement dysfonctionnel de la fonction correspondante. Par exemple, sur la Figure 4, la fonction de détection des incendies peut être accomplie de trois façons différentes (Arcs 1, 2 et 3 du noeud alternatif N3) : soit la fumée et la température sont détectées simultanément, soit successivement ou soit uniquement la fumée ou soit uniquement la température. Il est bon de noter qu'avec l'arbre de défaillance classique, la seconde possibilité, la détection successive, ne peut être modélisée. Dans le but d'expliquer comment les relations sont obtenues, considérons la fonction *détecter les incendies* qui considère l'alternative (Arc 2 du noeud alternatif N3) : détecter les fumées PUIS détecter l'augmentation de température. Les relations entre défaillances représentées pour ce noeud sont expliquées ci-dessous :

Composants	Mode de défaillance	Types de composants (Coût, {CFR})
Bloc d'alimentation	- Arrêt inattendu	- Standard (5, {1}) - Sûr (10, {2})
API	- Arrêt inattendu avec alarme - Arrêt inattendu sans alarme	- Standard (3, {1, 1}) - Sûr type 1 (8, {2, 1}) - Sûr type 2 (8, {1, 2})
Détecteur de température	- Continuellement actif - Continuellement passif	- Standard (1, {1, 1}) - Sûr type 1 (2, {2, 1}) - Sûr type 2 (2, {1, 2})
Détecteur de fumée	- Continuellement actif - Continuellement passif	- Standard (1, {1, 1}) - Sûr type 1 (2, {2, 1}) - Sûr type 2 (2, {1, 2})

Tableau 1. *Composants, modes de défaillances et types pour le système de protection incendie*

– Pas d'alarme incendie provenant du système de traitement quand un feu est détecté (mode de défaillance B1), si :

- La fonction *détecter les fumées* est continuellement passive (mode de défaillance E1), c'est à dire que la fonction ne produira pas d'alarme.

- ET

- La fonction *détecter la température* est continuellement passive (mode de défaillance D1), c'est à dire que la fonction ne produira pas d'alarme.

– Fausse alarme provenant du système (mode de défaillance B2), si :

- La fonction *détecter les fumées* est continuellement active (mode de défaillance E2), c'est à dire que la fonction produit une fausse alarme.

- p-ET

- La fonction *détecter la température* est continuellement active (mode de défaillance D2), c'est-à-dire que la fonction produit une fausse alarme.

Ces deux relations sont formalisées par les équations suivantes :
Détec.Fumées(E1) ET Détec.Temp(D1) $\Longrightarrow$ **B1**
Détec.Fumées(E2) p-ET Détec.Temp(D2) $\Longrightarrow$ **B2**

Pour chaque noeud de l'arbre, les relations de modes de défaillances sont associées avec le même principe pour chaque fonction jusqu'à atteindre les composants de base. L'arbre de défaillances multiples pour les deux événements redoutés est détaillé dans les Tables 2 et 3 et en figure 5. Les relations entre modes de défaillances sont listées en annexe. Cet arbre caractérise le comportement du système de détection des incendies en présence de défaillances grâce aux relations indirectes entre les modes de défaillances des composants à la base de l'arbre et les modes de défaillances des missions au sommet de l'arbre.

Dénomination de la fonction	Noeud correspondant	Modes de défaillances
Mission : Détecter et signaler incendies	Noeud alternatif N1	- ER1 : Non détection - ER2 : Fausse alarme
Utiliser 1 système de automatisation	Noeud associatif N2	- A1 : Non détection - A2 : Fausse alarme
Utiliser 2 systèmes de automatisation	Noeud associatif N2	- A1 : Non détection - A2 : Fausse alarme
Détecter un incendie	Noeud alternatif N3	- B1 : Non détection - B2 : Fausse alarme
Détecter Température ET Fumée	Noeud élémentaire N4	- D1, E1 : Continuellement passif - D2, E2 : Continuellement actif
Détecter Fumée PUIS température	Noeud élémentaire N4	- D1, E1 : Continuellement passif - D2, E2 : Continuellement actif
Détecter Fumée OU température	Noeud élémentaire N4	- D1, E1 : Continuellement passif - D2, E2 : Continuellement actif
Envoyer signal aux opérateurs	Noeud associatif N5	- C1 : Non détection - C2 : Fausse alarme
Détecter température	Noeud alternatif N6	- D1 : Continuellement passif - D2 : Continuellement actif
Détecter fumée	Noeud alternatif N7	- E1 : Continuellement passif - E2 : Continuellement actif
Envoyer signal depuis système de traitement	Noeud alternatif N8	- F1 : Arrêt inattendu ac alarme - F2 : Arrêt inattendu ss alarme
Alimenter système de traitement	Noeud alternatif N9	- G1 : Arrêt inattendu
Utiliser 1 détecteur température	Noeud alternatif N10	- H1 : Continuellement actif - H2 : Continuellement passif
Utiliser 2 détecteurs en série	Noeud alternatif N10	- H1 : Continuellement actif - H2 : Continuellement passif
Utiliser 2 détecteurs en parrallèle	Noeud alternatif N10	- H1 : Continuellement actif - H2 : Continuellement passif
Utiliser 1 détecteur fumée	Noeud alternatif N11	- I1 : Continuellement actif - I2 : Continuellement passif

Tableau 2. *Correspondance entre fonctions et modes de défaillances de la figure 5*

Dénomination de la fonction	Noeud correspondant	Modes de défaillances
Utiliser 2 détecteurs en série	Noeud alternatif N11	- I1 : Continuellement actif - I2 : Continuellement passif
Utiliser 2 détecteurs en parallèle	Noeud alternatif N11	- I1 : Continuellement actif - I2 : Continuellement passif
Utiliser 1 API	Noeud alternatif N12	- J1 : Arrêt avec alarme - I2 : Arrêt sans alarme
Utiliser 2 API avec priorité alarme	Noeud alternatif N12	- J1 : Arrêt avec alarme - I2 : Arrêt sans alarme
Utiliser 2 API sans priorité alarme	Noeud alternatif N12	- J1 : Arrêt avec alarme - I2 : Arrêt sans alarme
Utiliser 1 Alimentation	Noeud alternatif N13	- K1 : Arrêt inattendu
Utiliser 2 Alimentations en redondance active	Noeud alternatif N13	- K1 : Arrêt inattendu
Utiliser 2 Alimentations en redondance passive	Noeud alternatif N13	- K1 : Arrêt inattendu
Détecteur Température standard	Pas de noeud	- L1 : Continuellement passif - L2 : Continuellement actif
Détecteur Température sûr type 1	Pas de noeud	- L1 : Continuellement passif - L2 : Continuellement actif
Détecteur Température sûr type 2	Pas de noeud	- L1 : Continuellement passif - L2 : Continuellement actif
Détecteur Fumée standard	Pas de noeud	- M1 : Continuellement passif - M2 : Continuellement actif
Détecteur Fumée sûr type 1	Pas de noeud	- M1 : Continuellement passif - M2 : Continuellement actif
Détecteur Fumée sûr type 2	Pas de noeud	- M1 : Continuellement passif - M2 : Continuellement actif
API standard	Pas de noeud	- N1 : Arrêt inattendu avec alarme - N2 : Arrêt inattendu sans alarme
API sûr type 1	Pas de noeud	- N1 : Arrêt inattendu avec alarme - N2 : Arrêt inattendu sans alarme
API sûr type 2	Pas de noeud	- N1 : Arrêt inattendu avec alarme - N2 : Arrêt inattendu sans alarme
Alimentation standard	Pas de noeud	- O1 : Arrêt inattendu
Alimentation sûr	Pas de noeud	- O1 : Arrêt inattendu

Tableau 3. *Correspondance entre fonctions et modes de défaillances de la figure 5*

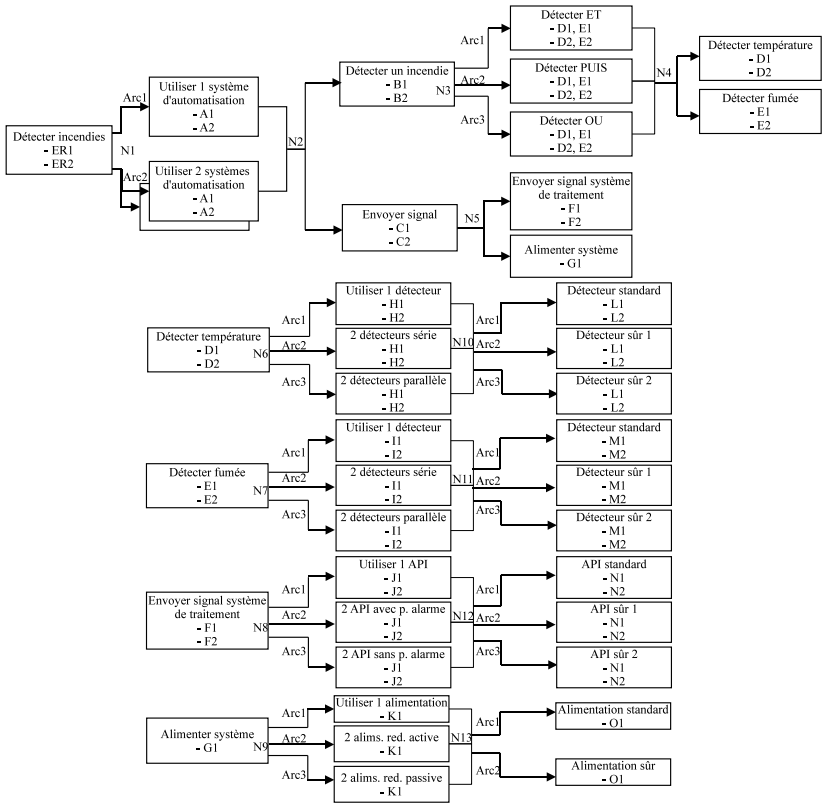

Figure 5. Arbre de défaillances du système de détection incendie automatique

4.3. *Phase d'optimisation*

4.3.1. *Ensemble d'architectures opérationnelles optimales*

L'évaluation de l'arbre de défaillances multiples amélioré par la méthode d'optimisation donne 74 solutions optimales pour le système de détection des incendies automatique. La Table 4 synthétise ces solutions par rapport aux deux événements redoutés. Le nombre de solutions ainsi que le coût minimum et maximum sont donnés pour chaque niveau de sûreté de fonctionnement. Par exemple, si le concepteur souhaite un système robuste à trois défaillances pour chaque événement redouté (donc de longueur égale à 4), il aura le choix parmi 9 systèmes dont le coût varie entre 38 et 49 unités. Si le coût le plus faible est recherché, la solution au coût de 38 pourra être choisie. Cependant en investissant plus, le concepteur pourra opter pour une solution qui, tout en respectant la limite des 3 défaillances tolérées, offre une probabilité moindre d'occurrence d'incidents par un nombre plus réduit de scénarios menant à ceux-ci. Par ailleurs, pour chaque solution optimale, la méthodologie donne les composants de

base et leur organisation dans l'architecture proposée (type de redondance, nombre et type de composants.).

Nombre de systèmes et coûts		Longueur minimale des scénarios pour ER_1			
		1	2	3	4
Longueur minimale des scénarios pour ER_2	1	2 systèmes C : 10	4 systèmes C : 13 à 16	2 systèmes C : 19, 20	1 système C : 25
	2	4 systèmes C : 13 à 16	2 systèmes C : 20	6 systèmes C : 23 à 25	9 systèmes C : 26 à 31
	3	2 systèmes C : 19, 20	3 systèmes C : 23 à 25	10 systèmes C : 26 à 31	7 systèmes C : 32 à 36
	4	2 systèmes C : 25, 26	2 systèmes C : 29, 30	9 systèmes C : 32 à 37	9 systèmes C : 38 à 49

Tableau 4. *Synthèse des systèmes trouvés de protection de l'incendie*

La Table 5 montre quelques systèmes issus de la Table 3 dont le L_{min} est égal à 3 pour chaque événement redouté. On constate dans cette table que si des composants sont ajoutés, le coût du système augmente (il passe de 26 à 31 unités) et le nombre de scénarios (N_{min}) diminue pour les deux événements redoutés (Pour l'événement redouté *Pas d'alarme incendie*, il passe de 36 scénarios pour le système ayant un coût de 26 unités à 6 scénarios pour le système ayant un coût de 31 unités). Cette diminution atteint un niveau minimal. Cela est dû à un nombre très important de composants dans cette architecture. En effet, plus il y a de composants dans l'architecture, plus importante est la probabilité d'un composant d'être défaillant. Par exemple, dans la Table 5, la solution ayant un coût de 31 unités montre que N_{min} augmente pour l'événement redouté fausse alarme par rapport à la solution ayant un coût de 30 unités. Il est donc important de choisir une bonne architecture représentant un bon compromis entre le coût et le niveau de sûreté de fonctionnement.

Plus spécifiquement, une solution optimale obtenue avec l'approche proposée est présentée Figure 6. Cette solution a un coût de 31 unités et son L_{min} est de 3 pour les deux événements redoutés. Cette architecture est composée de deux systèmes d'automatisation (Système N°1 et N°2) en redondance passive. Le système N°1 détecte la fumée puis la température à l'aide d'un capteur de température standard et d'un capteur de fumée standard. Il comprend 2 automates standards en redondance active

Systèmes ayant une longueur minimale de 3 pour ER_1 et ER_2					
Coût du système		26	28	30	31
Opérateurs temporels	N_{min} pour ER_1	36	24	9	6
	N_{min} pour ER_2	27	18	6	12
Opérateurs classiques	N_{min} pour ER_1	60	30	24	12
	N_{min} pour ER_2	60	30	15	36

Tableau 5. *Détails de systèmes issus de la table 4*

avec priorité d'alarme. Il est alimenté par 2 alimentations standards en redondance passive. Le système N˚2 détecte la fumée puis la température à l'aide d'un capteur de température standard et d'un capteur de fumée standard. Il comprend deux automates standards en redondance active. Il est alimenté par 1 alimentation robuste. Les deux systèmes envoient leurs informations aux opérateurs du système.

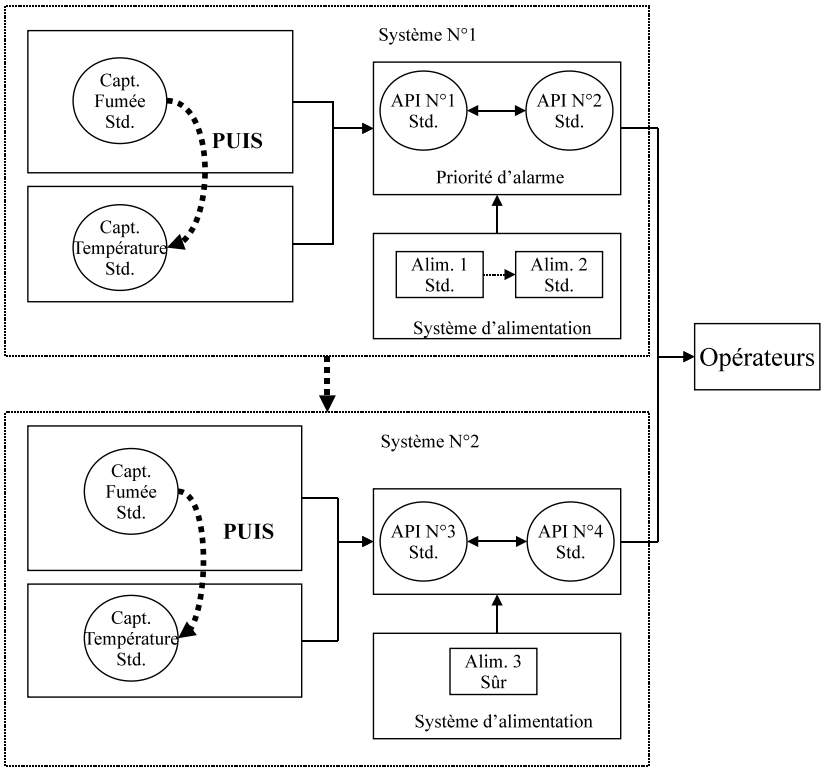

Figure 6. Exemple d'architecture opérationnelle optimale

4.3.2. *Comparaison avec les arbres de défaillances classiques*

Dans cette section, l'approche proposée est comparée avec la méthode d'évaluation traditionnelle à base d'arbres de défaillances. Nous rappelons ici que dans les arbres de défaillances classiques, seuls les opérateurs *ET* et *OU* sont utilisés et que les aspects temporels ne sont pas considérés. L'évaluation de l'arbre de défaillance classique par la méthode d'optimisation donne 117 solutions optimales. La Table 6 synthétise ces solutions. Nous constatons que l'ensemble des solutions issu de l'arbre de défaillances multiples amélioré est inclus dans l'ensemble des solutions issu de l'arbre de défaillances classique (Table 4). En effet, l'utilisation des opérateurs temporels n'a pas d'influence sur la limite du nombre de défaillances tolérées, mais sur

Nombre de systèmes et coûts		Longueur minimale des scénarios pour ER_1			
		1	2	3	4
Longueur minimale des scénarios pour ER_2	1	6 systèmes C : 10 à 12	8 systèmes C : 13 à 18	4 systèmes C : 19 à 22	1 système C : 25
	2	8 systèmes C : 13 à 16	7 systèmes C : 20 à 22	12 systèmes C : 23 à 25	16 systèmes C : 26 à 31
	3	4 systèmes C : 19 à 22	4 systèmes C : 23 à 25	14 systèmes C : 26 à 31	11 systèmes C : 32 à 37
	4	4 systèmes C : 25 à 28	3 systèmes C : 29 à 31	11 systèmes C : 32 à 37	8 systèmes C : 38 à 50

Tableau 6. *Synthèse des systèmes trouvés avec l'arbre de défaillance classique*

l'évaluation du nombre de scénarios menant aux événements redoutés. Ainsi, la méthode proposée, par rapport aux arbres de défaillances classiques permet de déterminer des solutions meilleures en termes de nombre de scénarios. De ce fait, lors de l'optimisation de l'arbre de défaillances multiples, l'existence de ces nouvelles architectures permet d'éliminer plus de possibilités d'architectures non optimales en comparaison d'une optimisation avec l'arbre classique.

Plus spécifiquement, les solutions obtenues avec l'approche proposée sont plus précises sur la quantification du niveau de sécurité-disponibilité, en raison d'un nombre de scénarios (N_{min}) plus faible pour chaque événement redouté par rapport à l'arbre classique (Table 5). En effet, les opérateurs temporels suppriment les scénarios impossibles du fait de l'architecture choisie contrairement aux coupes de l'arbre de défaillances. Par exemple, la solution optimale obtenue avec l'approche proposée présentée Figure 6 et qui correspond à la solution ayant un coût de 31 dans la Table 5 possède 6 scénarios amenant vers l'événement redouté « Pas d'alarme incendie » avec l'approche proposée contre 12 scénarios avec l'approche classique. Il y a donc 6 scénarios impossibles dont on tient compte avec l'approche classique. L'un de ces scénarios est Arrêt inattendu alim. 2 du système 1, Arrêt inattendu alim. 1 du système 1, Arrêt inattendu alim. 3 du système 2. Ce scénario est impossible car les alimentations 1 et 2 du système 1 sont en redondance passive (Figure 6) et donc l'alimentation 2 ne peut avoir son mode de défaillance avant celui de l'alimentation 1. Notre approche, à la différence de l'approche classique, permet de prendre en compte ces différences de comportement.

5. Conclusion

Dans cet article, une méthodologie complète de conception de systèmes d'automatisation sûrs de fonctionnement est présentée. Le point innovant de cette méthodologie est qu'elle utilise les séquences de modes de défaillances ordonnées dans le temps. Une nouvelle représentation, dénommée arbre de défaillances multiples amélioré, est définie. Cette représentation permet de modéliser les relations entre défaillances de

chacune des fonctions et d'évaluer le niveau de sûreté de fonctionnement d'un ensemble d'architectures matérielles en utilisant des séquences de modes de défaillances. La méthode de conception proposée fournit un ensemble d'architectures optimales caractérisées par un coût et un niveau de sûreté de fonctionnement. Le concepteur peut ainsi choisir l'architecture qui correspond le plus à ses besoins. La comparaison entre l'approche proposée et une méthode d'évaluation classique montre que l'ensemble de solutions optimales trouvées avec l'arbre de défaillances multiples est plus petit que l'ensemble de solutions optimales trouvées avec l'arbre classique. L'ensemble est plus petit mais les solutions sont mieux évaluées car l'approche proposée intègre des fonctions temporelles et évalue plus précisément le niveau de sûreté de fonctionnement.

Les futurs travaux de recherche visent à améliorer l'algorithme de comparaison dans le but de concevoir des systèmes d'automatisation plus complexes ayant un plus grand nombre de composants et de fonctions. Ces futurs travaux visent également à prendre en compte les ressources partagées entre composants pour évaluer les valeurs caractéristiques de la sûreté de fonctionnement de l'arbre de défaillances amélioré.

Ce travail a été réalisé grâce au soutien :

– du CISIT : Campus International sur la Sécurité et l'Intermodalité dans les Transports,

– de la région Nord-Pas-de-Calais,

– de l'union européenne,

– du ministère de l'enseignement supérieur et de la recherche

– du CNRS : Centre National de la Recherche Scientifique

– de l'INRETS : Institut National de Recherche sur les Transports et leur Sécurité.

6. Bibliographie

AFNOR French Standards NF X60-500. Terminologie relative à la Fiabilité-Maintenabilité-Disponibilité 1988.

AFNOR French Standards NF EN 292. Sécurité des machines Notions fondamentales, Principes généraux de conception, partie 1 : terminologie de base, méthodologie 1991.

AFNOR French Standards NF X50-151. Management de la valeur et ses outils, analyse fonctionnelle, analyse de la valeur, conception à objectif désigné, 2004.

Amer H. H., Daoud R.M., « Fault Secure multi-detectors Fire protection System for trains » IEEE IMTC – Instrumentation and Measurement Technology Conference, Ottawa, Canada, 17-19 May 2005.

Benard V., Cauffriez L., Renaux D., «The Safe-SADT method for aiding designers to choose and improve dependable architectures for complex automated systems » Journal of Reliability Engineering and System Safety, 93/2, pages 179-196, 2008.

Bouissou M., Bon J.L., « A new formalism that combines advantages of fault-trees and Markov models : Boolean logic driven Markov processes » Reliability Engineering and System Safety 82, Elsevier Ed., pages 149-163, 2003.

Bouissou M., Dutuit Y., « Reliability analysis of a dynamic phased mission system », MMR2004 congress, Santa Fe, June 2004.

Cauffriez L., Ciccotelli J., Conrard B., Bayart M.,« Design of intelligent distributed control systems : a dependability point of view » Reliability Engineering and System Safety 84, Elsevier Ed., pages 19-32, 2004.

Cepin M., Mavko B., « A dynamic fault tree » Reliability Engineering and System Safety 75, Elsevier Ed., pages 83-91, 2002.

Clarhaut J., Hayat S., Conrard B., Cocquempot V., « Safety intelligent system conception for piggyback service » IEEE ICIT Industrial Conference on Industrial Technology, Volume 6, pages 1659-1664, ISBN 1-4244-0726-5, 2006.

Clarhaut J., Hayat S., Conrard B., Cocquempot V., « Safety system conception by using a semi-quantitative reliability evaluation application. Application to railroad transportation systems » in Proceedings of international conference on Industrial Engineering and Systems Management (IESM 2007), May 30 – June 2, pages 330-331, ISBN 978-7-302-15312-2, Tsinghua University Press, Beijing, China, 2007.

Conrard B., Cocquempot V., Bayart M., « Design of Automation Systems with criterion of cost and dependability », Qualita 2007 congress, Tanger, Maroco, 20-22 March 2007.

Conrard B., Bayart M., « Design of Dependable Control System thanks to a semi-quantitative Optimisation », submission accepted to Safety and Reliability for Managing Risk (ESREL 06), Estoril, 18-22 September 2006.

Coppit D., Sullivan K.J., Dugan J.B., « Formal Semantics for Computational Engineering : A case Study on Dynamic Fault Trees » ISSRE, pp. 270, 11th International Symposium on Software Reliability Engineering (ISSRE'00), 2000.

D'Ariano A., Pacciarelli D., Pranzo M., « A branch and bound algorithm for scheduling trains in a railway network » European Journal of Operational Research, Volume 183, Issue 2, Pages 643-657 Science Direct Ed, 2006.

Dugan J.B., Bavuso S.J., Boyd M.A., « Dynamic fault-tree models for tolerant computer systems » IEEE Transactions on Reliability, pp. : 363-377, Volume : 41, Issue 3, September 1992.

Dugan J.B., « Fault tree Analysis of Computer-Based Systems » Proceedings of Annual Reliability and Maintainability Symposium, Philadelphia USA, January 2001.

Dutuit Y., Rauzy A., « Exact and Truncated Computations of Prime Implicants of Coherent and non-Coherent Fault Trees within Aralia » Reliability Engineering and System Safety 58, Elsevier Ed., pages 127-144, 1997.

Eisinger S., Rakowsky U.K., « Modeling of uncertainties in reliability centered maintenance – a probabilistic approach », Reliability Engineering and System Safety 71, Elsevier Ed., pages 159-164, 2001.

Elsayed H. M., Amer H.H., Daoud R.M., « Fire protection System for Cargo Trains using fuzzy logic » IEEE Workshop on Soft Computing in industrial Applications, Helsinki University of Technology, Espoo, Finland, June 28-30 2005.

IEC 60300-3-1 Dependability management. Part 3-1 : Application guide, Analysis techniques for dependability – guide on methodology. Geneva, Switzerland, IEC, International Electrotechnical Commission, ISBN 2-8318-6791-6, 2003.

Jampi D., Aubry J.F., Guilhem E., « Conception et sûreté de fonctionnement : deux activités indissociables » Congrès Francophone Mosim 2001, Troyes, 2001.

Jiang G., Shang F., Wang F., « A combined Intelligent Fire Detector with BP Networks » IEEE Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China, June 21-23, 2006.

Kerhen C., Seguin C., « Evaluation qualitative de systèmes physiques pour la sûreté de fonctionnement » Formalisation des activités concurrentes (FAC03), Toulouse, France, 2003.

Kumamoto H., Henley E.J., « Probabilistic risk assessment and management for engineers and scientists" New York, IEEE Press, ISBN 0-780-31004-7, 1996.

Laprie J.C., Guide de la sûreté de fonctionnement, Cépaduès Ed., Toulouse, 1995.

Moncelet G., Dependability evaluation of mecatronic automative systems using Petri Nets PhD Thesis in French, Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS, 1998.

Rausand M., Hoyland A., System Reliability Theory : Models, Statistical Methods and Applications Second Edition, Wiley Ed., Pages 99-103, 2004.

Rauzy A., « Mathematical Foundation of Minimal Cutsets » IEEE Transactions on Reliability, Vol. 50-4, pages 389-396, 2001.

Sallak M., Simon C., Aubry J.F., « Optimal design of Safety Instrumented Systems » Workshop on Advanced Control and Diagnosis, ACD'06, Nancy, France, 2006.

Schoenig R., Aubry J.F., Cambois T., Hutinet T., « An aggregation method of Markov graphs for the reliability analysis of hybrid system » Reliability Engineering and System Safety 91, Elsevier Ed., pages 137-148, 2006.

Simon C., Thiriet J.M., Barger P., « Reliability and credibility evaluation of networked control systems » Advances in Safety and Reliability – ESREL, Pologne, 2005

Simonot Lion F., Thomesse J.P., Bayart M., Staroswiecki M., « Dependable distributed computer control systems : analysis of the design step activities » in Sharaoui AEK, editor 13th IFAC Workshop on Distributed Control Systems, Toulouse, France, pages 119-124, 1995.

Sourisse C., Boudillon L., La sécurité des machines automatisées, Editions Institut Schneider Formation, 2 tomes, 1997

Swaminathan S., Smidts C., « The mathematical formulation for the event sequence diagram framework » Reliability Engineering and System Safety 65, Elsevier Ed., pages 103-118, 1999.

Villemeur A., Dependability of industrial systems. Book in French, Eyrolles, Paris, ISSN 0339-4198, 1988.

Villemeur A., Reliability, Maintainability and Safety Assessment Methods and techniques, Vol. n˚1, ISBN-13 : 978-0471930488, Wiley, 1st edition, February 4, 1992 ;

Zwingelstein G., Sûreté de fonctionnement des systèmes industriels complexes, Techniques de l'ingénieur N˚ S8250, 1999.

7. Annexes

7.1. *Principe général de l'algorithme d'optimisation*

Le principe général de l'optimisation est de déterminer l'ensemble des solutions optimales ou *systèmes optimaux*. Cet algorithme est décomposable en deux étapes :

– La première étape est de subdiviser le problème d'optimisation en sous-problèmes selon une démarche ascendante de scrutation des noeuds de l'arbre. Ainsi, chaque noeud de l'arbre est considéré comme un sous-problème d'optimisation sur lequel nous allons ensuite déterminer les solutions optimales pour la réalisation de chaque fonction de ce noeud. Cette démarche de décomposition en sous-problèmes n'est valide que sous la contrainte d'indépendance des fonctions.

– La seconde étape est de traiter chaque sous-problème d'optimisation en deux temps :

- La génération des solutions admissibles pour la réalisation d'une fonction.

- La sélection des solutions optimales par comparaison des solutions deux à deux.

La figure 7 présente le principe général de l'algorithme d'optimisation

7.1.1. *Génération des solutions*

La génération des solutions admissibles (solutions réalisables mais pas forcément optimales) dépend du type de noeud considéré et n'est valable que grâce à l'hypothèse d'indépendance des fonctions de l'arbre :

– Pour le noeud alternatif, l'ensemble des solutions admissibles est déterminé par l'union des différentes solutions admissibles de chaque fonction.

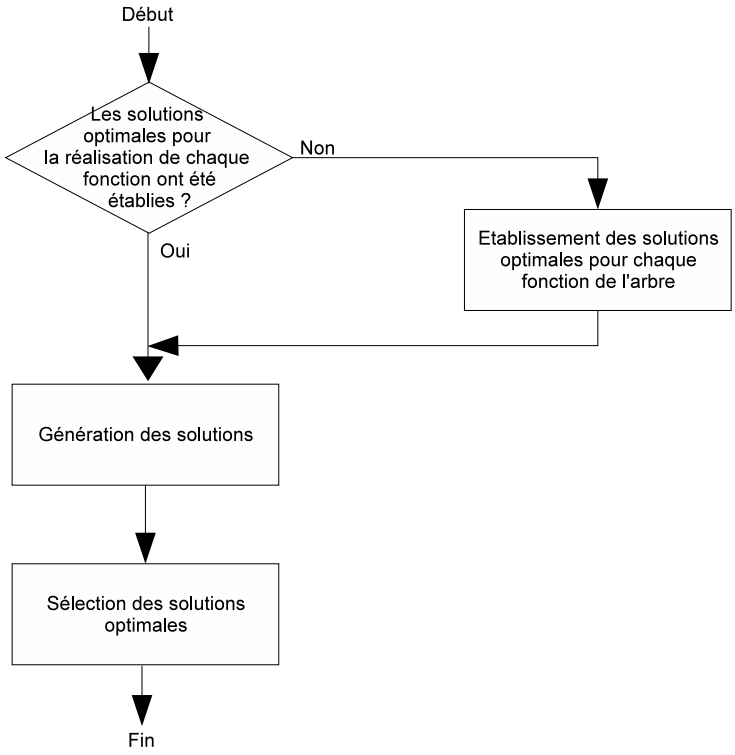

Figure 7. Principe général du mécanisme d'optimisation

– Pour le noeud associatif, l'ensemble des solutions admissibles est déterminé par scrutation de toutes les combinaisons de solutions admissibles de chaque fonction requise. Pour chaque combinaison de solutions, une solution est établie par l'évaluation :

- du coût par addition des coûts individuels des solutions de la combinaison étudiée.

- du niveau de sûreté de fonctionnement en utilisant, pour chaque mode de défaillance, les lois de composition correspondantes aux opérateurs de la relation associée à ce mode de défaillance.

– Pour le noeud élémentaire utilisé à la base de l'arbre, l'ensemble des solutions admissibles est composé d'une seule solution caractérisée par le coût de son composant et par un niveau de sûreté de fonctionnement composé des couples :

- La valeur du CFR permettant de caractériser sa robustesse (standard, sécuritaire, durci, ...) pour le mode de défaillance considéré.

- Le nombre de scénarios amenant au mode de défaillance considéré (N_{min}) égal à 1.

La figure 8 présente ce mécanisme d'établissement tenant compte des trois types de noeuds.

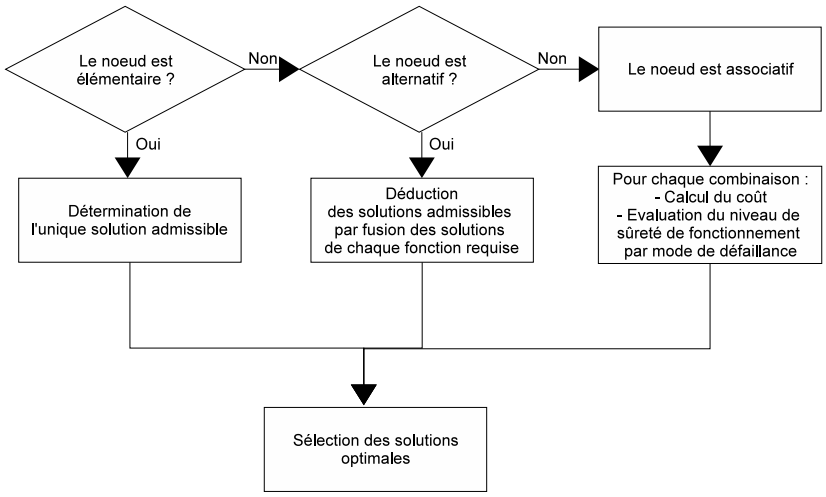

Figure 8. Schéma du mécanisme d'établissement des solutions suivant le type de noeud pris en considération

7.1.2. *Sélection des solutions optimales*

Les solutions de l'ensemble des solutions admissibles sont comparées entre elles deux à deux. Les solutions moins bonnes et donc non-optimales sont ensuite éliminées de l'ensemble.
A la fin de l'étape d'optimisation, la méthodologie fournit l'ensemble Pareto-optimal, c'est-à-dire qu'il n'existe pas de solutions qui fassent diminuer un critère sans augmenter dans le même temps au moins un autre critère. La figure 9 représente le principe de comparaison puis d'élimination des solutions non-optimales.

7.2. *Démonstration des lois de composition des opérateurs logiques*

Pour chaque démonstration, considérons A, B et C, trois événements redoutés, tels que C est le résultat de l'association de A et de B avec l'un des opérateurs (ET, p-ET ou SEQ). Δ_A, Δ_B et Δ_C sont les ensembles de scénarios minimaux associés à A, B et C. Nous prenons pour hypothèse que les événements redoutés A et B sont indépendants, c'est-à-dire qu'une défaillance ne peut pas apparaître simultanément dans A et B.

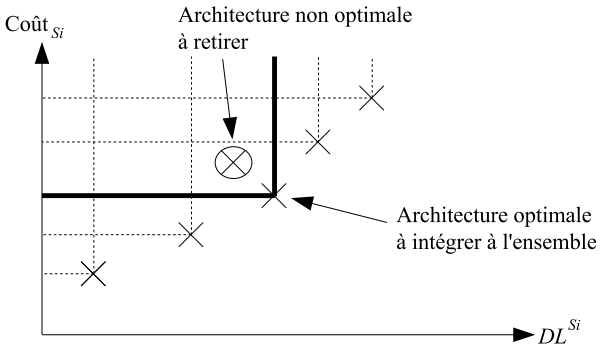

Figure 9. Constitution d'un ensemble de solution d'après (Conrard et Bayart, 2006)

7.2.1. *Lois de composition de l'opérateur ET*

L'opérateur ET représente le cas où l'occurrence de C se produit après l'occurrence de A et de B. Les valeurs caractéristiques correspondantes L^C_{min} et N^C_{min} peuvent être déterminés grâce au nombre de permutations entre 2 séquences de longueurs respectives L^A_{min} et L^B_{min}, noté $R^{L^A_{min},L^B_{min}}_{L^C_{min}}$, et grâce au nombre de combinaisons entre les scénarios des deux ensembles Δ_A et Δ_B.
Ainsi, l'ensemble Δ_C peut être caractérisé par les relations suivantes :

$$L^C_{min} = L^A_{min} + L^B_{min}$$

$$N^C_{min} = R^{L^A_{min},L^B_{min}}_{L^C_{min}} \times N^A_{min} \times N^B_{min} \; avec \; R^{L^A_{min},L^B_{min}}_{L^C_{min}} = \frac{L^C_{min}!}{L^B_{min}! \times L^A_{min}!}$$

7.2.2. *Lois de composition de l'opérateur p-ET*

L'opérateur p-ET est un opérateur temporel qui représente le cas où l'occurrence de C se produit après les occurrences successives de A puis de B. De la même façon que pour l'opérateur ET, L^C_{min} et N^C_{min} peuvent être déterminés grâce au nombre de permutations entre 2 séquences de longueurs respectives L^A_{min} et L^B_{min}, noté $R^{L^A_{min},L^B_{min}}_{L^C_{min}}$, et grâce au nombre de combinaisons entre les scénarios des deux ensembles Δ_A et Δ_B. Cependant, afin de prendre en compte les occurrences successives de A puis de B, cet opérateur impose une contrainte sur le premier mode de défaillance de l'ensemble Δ_A telle que la permutation de ce mode de défaillance ne soit pas possible.
Ainsi N^C_{min} peut être déterminé grâce au nombre de permutations possibles entre 2 séquences de longueurs respectives $L^A_{min} - 1$ et L^B_{min} tout en conservant une longueur finale pour les scénarios de l'ensemble Δ_C égale à $L^C_{min} = L^A_{min} + L^B_{min}$. De ce fait, l'ensemble Δ_C peut être caractérisé par les relations suivantes :

$$L^C_{min} = L^A_{min} + L^B_{min}$$

$$N^C_{min} = R^{(L^A_{min}-1),L^B_{min}}_{(L^A_{min}-1)+L^B_{min}} \times N^A_{min} \times N^B_{min},$$

$$avec\ R^{(L^A_{min}-1),L^B_{min}}_{(L^A_{min}-1)+L^B_{min}} = \frac{((L^A_{min}-1)+L^B_{min})!}{(L^A_{min}-1)! \times L^B_{min}!}$$

7.2.3. *Lois de composition de l'opérateur SEQ*

L'opérateur SEQ, comme l'opérateur p-ET, représente le cas où l'occurrence de C se produit après les occurrences successives de A puis de B. Cependant, l'opérateur SEQ impose qu'aucune défaillance amenant le système vers l'événement redouté B ne peut se produire avant l'occurrence de l'ensemble des défaillances amenant à l'événement redouté A. Dans la théorie des ensembles, cet opérateur correspond à un produit cartésien entre les ensembles Δ_A et Δ_B dont le résultat est l'ensemble Δ_C.
Ainsi L^C_{min} et N^C_{min} peuvent être déterminés grâce au nombre de combinaisons entre les scénarios de longueurs respectives L^A_{min} et L^B_{min} des deux ensembles Δ_A et Δ_B. De ce fait, l'ensemble Δ_C peut être caractérisé par les relations suivantes :

$$L^C_{min} = L^A_{min} + L^B_{min}$$
$$N^C_{min} = N^A_{min} \times N^B_{min}$$

7.3. *Démonstration de la relation entre le CFR et les probabilités d'occurrence de scénarios*

Dans cette section, nous montrons que l'inégalité $CFR_{\psi_1} < CFR_{\psi_2}$ implique que la probabilité d'occurrence de la séquence ψ_1 est supérieure à celle de ψ_2 sous certaines hypothèses peu restrictives.
Tout d'abord, pour un intervalle de temps donné, la probabilité d'occurrence d'une séquence ψ est donnée par la relation suivante :

$$Prob(\psi) = \frac{\prod r^{CFR_{F_i}}}{card(\psi)!}$$

On considère pour cette relation que chaque faute F_i de cette séquence ψ a la probabilité d'occurrence suivante :

$$Prob(F_i) = r^{CFR_{F_i}}$$

Par ailleurs, les valeurs des CFR affectées aux différents modes de défaillance sont des valeurs discrètes choisies par le concepteur. Ainsi, pour 2 valeurs distinctes

de CFR, un pas minimum est employé, noté Δ_{CFR}.
Ainsi :

$$CFR_{\psi_1} < CFR_{\psi_2} \Longleftrightarrow CFR_{\psi_2} - CFR_{\psi_1} \geq \Delta_{CFR}$$

d'où :

$$CFR_{\psi_1} < CFR_{\psi_2} \Longleftrightarrow r^{CFR_{\psi_2} - CFR_{\psi_1}} \leq r^{\Delta_{CFR}}$$

puisque 0 < r « 1.

Si la condition suivante est satisfaite, on montre que l'inégalité $CFR_{\psi_1} < CFR_{\psi_2}$ implique que la probabilité d'occurrence de la séquence ψ_1 est supérieure à celle de ψ_2.

$$r^{\Delta_{CFR}} < \frac{1}{lg!}$$

Cette hypothèse s'appuie sur le fait que r est petit et que les séquences de fautes étudiées sont d'une longueur maximale connue, notée lg (généralement de l'ordre de 2, 3 ou 4). Or puisque :

$$1 \leq card(\psi_1) \leq lg \; et \; 1 \leq card(\psi_2) \leq lg$$

on obtient :

$$r^{\Delta_{CFR}} < \frac{1}{lg!} \leq \frac{card(\psi_2)!}{card(\psi_1)!}$$

ainsi :

$$r^{CFR_{\psi_2} - CFR_{\psi_1}} \leq r^{\Delta_{CFR}} \leq \frac{card(\psi_2)!}{card(\psi_1)!}$$

d'où :

$$\frac{r^{CFR_{\psi_2}}}{card(\psi_2)!} < \frac{r^{CFR_{\psi_1}}}{card(\psi_1)!}$$

soit, au final :

$$CFR_{\psi_2} > CFR_{\psi_1} \Longrightarrow Prob(\psi_2) < Prob(\psi_1)$$

7.4. *Relations entre modes de défaillances du système de protection contre les incendies*

Relations entre modes de défaillances de la figure 5 :

$$\text{(N1)} \begin{cases} Arc1 \begin{cases} Syst.commande(A_1) \Rightarrow ER_1 \\ Syst.commande(A_2) \Rightarrow ER_2 \end{cases} \\ Arc2 \begin{cases} Syst.commande_1(A_1) \textbf{ SEQ } Syst.commande_2(A_2) \Rightarrow ER_1 \\ (Syst.commande_1(A_1) \textbf{ SEQ } Syst.commande_2(A_2)) \\ \textbf{OU } (Syst.commande_1(A_2) \textbf{ SEQ } Syst.commande_2(A_2)) \Rightarrow ER_2 \end{cases} \end{cases}$$

$$\text{(N2)} \begin{cases} Detec.incendie(B_1) \textbf{ OU } Envoi.signal(C_1) \Rightarrow A_1 \\ Detec.incendie(B_2) \textbf{ OU } Envoi.signal(C_2) \Rightarrow A_2 \end{cases}$$

$$\text{(N3)} \begin{cases} Arc1 \begin{cases} Detec.Temp(D_1) \textbf{ OU } Detec.Fumee(E_1) \Rightarrow B_1 \\ Detec.Temp(D_2) \textbf{ ET } Detec.Fumee(E_2) \Rightarrow B_2 \end{cases} \\ Arc2 \begin{cases} Detec.Temp(D_1) \textbf{ ET } Detec.Fumee(E_1) \Rightarrow B_1 \\ Detec.Fumee(E_2) \textbf{ p-ET } Detec.Temp(D_2) \Rightarrow B_2 \end{cases} \\ Arc3 \begin{cases} Detec.Temp(D_1) \textbf{ ET } Detec.Fumee(E_1) \Rightarrow B_1 \\ Detec.Temp(D_2) \textbf{ OU } Detec.Fumee(E_2) \Rightarrow B_2 \end{cases} \end{cases}$$

$$\text{(N4)} \begin{cases} D_1 \Rightarrow D_1 \\ D_2 \Rightarrow D_2 \\ E_1 \Rightarrow E_1 \\ E_2 \Rightarrow E_2 \end{cases}$$

$$\text{(N5)} \begin{cases} Syst.traitement(F_2) \textbf{ OU } Alim.systeme(G_1) \Rightarrow C_1 \\ Syst.traitement(F_1) \Rightarrow C_2 \end{cases}$$

$$\text{(N6)} \begin{cases} Arc1 \begin{cases} Detec.Temp(H_2) \Rightarrow D_1 \\ Detec.Temp(H_1) \Rightarrow D_2 \end{cases} \\ Arc2 \begin{cases} (Detec.Temp_1(H_2) \textbf{ OU } Detec.Temp_2(H_2)) \Rightarrow D_1 \\ (Detect.Temp_1(H_1) \textbf{ ET } Detec.Temp_2(H_1)) \Rightarrow D_2 \end{cases} \\ Arc3 \begin{cases} (Detec.Temp_1(H_2) \textbf{ ET } Detec.Temp_2(H_2)) \Rightarrow D_1 \\ (Detec.Temp_1(H_1) \textbf{ OU } Detec.Temp_2(H_1)) \Rightarrow D_2 \end{cases} \end{cases}$$

$$\text{(N7)} \begin{cases} Arc1 \begin{cases} Detec.Fumee(I_2) \Rightarrow E_1 \\ Detec.Fumee(I_1) \Rightarrow E_2 \end{cases} \\ Arc2 \begin{cases} (Detec.Fumee_1(I_2) \textbf{ OU } Detec.Fumee_2(I_2)) \Rightarrow E_1 \\ (Detec.Fumee_1(I_1) \textbf{ ET } Detec.Fumee_2(I_1)) \Rightarrow E_2 \end{cases} \\ Arc3 \begin{cases} (Detec.Fumee_1(I_2) \textbf{ ET } Detec.Fumee_2(I_2)) \Rightarrow E_1 \\ (Detec.Fumee_1(I_1) \textbf{ OU } Detec.Fumee_2(I_1)) \Rightarrow E_2 \end{cases} \end{cases}$$

$$\text{(N8)} \begin{cases} Arc1 \begin{cases} API(J_1) \Rightarrow F_1 \\ API(J_2) \Rightarrow F_2 \end{cases} \\ Arc2 \begin{cases} (API_1(J_1) \textbf{ OU } API_2(J_1)) \Rightarrow F_1 \\ (API_1(J_2) \textbf{ ET } API_2(J_2)) \Rightarrow F_2 \end{cases} \\ Arc3 \begin{cases} (API_1(J_1) \textbf{ ET } API_2(J_1)) \Rightarrow F_1 \\ (API_1(J_2) \textbf{ OU } API_2(J_2)) \Rightarrow F_2 \end{cases} \end{cases}$$

$$\text{(N9)} \begin{cases} Arc1 \begin{cases} Alim.(K_1) \Rightarrow G_1 \end{cases} \\ Arc2 \begin{cases} Alim._1(K_1) \textbf{ ET } Alim._2(K_1) \Rightarrow G_1 \end{cases} \\ Arc3 \begin{cases} Alim._1(K_1) \textbf{ SEQ } Alim._2(K_1) \Rightarrow G_1 \end{cases} \end{cases}$$

$$\text{(N10)} \begin{cases} Arc1 \begin{cases} Detec.Temp - Standard(L_1) \Rightarrow H_1 \\ Detec.Temp - Standard(L_2) \Rightarrow H_2 \end{cases} \\ Arc2 \begin{cases} Detec.Temp - Sur1(L_1) \Rightarrow H_1 \\ Detec.Temp - Sur1(L_2) \Rightarrow H_2 \end{cases} \\ Arc3 \begin{cases} Detec.Temp - Sur2(L_1) \Rightarrow H_1 \\ Detec.Temp - Sur2(L_2) \Rightarrow H_2 \end{cases} \end{cases}$$

Les noeuds N11 à N13 ne sont pas détaillés car ils reprennent la même structure que le noeud N10 à la différence que leurs modes de défaillances correspondent à

ceux de la figure 5 et que le terme *Detec.Temp-Standard* est remplacé par celui du composant correspondant (Detec.Fumee-Sur1, Alim.-Standard, ...).