

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Automatique et robotique

Présentée et soutenue par :

Mme QUYNH ANH DO HOANG

le mardi 17 mars 2015

Titre :

ANALYSE ET JUSTIFICATION DE LA SECURITE DE SYSTEMES ROBOTIQUES EN INTERACTION PHYSIQUE AVEC L'HUMAIN

Ecole doctorale :

Systèmes (Systèmes)

Unité de recherche :

Laboratoire d'Analyse et d'Architecture des Systèmes (L.A.A.S.)

Directeur(s) de Thèse :

M. MOHAMED KAANICHE

M. JEREMIE GUIOCHET

Rapporteurs :

Mme NICOLE LEVY, CNAM PARIS

M. WALTER SCHON, UNIVERSITE DE TECHNOLOGIE DE COMPIEGNE

Membre(s) du jury :

M. GILLES MOTET, INSA TOULOUSE, Président

M. DAVID ANDREU, UNIVERSITE MONTPELLIER 2, Membre

M. JEREMIE GUIOCHET, UNIVERSITE TOULOUSE 3, Membre

M. MOHAMED KAANICHE, LAAS TOULOUSE, Membre

Avant-propos

Les travaux de recherche présentés dans ce manuscrit ont été réalisés au Laboratoire d'Analyse et d'Architecture de Systèmes du Centre National de la Recherche Scientique (LAAS-CNRS). Je remercie les Directeurs successifs du LAAS-CNRS, Messieurs Raja Chatila, Jean-Louis Sanchez et Jean Arlat, de m'avoir accueillie dans ce laboratoire. Je tiens à exprimer ma gratitude envers Madame Karama KANOUN et Monsieur Mohamed KAÂNICHE pour m'avoir permis de travailler dans d'aussi bonnes conditions au sein de l'équipe Tolérance aux fautes et Sûreté de Fonctionnement informatique.

La rédaction de ce manuscrit m'a apprise une chose importante : la thèse n'est jamais un travail solitaire. En effet, je n'aurais jamais pu réaliser ce travail sans le soutien d'un grand nombre de personnes dont la générosité, la connaissance et l'attention portée à ma recherche m'ont permis de progresser pendant ces dernières années. J'adresse toute ma gratitude à toutes les personnes qui m'ont aidée dans la réalisation de ce travail.

Mes plus vifs remerciements vont aux directeurs de ma thèse, Messieurs Jérémie GUIOCHET, Mohamed KAÂNICHE et David POWELL. Ils ont été toujours disponibles et à l'écoute de mes nombreuses questions. Leur intérêt manifesté à l'égard de ma recherche et leurs conseils lors de nos échanges ont été indispensables au résultat final de cette thèse. Ils ont su me motiver, m'encourager et m'aider à me surpasser pour aller plus loin pendant toute la durée de ma thèse et plus particulièrement durant les derniers mois de rédaction. Pour tout cela, merci.

Je voudrais exprimer ma profonde reconnaissance à Madame Nicole LEVY et Monsieur Walter SCHÖN d'avoir accepté de relire cette thèse et d'en être rapporteurs. La version finale de ce mémoire a bénéficié de leur lecture très attentive et de leurs remarques précieuses. Je tiens à remercier Gilles MOTET d'avoir accepté d'être le Président du jury. Ma gratitude va également à Monsieur David ANDREU et tous les membres du jury.

Je remercie toutes les personnes formidables que j'ai rencontrées par le biais du LAAS. Merci à vous les membres et anciens membres de l'équipe TSF, les personnes des services techniques, administratifs et logistiques, les personnes dont les sourires échangés restent la preuve irréfutable d'une bonne ambiance de travail.

En dernier, ma gratitude la plus profonde va à ceux qui m'ont donné le courage nécessaire pour avancer. Je n'oublie jamais la main tendue d'Odon VALLET sans qui mon aventure n'aurait jamais commencé. Je remercie Henriette, Jean Jean et Étienne pour leur accueil chaleureux et de la seconde famille qu'ils m'ont offerte. J'adresse toute mon affection à ma famille qui me soutient de loin ou de près. Malgré mon éloignement depuis de nombreuses années, leur confiance, leur tendresse et leur amour me portent et me guident tous les jours. J'exprime mon amitié à mes amis qui ont cru en moi et qui ont su me divertir du travail quand j'en avais besoin. Merci pour avoir fait de moi ce que je suis aujourd'hui.

Une pensée très forte à toi Grande-Mère, toi qui me manque.

Table des matières

Introduction générale **6**

1 Maîtrise de la confiance en robotique de service **11**
1.1 Introduction 11
1.2 Gestion du risque 16
1.3 Technique d'analyse du risque HAZOP 22
1.4 Analyse quantitative : les réseaux bayésiens 26
1.5 Analyse quantitative : fonction de croyance 31
1.6 Argumentaire de sécurité 33
1.7 Vue générale des contributions de la thèse 37

2 Analyse du risque basée modèle **41**
2.1 Introduction 41
2.2 Utilisation des modèles dans l'analyse du risque 41
2.3 Langage de modélisation unifié UML 44
2.3.1 Diagramme des cas d'utilisation 46
2.3.2 Diagramme de séquence 47
2.3.3 Diagramme d'états-transitions 49
2.4 Méthode d'analyse du risque HAZOP-UML 51
2.4.1 Introduction 51
2.4.2 Mots-guide appliqués au diagramme des cas d'utilisation 52
2.4.3 Mots-guide appliqués au diagramme de séquence 53
2.4.4 Mots-guide appliqués au diagramme d'états-transitions 54
2.4.5 Documents produits par HAZOP-UML 55
2.5 Conclusion 57

3 Confiance dans un argumentaire de sécurité **61**
3.1 Introduction 61

3.2 Approches existantes pour l'évaluation de la confiance dans un argumentaire 63
3.2.1 Approches qualitatives . . . 63
3.2.2 Approche quantitative : probabilité Baconienne . . . 66
3.2.3 Approche quantitative : réseaux Bayésiens . . . 67
3.2.4 Approche quantitative : théorie de Dempster-Shafer . . . 70
3.3 Vers une nouvelle approche d'évaluation de la confiance d'un argumentaire . 74
3.3.1 Approche générale . . . 74
3.3.2 Mesure de confiance . . . 75
3.3.3 Types d'argument . . . 79
3.3.4 Argumentation simple . . . 80
3.3.5 Argumentation alternative . . . 83
3.3.6 Argumentation complémentaire . . . 86
3.3.7 Argumentation mixte . . . 90
3.3.8 Étude de sensibilité . . . 91
3.4 Conclusion . . . 93

4 Application à un robot d'aide à la déambulation 97
4.1 Présentation générale du projet MIRAS . . . 97
4.2 Vue générale du travail réalisé . . . 98
4.3 Les fonctionnalités du robot d'aide à la déambulation . . . 99
4.4 Analyse et évaluation des risques . . . 102
4.4.1 Application d'HAZOP-UML . . . 102
4.4.2 Validation de l'approche HAZOP-UML . . . 107
4.4.3 Critères de risque . . . 110
4.4.4 Estimation et évaluation du risque pour la version évaluation clinique 114
4.4.5 Estimation et évaluation du risque pour la version finale . . . 117
4.5 Argumentaire de sécurité du robot . . . 117
4.6 La confiance dans l'argumentaire de sécurité . . . 121
4.6.1 Construction du réseau de confiance . . . 121
4.6.2 Choix de l'outil AgenaRisk . . . 122
4.6.3 Étude de la sensibilité de la confiance . . . 124
4.7 Conclusion . . . 126

Synthèse, contributions majeures et perspectives 129

A Modèle GSN de MIRAS 133

Bibliographie 143

Introduction générale

La robotique et ses applications dans le domaine de la vie courante ont connu ces dernières années un développement exponentiel. Aujourd'hui, les systèmes robotiques ne sont plus réservés à une utilisation industrielle, nous les retrouvons de plus en plus utilisés dans différents domaines (santé, assistance aux personnes, loisir et confort, etc.) et surtout beaucoup plus présents dans la vie quotidienne : du simple robot aspirateur, au robot compagnon interagissant avec l'humain au domicile ou bien dans des espaces publics. Ces nouveaux types de robots, dits « de service », sont déployés dans des espaces partagés avec l'homme. Ils sont également de plus en plus conçus pour interagir physiquement avec l'homme et s'adapter à son environnement. Ces nouveaux contextes d'usage présentent de fortes contraintes du point de vue de la sécurité. De par la criticité de leur utilisation, les nouvelles générations de robots de service nécessitent des études de sécurité approfondies afin que l'on puisse placer une confiance justifiée dans leur capacité à fournir les services attendus sans engendrer de conséquences catastrophiques sur l'homme et son environnement.

La gestion des risques associés à ces nouveaux types de systèmes soulève de nouveaux défis pour la communauté robotique et la communauté de sûreté de fonctionnement. Deux questions fondamentales se posent aux concepteurs et aux utilisateurs de ces systèmes, ainsi qu'aux autorités et organismes de certification qui sont amenés à valider leur utilisation et commercialisation :

1. Comment identifier et évaluer les risques relatifs à la sécurité de ces systèmes ?
2. Comment construire un argumentaire rigoureux sur la capacité de ces systèmes à satisfaire leurs objectifs de sécurité et cerner les éléments de preuve permettant d'avoir confiance dans cet argumentaire ?

La construction d'argumentaire de sécurité (ou dossier de sécurité, ou *safety case*), est un des moyens permettant de préparer la certification de tels systèmes. Il s'agit principalement de justifier pour chaque danger comment il a été traité et ramené à un niveau

acceptable. Dans le cas des systèmes robotiques en interaction physique avec l'homme, de nombreuses incertitudes subsistent, et il n'existe pas à l'heure actuelle de méthode systématique et globale permettant la construction de tels argumentaires et la démonstration du niveau de confiance sous-jacent. L'objectif des travaux présentés dans ce manuscrit est de contribuer à la définition d'une telle méthode en partant d'une technique d'analyse du risque dédiée à l'analyse des interactions humain-robot, puis en s'appuyant sur des modèles formalisés permettant de construire l'argumentaire de sécurité et d'évaluer automatiquement le niveau de confiance dans cet argumentaire.

Ce manuscrit de thèse est structuré en quatre chapitres dont nous résumons dans la suite le contenu et les principales contributions.

Le Chapitre 1 présente la problématique liée à l'analyse des risques et la justification de la confiance dans le contexte des systèmes robotiques interagissant avec l'homme. Il présente brièvement les principaux concepts de la sûreté de fonctionnement et donne des exemples de travaux qui se situent à l'interface des domaines de la robotique et de la sûreté de fonctionnement. Cette analyse montre qu'il existe à notre connaissance peu de travaux de recherche dédiés à la prévision de fautes de systèmes robotiques en interaction avec l'homme et qui se sont intéressés à la problématique soulevée dans le cadre de nos travaux. La deuxième partie de ce chapitre introduit les principales techniques existantes d'analyse du risque et d'estimation de la confiance et résume les principales contributions de nos travaux.

Le Chapitre 2 présente la première contribution de nos travaux qui porte sur la définition d'une méthode d'analyse de risque HAZOP-UML qui est basée sur l'utilisation conjointe de la technique d'identification des dangers HAZOP et des modèles UML décrivant le système cible et ses interactions avec l'environnement. Cette méthode, initiée au LAAS-CNRS avant le début de cette thèse, a été enrichie et finalisée dans le cadre de nos travaux. Parmi les points importants de notre contribution, nous avons proposé une adaptation des mots-guides définis dans le cadre de la méthode HAZOP afin d'intégrer les diagrammes d'états-transitions de modèles UML dans un processus de gestion du risque.

Le Chapitre 3 présente la deuxième contribution de nos travaux qui porte sur la problématique de la construction d'argumentaire de sécurité et l'évaluation quantitative de la confiance que l'on peut lui accorder. Après une analyse détaillée de l'état de l'art sur ce sujet, nous proposons une nouvelle approche qui consiste à transformer un argumentaire construit avec la notation graphique GSN en un réseau de confiance qui est interprété comme un réseau bayésien pour calculer la confiance globale dans l'argumentaire et effectuer des études de sensibilité permettant d'identifier les éléments de preuve les plus influents. Cette approche permet de fournir des indicateurs à l'analyste pour améliorer ou

comparer différents argumentaires de sécurité, et éventuellement de prendre des décisions relatives à l'acceptation d'un argumentaire.

Le Chapitre 4 met en œuvre les deux contributions au sein d'un processus complet de gestion du risque dans le cadre du projet ANR MIRAS, pour le développement d'un robot d'aide à la déambulation. Ce robot permet à des patients de se lever, déambuler et de se rassoir. Il possède également des fonctions de détection de problèmes physiologiques. Nous utilisons les résultats de ce projet pour effectuer une validation de l'approche HAZOP-UML. Puis nous présentons comment les critères de risques ont été déterminés dans le projet MIRAS pour effectuer une estimation qualitative et une évaluation des risques pour une version du robot prévue pour les essais cliniques, et une autre pour une version finale. Pour cette deuxième version, le fait de ne pouvoir estimer les risques même qualitativement est pallié par la construction d'un argumentaire de sécurité. À partir de cet argumentaire nous montrons comment il est possible de construire un réseau de confiance et réaliser une étude de sensibilité.

Enfin, la conclusion présente un bilan des travaux réalisés dans cette thèse, ainsi qu'un rappel des contributions majeures. Nous identifions également un ensemble de perspectives de recherche pour l'amélioration et l'extension de notre approche.

Chapitre 1

Maîtrise de la confiance en robotique de service

1.1 Introduction

Après plusieurs décennies de recherche dans le domaine de la robotique, nous assistons aujourd'hui à une explosion du nombre d'applications hors des industries manufacturières, dans une robotique dite de «service [1]», c'est-à-dire effectuant des tâches non industrielles, liées à l'aide, au confort, ou la santé de l'humain (si on inclut les robots médicaux). Les applications étant très diverses, les conséquences de leur défaillance sont également très différentes. Ainsi, entre un robot aspirateur et un robot chirurgical, il est évident qu'une défaillance n'aura pas le même impact sur la sécurité-innocuité des utilisateurs. Parmi ces systèmes, certains sont en interaction physique avec l'humain, et parfois nommés robots collaboratifs, ou « cobots ». Ces systèmes posent de manière plus critique le problème de la confiance que l'on peut leur accorder. Cette question est d'autant plus complexe que de nombreuses caractéristiques de la robotique industrielle ont été modifiées. Le Tableau 1.1 présente les principales évolutions entre robotique industrielle et robotique de service (interactive), ainsi que quelques exemples des nouveaux dangers à considérer.

Les sources potentielles de ces dangers sont multiples mais peuvent être classées selon les catégories suivantes :

— Matérielle (défaillance des composants électroniques, par ex. les capteurs)
— Mécanique (défaillance d'une partie mécaniques, par ex. un axe)
— Logicielle (présence de fautes dans les programmes, par ex. une boucle infinie)

1. La norme ISO13482 (2014) propose la définition suivante : *Robot that performs useful tasks for humans or equipment excluding industrial automation applications*

	Robotique industrielle	**Robotique de service**	**Exemples de nouveaux dangers en Robotique de service**
Mouvements	Aucun mouvement en présence d'humains	Mouvements simultanés	Mauvaise synchronisation entre l'humain et le robot / mouvement non-interprétables par l'humain
Distance humain-robot	L'humain est « loin »	L'humain est proche/ en contact physique	Collisions, forces de contact trop importantes
Interaction humain-robot	Tableau de contrôle électronique	Interaction avancée (cognitive)	Confusion de modes / erreurs d'interprétation
Niveau de contrôle	Automatique	Autonome	Décisions du robot dangereuses
Architecture mécanique	Lourde / Rigide / Puissante	Légère / Souple / Limitée	Imprécisions mouvements/stockage énergie mécanique du à la souplesse
Complexité de la tâche	Mono-fonction	Multi-fonctions	Règles de sécurité complexes à vérifier
Structuration de l'environnement de travail	Structuré	Non-structuré	Situations adverses, incertitudes dans la perception

TABLE 1.1 – Nouveaux dangers de la robotique de service en interaction avec l'humain

— Humaine (erreur humaine, par ex. une collision avec le robot)
— Liées à l'environnement (condition d'exécution adverses, par ex. un éclairage inexistant)

Nous voyons à ce niveau qu'il est possible d'identifier des sources de dangers (liste ci-dessus), et leurs conséquences (Tableau 1.1). Dans le domaine de la sûreté de fonctionnement (incluant la sécurité, la fiabilité, etc.), il est proposé de décomposer la chaine cause-conséquence en utilisant les définitions suivantes (Avižienis et al., 2004; Laprie, 2004) :

— Défaillance : l'événement survenant lorsque le service délivré dévie de l'accomplissement de la fonction du système
— Erreur : partie de l'état du système qui est susceptible d'entraîner une défaillance
— Faute : la cause adjugée ou supposée d'une erreur

Ainsi, une faute, telle qu'un bogue logiciel (par ex. une boucle infinie), peut donner lieu à une erreur (le pointeur d'exécution arrive dans la boucle infinie), puis à une défaillance (le logiciel ne répond plus). Afin de traiter ces trois types de menaces, il est proposé de mettre en œuvre des techniques de sûreté de fonctionnement classées selon quatre catégories :

— Prévention des fautes : comment empêcher l'occurrence ou l'introduction de fautes (notamment par l'utilisation de« bonnes pratiques » de développement)

— Tolérance aux fautes : comment fournir un service à même de remplir la fonction du système en dépit des fautes (regroupe tous les mécanismes de détection et de recouvrement d'erreur)
— Élimination des fautes : comment réduire la présence (nombre, sévérité) des fautes (comme par exemple les techniques de test, ou de vérification formelle)
— Prévision des fautes : comment estimer la présence, l'occurrence future, et les possibles conséquences des fautes (en s'appuyant par exemple sur les méthodes d'analyse de risque présentées ultérieurement)

Il est important de noter que le terme de faute est utilisé ici de façon générique dans ces définitions, mais peut être substitué par erreur ou défaillance selon leur utilisation. Ainsi la plupart des mécanismes de tolérance aux fautes, permettent en réalité de traiter les erreurs (une faute a été activée) avant qu'elles ne se transforment en défaillances. Il est également possible d'utiliser les concepts d'évitement des fautes (englobant élimination et prévention) et d'acceptation des fautes (englobant prévision et tolérance).

L'utilisation de ces techniques s'est largement répandue dans le domaine de la robotique industrielle, mais en se limitant principalement aux techniques utilisées pour les machines, les robots étant considérés comme des machines industrielles au même titre que des fraiseuses par exemple. Cependant, le fait que les robots ne soient plus isolés derrière des barrières de protection et sont beaucoup plus complexes en termes de perception, décision et réaction, ne permettent pas d'utiliser les techniques usuelles de la robotique industrielle. Ainsi, de nouveaux thèmes de recherche ont émergé, à la croisée des domaines de la sûreté de fonctionnement et de la robotique. On présente ci-dessous des exemples de travaux en utilisant la classification de la sûreté de fonctionnement :

— Prévention des fautes
  — La prévention des fautes logicielles dans les contrôleurs de robots est notamment couverte par l'utilisation de techniques issues du génie logiciel telle que l'utilisation de *framework* ou de *middleware* pour le développement des contrôleurs de robot (comme Genom [2] au LAAS, ou ROS [3])
  — La prévention d'erreurs humaines dues à une mauvaise synchronisation entre l'homme et le robot, notamment dans les travaux concernés par la planification de mouvements éligibles par l'homme (c.à.d. des mouvements des parties mobiles du robots tels qu'ils paraissent naturels pour l'homme, et donc qu'il peut anticiper) (Mainprice et al., 2010).
  — La prévention des collisions par des algorithmes réactifs d'évitement de l'humain (Haddadin, 2014)

2. https ://www.openrobots.org/wiki/genom
3. http ://www.ros.org/

  - La prévention de fautes par l'introduction de limites intrinsèques de performance en termes de poids, de force, de vitesse comme dans le bras LWR développé par le DLR [4] et fabriqué par KUKA [5] ou celui d'Universal Robots [6]. Il existe également des limites physiques aux déplacements possibles et aux degrés de liberté des robots, notamment en robotique médicale (Glauser et al., 1993)
- Élimination des fautes
  - L'élimination des fautes logicielles par le test est rarement abordée mais on peut citer des travaux qui s'intéressent à la génération d'environnements (carte, obstacles, etc.) pour effectuer des tests de navigation de robots (Arnold et Alexander, 2013).
  - L'élimination des fautes logicielles par l'utilisation de techniques issues de la vérification formelle (Bensalem et al., 2011)
- Tolérance aux fautes
  - La souplesse (ou *compliance* en anglais) des mouvements du robot (actionneurs souples ou à rigidité variable) (Albu-Schaffer et al., 2008; Filippini et al., 2008)
  - Systèmes de sécurité indépendants pour les systèmes autonomes vérifiant des règles de sécurité en-ligne (Machin et al., 2014; Roger et al., 2012; Mekki-Mokhtar et al., 2012)
  - La tolérance aux fautes des couches fonctionnelles (Durand, 2011), décisionnelle (Lussier et al., 2007) ou pour la perception (Bader et al., 2014), d'une architecture robotique
  - La détection et réaction à la présence de l'humain (Détection de collision et réaction) (Haddadin et al., 2011)

Comme on peut le noter dans la liste précédente, très peu de travaux ont porté sur la prévision des fautes. Il existe en effet quelques travaux, comme ceux du DLR (Haddadin, 2014) sur l'estimation des conséquences d'une collision, ou ceux du LAAS (Martin-Guillerez et al., 2010b; Guiochet et al., 2010; Martin-Guillerez et al., 2010c; Do Hoang et al., 2012; Guiochet et al., 2013) sur les techniques d'analyse du risque pour la robotique collaborative, mais c'est un domaine de recherche peu exploré. Cela provient notamment du fait que cet aspect est généralement couvert par les normes, qui sont spécifiques aux domaines. Cependant, aujourd'hui, du fait des spécificités de la robotique de service, les directives ou les normes machines utilisées précédemment (2006/42/EC, 2006; ISO13849-1, 2006) ne sont pas entièrement applicables, en particulier pour l'aspect contact physique en continu entre l'homme et le robot. Les normes génériques comme IEC61508-5 (2010), sont égale-

4. http ://www.dlr.de/rm/en/desktopdefault.aspx/tabid-3803/
5. http ://www.kuka-labs.com/en/medical_robotics/lightweight_robotics/
6. http ://www.universal-robots.com/GB/Products.aspx

ment difficilement applicables du fait des incertitudes des réactions du robot face à des situations différentes (on peut noter que dans cette norme l'utilisation de techniques issues de l'intelligence artificielle est non recommandée pour les systèmes les plus critiques).

Plus récemment, la norme robotique ISO10218-1 (2011) portant uniquement sur les robots en milieu industriel a été adaptée au robots « personnels » dans la toute nouvelle norme ISO13482 (2014). Il est cependant difficile d'en estimer aujourd'hui l'impact dans l'industrie, tant les applications robotiques sont variées. En effet, pour un robot certifié de manière générique, son utilisation dans un environnement réel peut tout changer. C'est sans doute la raison pour laquelle il n'existe que très peu de robots certifiés. On peut citer celui de Universal Robots, dont il est mentionné dans la spécification commerciale [7] que toutes les fonctions de sécurité ont été certifiées par le TÜV (Technischer Überwachungs-Verein – une organisation allemande qui travaille pour valider la sécurité des produits) et « testées conformément à la norme EN ISO 13849 : 2008 PL d et EN ISO 10218-1 : 2011, clause 5.4.3. ». Il est important de bien voir que cette certification concerne uniquement la présence d'un dispositif de sécurité (clause 5.4.3), ayant un certain niveau de performance (PL pour *Performance Level*). Un PL d, équivaut à un niveau d'intégrité SIL2 de la norme IEC61508-5 (2010). Cependant, cela ne garantit pas que pour une utilisation particulière du robot, les utilisateurs ne soient pas confrontés à des risques inacceptables. On peut s'attendre à retrouver la même limitation avec la norme ISO13482 (2014) sur les robots personnels. Il est également important de noter que cette norme ne couvre pas les robots médicaux, qui en tant que dispositifs médicaux, sont couverts par la directive Européenne 93/42/EEC (1993), et par l'ensemble des normes ISO du domaine médical comme la norme ISO/FDIS14971 (2006) qui décrit l'activité de gestion du risque.

Il existe donc un déficit en recherche, qui concerne la mise en œuvre des principes énoncés dans les normes, autour de la gestion du risque, pour les applications de la robotique de service. Le cadre conceptuel existe, mais l'application des méthodes d'analyse du risque (ou de prévision des fautes) est aujourd'hui une véritable problématique qu'il est important de traiter. La difficulté d'appliquer les normes a également pour conséquence directe un manque de méthodes systématiques pour la certification en robotique de service. Ces deux défis interconnectés sont abordés dans ce manuscrit grâce à la combinaison et à l'adaptation d'un ensemble de techniques existantes dans le domaine de la sûreté de fonctionnement. La suite de ce chapitre présente ces techniques, et conclut sur une vue générale de l'approche proposée, permettant d'analyser les risques d'un système de robotique de service, et de construire un argumentaire de sécurité.

7. http ://media1.limitless.dk/UR_Tech_Spec/UR5_EN.pdf, consulté le 7.01.2015

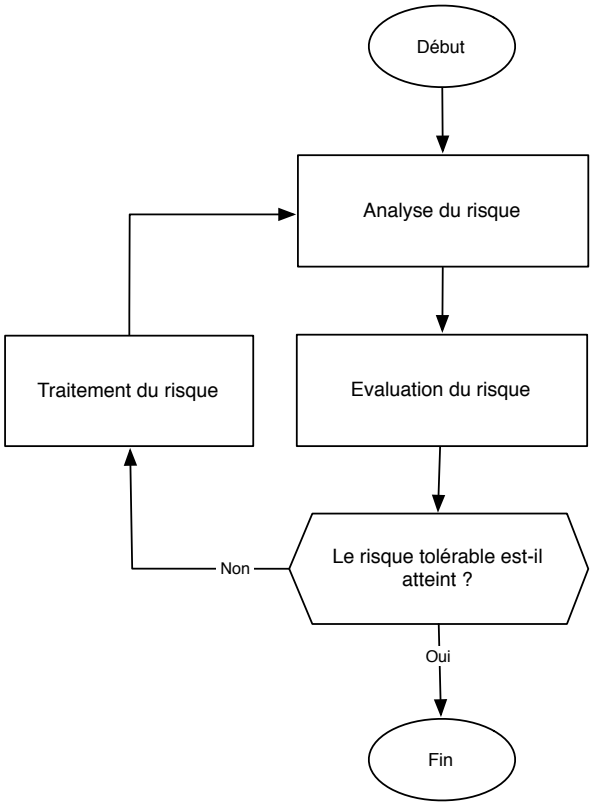


FIGURE 1.1 – Représentation schématique du processus de gestion des risques

1.2 Gestion du risque

La gestion du risque est aujourd'hui une activité normalisée dans de nombreux domaines, et déclinée dans des directives européennes et des normes ISO telle que la norme générique ISO/DIS31000 (2009) dont on retrouve les concepts dans la norme robotique ISO13482 (2014). Dans le domaine de la sécurité (au sens *safety*), le concept de base est le dommage, désignant toute blessure physique ou atteinte à la santé des personnes, ou atteinte aux biens tels que les systèmes robotisés eux mêmes, ou à l'environnement. À partir d'un dommage, est définie la notion de risque comme étant la combinaison de la probabilité de ce dommage et de sa gravité. Les principales activités de la gestion du risque sont représentées Figure 1.1, et sont définis par (ISO/IEC-Guide73, 2009) :

— L'analyse du risque met en œuvre toute technique permettant d'identifier et de comprendre la nature d'un risque, et d'en déterminer son niveau d'importance.

Terme habituel	Description possible
Significatif	Décès ou perte de fonction ou de structure
Modéré	Blessure réversible ou légère
Négligeable	Ne provoquera pas de blessure ou une blessure bénigne

FIGURE 1.2 – Exemples de niveaux de gravité qualitatifs issus de la norme médicale ISO/FDIS14971 (2006)

Terme habituel	Description possible
Élevé	Devrait se présenter souvent, fréquemment
Moyen	Peut se présenter, mais pas fréquemment
Faible	Ne devrait pas se présenter, rare, isolé

FIGURE 1.3 – Exemples simplifiés de niveaux de probabilité d'occurrence qualitatifs définis dans la norme médicale ISO/FDIS14971 (2006)

— L'évaluation du risque correspond au processus de comparaison des résultats de l'analyse du risque avec les critères de risque afin de déterminer si le risque et/ou son importance sont acceptables ou tolérables.
— Le traitement du risque qui est un processus destiné à modifier un risque, principalement par l'élimination des sources de risque, ou la réduction des conséquences et de leurs probabilités d'occurrence.

Ce découpage classique de la gestion du risque a été révisé dans la norme ISO/DIS31000 (2009), qui a extrait de l'étape d'analyse du risque, une étape amont pour l'identification du risque. Il existe également dans cette norme de nombreuses autres activités autour du risque que nous ne détaillerons pas dans ce manuscrit afin de nous concentrer sur les étapes importantes pour cette thèse.

L'activité centrale de la gestion du risque est l'analyse du risque, et notamment l'estimation des niveaux de risque. Elle repose sur l'utilisation de critères de risque, c-à-d. de tables de mesures de niveaux de probabilité d'occurrence d'un dommage et de sa gravité. Les Figures 1.2 et 1.3 donnent deux exemples simples de mesures de niveaux de gravité et de probabilité d'occurrence selon la norme médicale ISO/FDIS14971 (2006). Le niveau d'importance d'un risque est donc déterminé par l'évaluation des niveaux de gravité et d'occurrence du dommage associé selon ces tables. L'activité d'analyse du risque consiste alors à identifier les prémisses d'un dommage, définis par les concepts de situation dangereuse et de phénomène dangereux, que l'on regroupera sous le concept de danger (*hazard*). La Figure 1.4 illustre l'utilisation de ces concepts dans le cas d'un calcul d'un niveau de risque

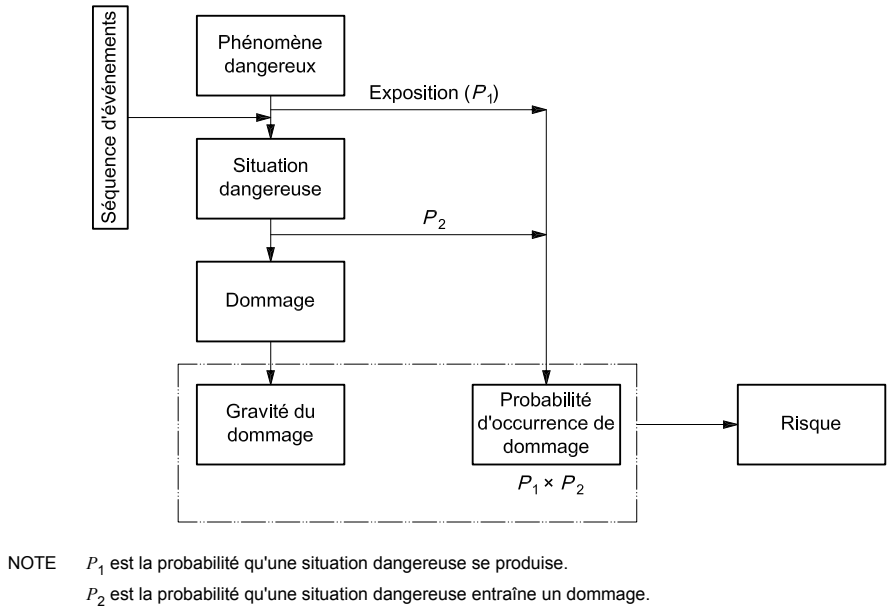


FIGURE 1.4 – Calcul de niveau de risque en fonction de la probabilité et de la gravité, extrait de ISO/FDIS14971 (2006)

en prenant également en compte la probabilité d'exposition à un phénomène dangereux. Cette estimation peut être qualitative en utilisant les tables précédentes, mais également quantitative comme présenté ultérieurement dans cette section. À la suite de cette estimation, les risques sont placés dans des matrices, selon leur niveau. À titre d'exemple, la matrice de la Figure 1.5, présente les neufs paires $(probabilité, gravité)$ possibles à partir des tables précédentes (Figures 1.2 et 1.3), ainsi que six risques (R1 à R6) identifiés pour un système donné. Cette matrice présente également un critère important, qui est le niveau d'acceptabilité des risques, qui est dans ce cas binaire (acceptable/inacceptable). On retrouve dans la littérature et les normes de nombreuses variantes plus complexes de ce type de matrice. La tendance principale est l'utilisation de 3 niveaux de risques :

— Acceptable
— Tolérable
— Intolérable

Cette échelle est reprise notamment dans le principe ALARP (*As Low As Reasonably Practicable*) présenté Figure 1.6, qui pour chaque niveau de risque fait le lien avec les possibilités de traitement du risque (ici limité à la réduction) en fonction des coûts engendrés par les traitements.

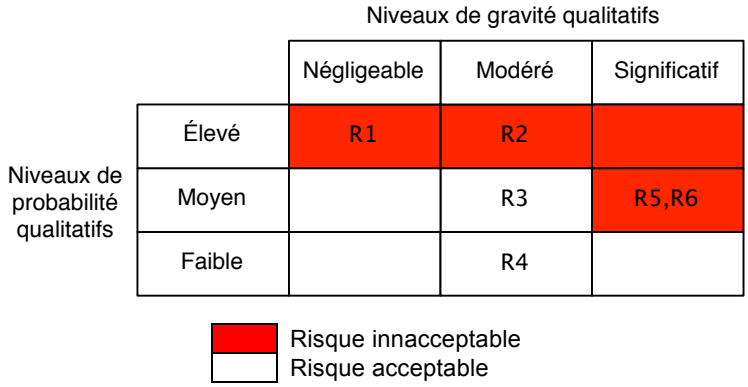


FIGURE 1.5 – Exemple de matrice 3x3 de risques

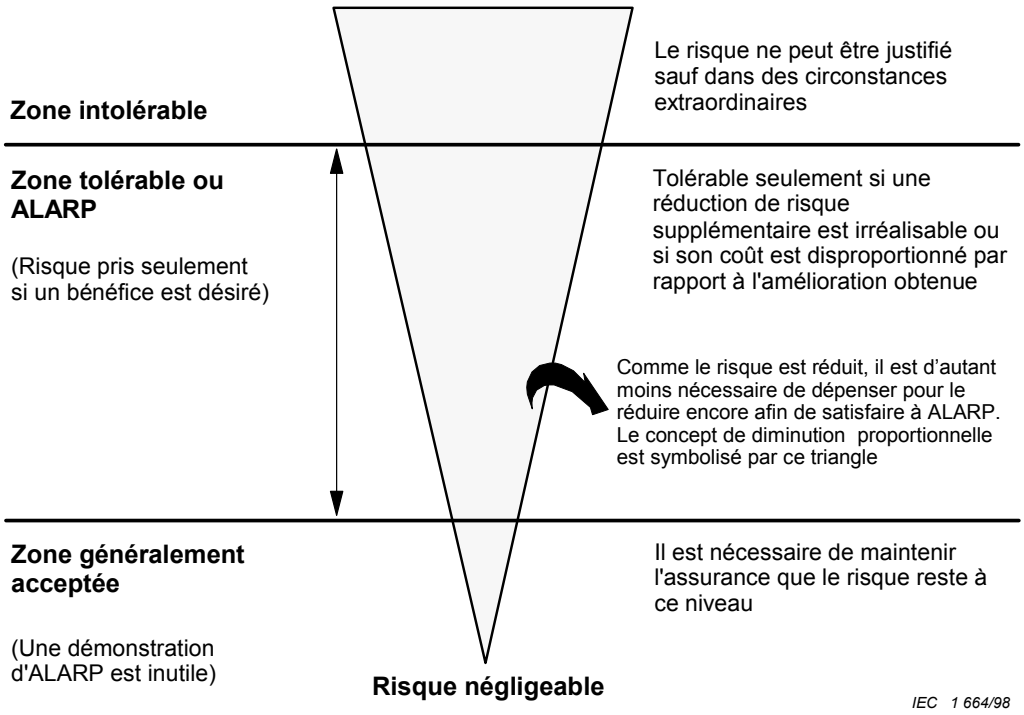


FIGURE 1.6 – Risque tolérable et ALARP, extrait de IEC61508-5 (2010)

La confiance dans l'estimation d'un risque est accrue lorsqu'une estimation quantitative de la probabilité peut être faite sur la base de données précises et fiables ou lorsqu'une estimation qualitative raisonnable est possible. Cela n'est cependant pas toujours possible. Des exemples de cas où il est très difficile d'estimer les probabilités sont :

— la défaillance de logiciels critiques ;
— les situations particulières liées à l'humain telles qu'une erreur humaine, un sabotage ou une falsification ;
— l'apparition de nouveaux phénomènes dangereux et de situations inattendues ;

Le cas de systèmes innovants, comportant des interactions complexes avec l'humain, et incluant du logiciel, comme dans le cas de la robotique de service, pose donc de nombreuses incertitudes quant à l'estimation quantitative et même qualitative des probabilités d'occurrence. L'approche généralement utilisée en l'absence de données relatives à la probabilité d'occurrence d'un dommage, est de réaliser l'estimation du risque sur la base d'une estimation de probabilité raisonnable du pire des cas. Dans certains cas, il est même recommandé de définir cette valeur par défaut de la probabilité sur un et de baser les mesures de maîtrise du risque sur la prévention du phénomène dangereux dans son ensemble. Cette approche présente l'inconvénient majeur de normaliser tous les risques, et d'obtenir une liste trop importante de risques inacceptables car tous jugés comme étant trop fréquents.

L'étape d'identification des prémisses des risques et de calcul de leurs niveaux (analyse du risque), est en général réalisée en utilisant des techniques d'analyse du risque, dont les plus utilisées sont certainement aujourd'hui l'APR (Analyse Préliminaire des Risques), l'AMDEC (Analyse de Modes de Défaillances, de leurs Effets, et de leur Criticité), l'analyse des AdF (Arbres de Fautes), et l'HAZOP (*Hazard Operational*), toutes décrites dans les paragraphes suivants ainsi que dans la section suivante pour la méthode HAZOP.

L'APR est une méthode d'analyse essentiellement inductive (raisonnement de la cause à l'effet, aussi appelée *bottom-up*) dont l'objectif est d'identifier les phénomènes dangereux, les situations dangereuses et les événements susceptibles de provoquer un dommage relatif à une activité donnée, à une installation ou à un système. Elle est plus généralement utilisée au début d'un processus de développement, lorsqu'on dispose de peu d'informations relatives à la conception ou aux modes opératoires de fonctionnement. Elle consiste lors de séances de remue-méninges (*Brainstorming*) à lister dans des tableaux les phénomènes dangereux notamment en utilisant des catégories comme celles présentées dans la norme machine ISO12100 (2010), reprises dans la norme robotique industrielle ISO10218-1 (2011), puis adaptées dans la norme ISO13482 (2014). Un extrait de tableau générique reprenant les dangers listés dans les normes précédentes est donné dans la Figure 1.7.

La technique AMDEC permet d'identifier et d'évaluer systématiquement les conséquences d'un mode de défaillance d'un composant ou d'une fonction d'un composant (il

Type d'origine du danger	Origine du danger	Dommage potentiel	Gravité	Recommendations
Mécanique	Le robot agrippe les doigts de l'utilisateur avec sa pince	Pincement de la peau, ou d'un muscle ou d'une articulation	Modéré	La force de serrage de la pince doit limitée en cas de collaboration humain-robot
Mécanique	Le robot lance un objet qu'il tient dans sa pince en direction de l'opérateur	Impact avec une partie vitale de l'opérateur (tête)	Significatif	Le couple vitesse maximum du bras robot et poids maximum de l'objet doivent être dimensionnés pour réduire la gravité de l'impact
Thermique	Les articulations du robot que l'opérateur peut toucher sont en surchauffe	L'opérateur se brûle lorsqu'il souhaite arrêter le robot en le touchant	Modéré	Une étude des température de surchauffe doit être effectuée
Nuisance sonore	Le robot dépose les objets en tapant les supports avec un niveau sonore trop élevé (1 dépose toutes les minutes)	Inconfort, stress de l'opérateur	Modéré	La table et les objets peuvent être équipés d'amortisseurs sonores

FIGURE 1.7 – Exemple de tableau d'APR pour l'utilisation d'un robot en interaction avec l'homme

existe également des AMDEC processus). Il s'agit également d'une technique inductive qui, partant d'un ensemble de composants et de leurs modes de défaillance, liste les conséquences possibles dans des tableaux dont un exemple est donné Figure 1.8. Les composants sont analysés individuellement, considérant ainsi, en général, une condition de premier défaut. Les composants peuvent être mécaniques, électroniques, logiciels, ou même humains. La principale difficulté est alors de relier entre elles les tables AMDEC de ces différents domaines pour établir les liens causes-conséquences, jusqu'à la conséquence finale : le dommage.

La technique d'analyse des Arbres de Fautes (AdF) est une méthode d'analyse des phénomènes dangereux identifiés par d'autres techniques comme l'AMDEC ou l'APR ; elle a pour point de départ une conséquence postulée non désirée, également dénommée événement principal. Elle procède par déduction (raisonnement de l'effet à la cause ou *top-down*) en partant de l'événement principal. Les causes possibles entraînant la conséquence non souhaitée sont identifiées. Une identification par étapes du fonctionnement non désiré d'un système vers des niveaux de système de plus en plus bas mènera au niveau de système désiré, qui est habituellement une faute d'un composant, auquel des mesures de maîtrise du risque peuvent être appliquées. Un arbre de fautes révèle les séquences les mieux à même de mener à l'événement principal. Les résultats sont représentés sous forme d'un

Composant	**Modes de défaillance**	**Cause**	**A. Effet local B. Effet sur le système**	**Estimation du risque**			**A. Moyens de détection possibles B. Solutions**
				Occurrence	Gravité	Risque	
Processeur du contrôleur de robot	Figé	Interblocage du programme ou du système d'exploitation	A. Envoie commande constante B. Mouvement bloqué en un point	Fréquent	Mineur	Faible	A. Système externe type watchdog B. Réinitialisation et Alerte utilisateur

FIGURE 1.8 – Exemple de tableau pour une AMDEC

arbre des modes de défaillance. À chaque niveau de l'arbre, des combinaisons des modes de défaillance sont décrites avec des opérateurs logiques (ET, OU, etc.) comme cela est présenté dans la Figure 1.9. Les modes de défaillance identifiés dans l'arbre peuvent être associés à des fautes matérielles, logicielles, mécaniques, ou à des erreurs humaines ou à tout autre événement pertinent ayant mené à un événement non désiré. À partir des arbres, deux études sont généralement menées : la recherche des combinaisons minimales menant à l'événement principal redouté, et la réduction de leur probabilité d'occurrence (en introduisant, par exemple, de la redondance). Toutes ces techniques ont été utilisées dans l'industrie de robotique industrielle, et expérimentées dans les applications de robotique de service. Cependant leur utilisation repose souvent sur une bonne connaissance du système et de son contexte d'utilisation (notamment, des humains pouvant effectuer des actions diverses), et sur des expertises s'appuyant sur des années de développement et d'utilisation de systèmes similaires. Dans le cadre de la robotique de service, le nombre d'incertitudes liées aux interactions humain-robot, au logiciel, etc., ainsi que le caractère innovant, font que l'utilisation de ces techniques reste encore marginale.

1.3 Technique d'analyse du risque HAZOP

La méthode HAZOP, pour *HAZard OPerability*, a été développée par la société Imperial Chemical Industries (ICI) au début des années 1970. Elle a depuis été adaptée dans différents secteurs d'activité. Considérant de manière systématique les dérives des paramètres d'une installation en vue d'en identifier les causes et les conséquences, cette méthode est utilisée pour l'examen de systèmes thermo-hydrauliques pour lesquels des paramètres comme le débit, la température, la pression, le niveau, la concentration sont particulièrement im-

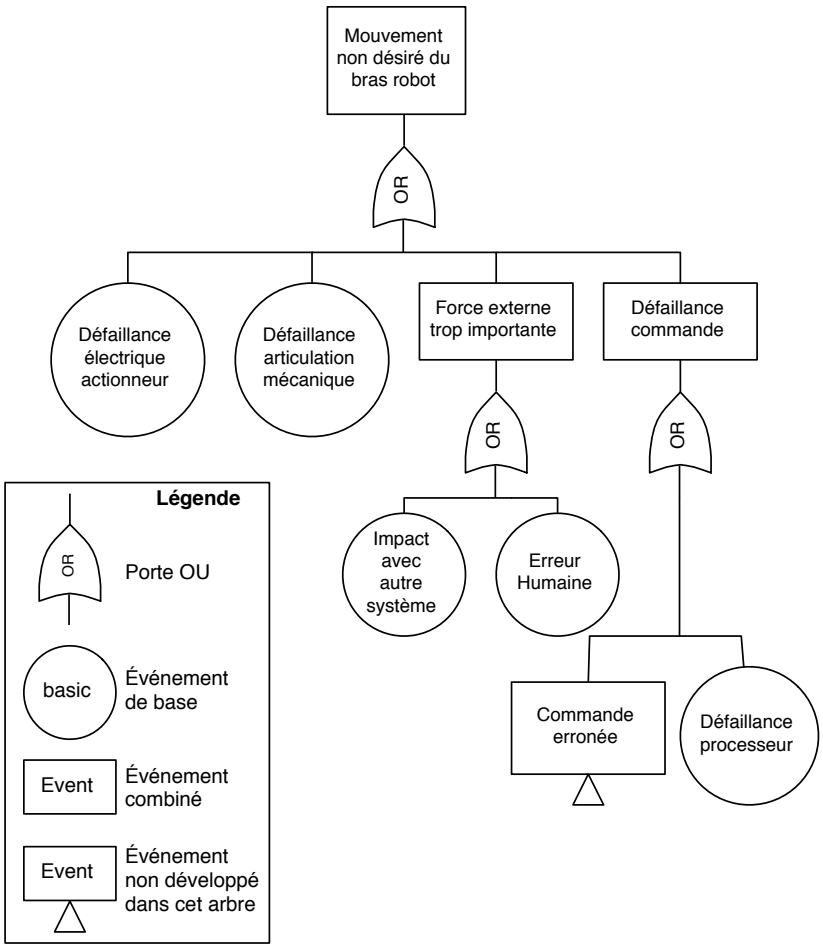


FIGURE 1.9 – Exemple d'arbre de défaillance pour un bras robot

portants pour la sécurité de l'installation. Elle a fait l'objet d'une norme IEC61882 (2001) et est largement utilisée aujourd'hui. Son succès tient à sa simplicité et à la possibilité de réaliser des HAZOP très tôt dans le processus de développement. Elle a également comme avantage d'être adaptable au formalisme utilisé pour décrire un système.

HAZOP ne considère pas des modes de défaillances comme l'AMDEC mais les dérives potentielles (ou déviations) des principaux paramètres liés à l'exploitation de l'installation. Pour chaque partie constitutive du système examiné, la génération (conceptuelle) des déviations est effectuée de manière systématique par la conjonction :

— de mots-guides comme par exemple `PAS DE` ; `PLUS DE` ; `MOINS DE` ; `TROP DE`

— des paramètres associés au système étudié comme par exemple dans le cas d'un procédé industriel : la `température`, la `pression`, le `débit`, etc. mais également `le temps` ou des `opérations` à effectuer.

Les mots-guides, accolés aux paramètres importants pour le procédé, permettent de générer de manière systématique les dérives à considérer. La combinaison de ces paramètres avec les mots-guides précédemment définis permet donc de générer des dérives de ces paramètres comme par exemple :

— `Plus de` et `Température` = `Température trop haute`

— `Moins de` et `Pression` = `Pression trop basse`

— `Inverse` et `Débit` = `Retour de produit`

— `Pas de` et `Niveau` = `Capacité vide`

Dans ce manuscrit, nous nous appuierons sur la version anglaise de ces mots clés donnée dans la Table 1.2. En effet, les tables qui ont été réalisées ont servi de base dans différents projets Européens, et tout le travail repose sur le sens de ces mots. Il ne nous semble donc pas judicieux de présenter des résultats et des conclusions sur l'utilisation des mots guides en les traduisant en français.

Les paramètres auxquels sont accolés les mots-guides dépendent bien sûr du système considéré. Généralement, l'ensemble des paramètres pouvant avoir une incidence sur la sécurité de l'installation doit être sélectionné. Sur l'exemple précédent de procédé industriel, les paramètres sur lesquels porte l'analyse sont : le débit, la température, la pression, le niveau, le temps. Pour d'autres systèmes, les paramètres sont plus complexes à déterminer. Ainsi pour un robot, les paramètres pourraient être la vitesse du robot, la force exercée ou l'accélération, mais cette identification n'est pas faite de manière structurée. L'approche utilisée dans plusieurs domaines (également dans la norme IEC61882 (2001)), est de définir pour un système deux niveaux de description : le système est composé d'entités physiques ou logiques, chaque entité étant composée d'attributs permettant d'identifier ses caractéristiques essentielles. L'application des mots-guide revient donc à exécuter le processus

Guideword	**Interpretation**
No/None	Complete negation of the design intention No part of the intention is achieved and nothing else happens
More	Quantitative increase
Less	Quantitative decrease
As Well As	All the design intention is achieved together with additions
Part of	Only some of the design intention is achieved
Reverse	The logical opposite of the design intention is achieved
Other than	Complete substitution, where no part of the original intention is achieved but something quite different happens
Early	Something happens earlier than expected relative to clock time
Late	Something happens later than expected relative to clock time
Before	Something happens before it is expected, relating to order or sequence
After	After Something happens after it is expected, relating to order or sequence

TABLE 1.2 – Liste des mots-guides pour HAZOP extrait de IEC61882 (2001)

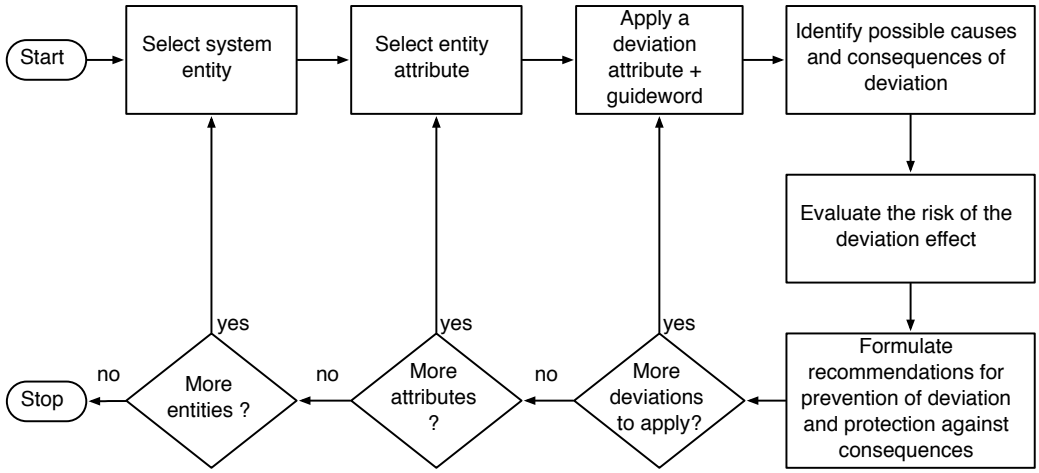


FIGURE 1.10 – Processus HAZOP

décrit dans la Figure 1.10, où chaque attribut de chaque entité se voit appliquer chaque mot-guide pour identifier les déviations possibles.

Une fois la déviation envisagée, l'analyste doit identifier les causes de cette déviation, puis les conséquences potentielles de cette déviation. Ainsi sont prévus les moyens destinés à sa détection et les barrières de sécurité pour en réduire l'occurrence ou les effets. La Figure 1.11 présente un exemple de table HAZOP utilisée lors de cette analyse, mais de nombreuses variantes existent à partir de cette base.

1.4 Analyse quantitative : les réseaux bayésiens

La plupart des techniques d'analyse quantitative se basent aujourd'hui sur la théorie des probabilités pour traiter les phénomènes comme les défaillances, caractérisés par l'aléatoire et l'incertitude. Les probabilités bayésiennes constituent un exemple de techniques permettant de prendre en compte de telles incertitudes. Cette théorie est également utilisée pour estimer des probabilités d'évènements induits ou conditionnés par l'occurrence d'autres événements ou de mettre à jour ces estimations suite à l'observation d'informations complémentaires. Cette branche de la théorie des probabilités s'appuie sur l'étude des probabilités conditionnelles : la probabilité conditionnelle d'un événement A, sachant qu'un autre événement B de probabilité non nulle s'est réalisé (ou probabilité de A, sachant B) est notée $P(A|B)$.

Le calcul des probabilités conditionnelles peut s'appuyer sur les réseaux bayésiens. Ils sont en premier lieu schématisés par un graphe orienté acyclique, représentant les liens de

Study title:							Page: of		
Drawing no.:		Rev no.:					Date:		
HAZOP team:							Meeting date:		
Part considered:									
Design intent:		Material: Source:		Activity: Destination:					
No.	Guide-word	Element	Deviation	Possible causes	Conse-quences	Safeguards	Comments	Actions required	Action allocated to

– Source: IEC 61882

FIGURE 1.11 – Exemple de table HAZOP extrait de IEC61882 (2001)

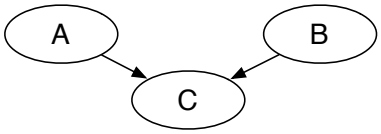


FIGURE 1.12 – Modèle étudié : 2 nœuds parents (A, B) et 1 nœud fils (C)

causalité entre des variables aléatoires comme cela est représenté dans la Figure 1.12. Dans cet exemple il existe un lien de causalité entre A, B et C. Prenons l'exemple suivant :

— A : il pleut
— B : il y a une grève de train
— C : Bob est en retard

Nous considérons que l'analyse amont a montré qu'il existait un lien de causalité entre le fait qu'il pleuve ou qu'il y ait une grève, et le fait que Bob soit en retard. Dans cet exemple, ainsi que dans les calculs suivants nous choisirons des valeurs Booléennes pour les variables. Ainsi le nœud A peut avoir les valeurs Vrai (noté A), ou Faux (noté $\overline{A}$). Par souci de clarté, on notera $P(A)$ pour $P(A = Vrai)$ et $P(\overline{A})$ pour $P(A = Faux)$. Les réseaux bayésiens permettent également de calculer des probabilités avec des variables ayant plusieurs valeurs possibles (par exemple un nœud pourrait correspondre à « Bob

arrive au travail » avec trois valeurs possibles : *en avance, à l'heure, en retard*), mais nous n'utiliserons ici que des variables booléeennes.

L'étape suivante consiste à attribuer à chaque nœud des tables des probabilités (TdP). Les valeurs d'une TdP sont souvent fixées manuellement mais comme nous le verrons ultérieurement dans cette section, il existe des outils permettant de réduire l'effort de détermination de ces tables. La saisie de la table de probabilité des états pour chacun des nœuds peut se faire donc :

— manuellement
— par calcul arithmétique ou avec une expression
— par calcul avec une loi de distribution

À titre d'exemple, plaçons nous dans le cas où l'on observe que C ne peut être VRAI que si et seulement si au moins l'un des parents est VRAI (l'équivalent d'un OU logique). On obtient alors une TdP pour chaque nœud comme présenté au tableau 1.3. On note que dans cette table par exemple $P(C|A,B) = 1$.

A	
$P(A)$	0,8
$P(\overline{A})$	0,2

B	
$P(B)$	0,7
$P(\overline{B})$	0,3

C				
A	0		1	
B	0	1	0	1
$P(C)$	0	1	1	1
$P(\overline{C})$	1	0	0	0

TABLE 1.3 – Exemple de tables de probabilité des nœuds, renseignées manuellement

À partir de cette étape, il est possible de réaliser principalement les 3 calculs suivants :

1. calcul de la probabilité totale définie par : si $(Y_i)_{i\in I}$ est un système exhaustif (fini ou dénombrable) d'événements, avec I l'ensemble des variables, et si quel que soit $i \in I, P(Y_i \neq 0)$ alors, pour tout événement X :

$$P(X) = \sum_{i\in I} P(X|Y_i)P(Y_i) \tag{1.1}$$

Dans notre exemple on suppose que A et B sont indépendants, c.à.d. $P(A \cap B) = P(A)P(B)$. En appliquant la formule de probabilité totale sur C on obtient :

$$\begin{aligned} P(C) = {} & P(C|A \cap B)P(A)P(B) + P(C|A \cap \overline{B})P(A)P(\overline{B}) \\ & + P(C|\overline{A} \cap B)P(\overline{A})P(B) + P(C|\overline{A} \cap \overline{B})P(\overline{A})P(\overline{B}) \end{aligned} \tag{1.2}$$

$$P(C) = 1 \times 0,8 \times 0,7 + 1 \times 0,8 \times 0,3 + 1 \times 0,2 \times 0,7 + 0 \times 0,2 \times 0,3 = 0,94$$

Donc $P(C) = 0,94$ et $P(\overline{C}) = 1 \times 0,2 \times 0,3 = 0,06$. Ce calcul s'interprète de la manière suivante : si l'on a estimé que la probabilité qu'il pleuve est de 0,8, et que celle qu'il y ait grève est de 0,7, alors la probabilité que Bob soit en retard est de 0,94 !

2. calcul de la probabilité d'un nœud enfant, sachant un nœud parent (par exemple, A et B sont des nœuds «parents» de C). Cela revient à fixer la valeur d'un nœud (noté observation), par exemple $P(A) = 1$, puis faire le calcul de la probabilité totale tel que défini ci-dessus ; Par exemple, si on cherche la valeur de $P(C|A)$, c.à.d., on a observé A (donc $P(A) = 1$), et l'on souhaite calculer la probabilité de C. La TdP de A est donc mise à jour, et on effectue le calcul de probabilité totale :

$$P(C|A) = 1 \times 1 \times 0,7 + 1 \times 1 \times 0,3 + 1 \times 0 \times 0,7 + 0 \times 0 \times 0,3 = 1$$

3. calcul de la probabilité d'un nœud parent, sachant un nœud enfant (par exemple, C est un nœud « enfant » de A et B). Pour cela il faut utiliser le théorème de Bayes, soit étant donnés 2 événements X et Y, la probabilité de X *sachant* Y est :

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} = \frac{P(Y|X)P(X)}{P(Y|X)P(X) + P(Y|\overline{X})P(\overline{X})} \tag{1.3}$$

Par exemple,

$$P(A|C) = \frac{P(C|A)P(A)}{P(C)}$$

avec $P(C|A) = 1$ (calcul précédent) ; $P(A) = 0,8$ (TdP) ; $P(C) = 0,94$ (calcul précédent).
Alors :

$$P(A|C) = \frac{1 \times 0,8}{0,94} = 0,85106$$

Une des principales difficultés dans l'utilisation des réseaux Bayésiens, est d'identifier les valeurs dans les TdP, et plus particulièrement dans celle des nœuds enfants. En effet, pour un nœud avec n parents, chaque nœud ayant k valeurs possibles (dans notre exemple k=2, c.à.d. VRAI et FAUX), la TdP du nœud enfant contient k^n valeurs, qu'il convient de déterminer ! Pour simplifier ces identifications, il existe plusieurs travaux, dont ceux de Fenton et al. (2007), qui consistent à décrire des fonctions permettant de calculer les valeurs d'une TdP d'un nœud enfant. Ses travaux sont intégrés dans l'outil AgenaRisk [8], qui permet d'accéder à un certain nombre de fonctions, allant de simples opérateurs comme

8. www.agenarisk.com

OR, AND, XOR, à des fonctions plus complexes comme des minimums avec des poids sur chaque nœud parent.

Prenons comme exemple le cas précédent où il a été déterminé que les 2 causes A et B se combinaient en un OR pour induire C. En partant de la porte logique booléenne OR, on peut en déduire la table de probabilité pour le nœud C de la Table 1.3, ainsi que la valeur de la probabilité totale $P(C)$ en utilisant la formule 1.2. En simplifiant cette expression, on retrouve bien la formule classique de la probabilité de l'union de deux événements indépendants : $P(C) = P(A) + P(B) - P(A)P(B)$

Díez et Druzdzel (2007) présentent également l'utilisation du *Noisy-OR*, introduit par Pearl (1988), qui correspond au tableau 1.4. Dans ce cas, lorsque l'on observe B uniquement, la probabilité n'est pas de 1, mais d'une valeur q (en général proche de 1). L'expression « *noisy* » indique que les relations entre les variables parents et la variable enfant ne sont pas nécessairement déterministes. En d'autres termes, chaque parent peut produire un effet avec une certaine probabilité. La valeur dans le cas où l'on observe A et B est alors de $1 - (1 - p)(1 - q) = p + q - p.q$, qui correspond bien à un OR entre les deux cas. Dans ce cas, si aucun des 2 événements n'est vérifié, alors la probabilité est nulle.

A	0		1	
B	0	1	0	1
$P(C)$	0	q	p	1-(1-p)(1-q)

$$P(C) = p.P(A) + q.P(B) - p.q.P(A)P(B)$$

TABLE 1.4 – *Noisy-OR*

Une variante, le *Leaky Noisy-OR*, permet de spécifier qu'il existe une probabilité, l, non nulle même lorsque les deux probabilités des parents sont nulles, soit pour X, nœud enfant des Y_i :

$l = P(X = \{Vrai\}|Y_i = \{Faux\})$, $\forall i = 1, ..., n$ tel que $0 \leq l \leq 1$

Díez et Druzdzel (2007) proposent notamment la formule permettant de compléter la table pour n noeuds (il existe d'autres variantes du *Noisy-OR* non présentées ici), en considérant l'ensemble des Y_i dans l'état $\{Vrai\}$ (noté Y_v) :

$$P(X = \{Vrai\}|Y_i) = 1 - (1 - l) * \prod_{Y_i \in Y_v} (1 - p_i)$$

$$P(X = \{Faux\}|Y_i) = (1 - l) * \prod_{Y_i \in Y_v} (1 - p_i)$$

A	0		1	
B	0	1	0	1
$P(C)$	l	$q + l(1-q)$	$p + l(1-p)$	$1 - (1-p)(1-q)(1-l)$

TABLE 1.5 – *Leaky Noisy OR* pour 2 nœuds parents

Il existe de nombreuses autres fonctions utilisées dans les réseaux Bayésiens mais qui ne sont pas présentées ici, car nous avons uniquement focalisé sur ce qui sera utilisé dans la suite de ce manuscrit.

1.5 Analyse quantitative : fonction de croyance

Les analyses quantitatives pour la gestion du risque reposent principalement sur des prédictions du comportement des défaillances/événements indésirables en fonction du temps. Pour cela, il est considéré qu'il existe une variabilité naturelle dans le comportement de ces défaillances, c'est-à-dire, que la défaillance d'une entité est un phénomène aléatoire. La théorie des probabilités permet d'attribuer une distribution de probabilités de défaillance sur la durée de vie de l'entité (si l'on connait la variabilité naturelle du phénomène de défaillance). Cependant, une autre proposition émerge en identifiant deux types d'incertitudes (Aven, 2010) :

— les incertitudes épistémiques : liées à un manque de connaissance

— les incertitudes aléatoires : liées à la variabilité d'un phénomène naturel

Aguirre et al. (2013) présentent les débats qui existent autour de ces concepts et mentionnent que la théorie des probabilités confond ces deux types d'incertitudes, et que d'autres théories comme les probabilités imprécises, la théorie des fonctions de croyance et la théorie des possibilités, ont été proposées pour pallier cette ambiguïté.

La théorie des fonctions de croyance, également connue sous le nom de théorie de l'évidence ou théorie de Dempster-Shafer (D-S), est issue des travaux de A.P. Dempster en 1967, repris par G. Shafer en 1976. Cette théorie repose sur la manipulation de fonctions définies sur des sous-ensembles d'un domaine fini Ω appelé cadre de discernement (qui est constitué de toutes les hypothèses). Ces sous-ensembles sont les parties de l'ensemble des parties de Ω, c.à.d de 2^{Ω}.

Exemple Soit ω l'état d'un voyant lumineux qui peut prendre deux valeurs : *allumé*, *éteint*. On obtient alors :
$\Omega = \{allumé, éteint\}$, et $2^{\Omega} = \{\{allumé\}, \{éteint\}, \{allumé, éteint\}, \varnothing\}$.

Les fonctions définies sur ces sous-ensembles dans l'intervalle $[0, 1]$, appelées fonctions de masse, ou masses de croyance vont représenter la croyance que l'on a dans la vérité d'une proposition. Cette croyance est représentée par une masse qui est une fonction de l'ensemble 2^{Ω} des parties de Ω dans $[0, 1]$, et $m(\lambda)$ représentant la croyance que l'on met dans la proposition λ :

$$\sum_{\lambda \subseteq \Omega} m(\lambda) = 1 \tag{1.4}$$

Ainsi, il est possible de modéliser les incertitudes sur plusieurs valeurs. Par exemple, soient A et B deux éléments de Ω, et $\lambda = \{A, B\}$, alors $m(\lambda) = m(\{A, B\})$, représente la croyance que la proposition A ou la proposition B soit vraie, sans savoir laquelle est correcte.

Exemple D'après l'exemple précédent, on peut définir la masse m telle que : $m(\{allumé\}) = 0,2$, $m(\{éteint\}) = 0,5$ et $m(\{allumé, éteint\}) = m(\Omega) = 0,3$

La fonction de crédibilité, Bel, représente la croyance minimale dans une proposition à partir des masses élémentaires de croyance portées par les éléments de masse non nulle. Pour $A \subseteq \Omega$, $Bel(A)$ est la somme des masses données aux sous-ensembles non vides de A. C'est donc la croyance totale allouée à A par la connaissance disponible :

$$Bel(A) = \sum_{\emptyset \neq B \subseteq A} m(B) \quad \forall A \subseteq \Omega \tag{1.5}$$

Exemple $Bel(\{allumé\}) = m(\{allumé\}) = 0,2$

La plausibilité est la fonction duale de la fonction de crédibilité, elle mesure le degré maximal susceptible d'être affecté à une proposition. Pour une proposition $A \subseteq \Omega$, $Pl(A)$ correspond à la somme des masses données aux sous-ensembles non vides qui intersectent A :

$$Pl(A) = \sum_{A \cap B \neq \emptyset} m(B) \quad \forall A \subseteq \Omega \tag{1.6}$$

Exemple $Pl(\{allumé\}) = m(\{allumé\}) + m(\{allumé, éteint\}) = 0,5$

Si on prend le cas particulier de notre exemple, c.à.d, un domaine de deux valeurs, on peut également définir une fonction $disbelief$, qui correspond à la fonction $1 - Pl(A)$ pour une proposition A donnée. Ces concepts peuvent être modélisés comme présenté sur la Figure 1.13.

Une des limites importantes à l'utilisation de cette théorie, est la difficulté d'estimer les fonctions de masses. Pour cela il existe dans la littérature de nombreuses propositions, mais qui dépendent de l'application de cette théorie. On peut noter tout de même, une

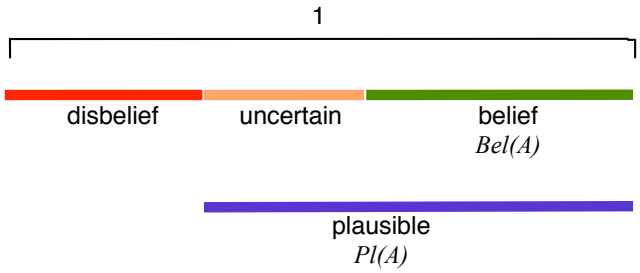


FIGURE 1.13 – Les principaux concepts dans la théorie de Dempster-Shafer

fonction introduite par Dempster, intitulée « à support simple », qui consiste à affecter toute la masse d'une source à un sous-ensemble non vide A de 2^{Ω}. Si l'on applique cette fonction on obtient :

$$\begin{cases} m(A) = s \quad s \in [0,1] \\ m(\Omega) = 1 - s \\ m(B) = 0 \quad \forall B \in 2^{\Omega}, B \neq A, B \neq \Omega \end{cases} \tag{1.7}$$

Exemple D'après l'exemple précédent, si on a $m(\{allumé\}) = 0,3$, alors on aura $m(\{allumé, éteint\}) = m(\Omega) = 0,7$ et $m(\{éteint\}) = 0$

Il existe également dans la théorie de D-S un ensemble d'opérateurs afin de combiner les croyances, mais qui ne seront pas utilisées dans ce manuscrit.

1.6 Argumentaire de sécurité

Suite à de nombreux accidents industriels parmi lesquels l'incendie nucléaire à Windscale en 1957, l'explosion chimique à Flixborough en 1974, l'explosion suivie par un incendie de la plateforme pétrolière Piper Alpha en 1988 ont poussé le gouvernement britannique à obliger les futurs exploitants à présenter des rapports démontrant la sécurité de l'installation et les opérations qui y sont pratiquées. Cette procédure est l'un des premiers exemples d'un dossier de sécurité.

Cependant, il n'existait pas de consigne claire et précise pour analyser ou construire un tel dossier de sécurité. Ce manque de clarté a été la motivation de nombreuses recherches portant sur la construction d'un argumentaire de sécurité que l'on peut aussi bien appliquer à une installation ou à un système. On peut notamment citer les travaux menés dans

le cadre de la thèse de Kelly (1998). Le concept a vite prouvé son intérêt aux yeux du gouvernement britannique et son Ministère de la Défense a rendu obligatoire en 2004 dans la norme DefStan 00-56 (2004) la présence d'un document nommé *Safety case* ou « Dossier de sécurité » (aussi appelé « Cas de sécurité ») pour toute acquisition d'un système de caractère critique. Ce standard définit le Dossier de Sécurité comme (traduction libre) «un document structuré comprenant des éléments de preuve qui présente un argument convaincant, compréhensible et valide prouvant que le système est sûr pour une utilisation prévue dans un environnement prédéfini» [9].

La notion d'argumentaire de sécurité a été introduite pour guider l'élaboration de ce dossier et s'est appuyé sur les travaux de Toulmin (1958), qui a défini les 6 éléments d'un argument solide et réaliste :

- **La revendication (*claim*)** : qui correspond à l'affirmation de ce que l'on estime vrai.
- **Les données (*data* ou *ground*)** : qui constituent les éléments de preuve qui fondent la revendication.
- **Les garanties (*warrant*)** : qui explicitent les principes du raisonnement qui fait le lien entre les données et la revendication.
- **Les fondements (*backing*)** : qui constituent la structure profonde du raisonnement et de l'argumentaire.
- **Les restrictions (*rebuttal*)** : qui signalent les exceptions éventuelles.
- **Les modalités (*qualifier*)** : qui précisent les conditions particulières à respecter pour que la revendication soit vraie.

Différents travaux ont repris ces concepts pour représenter graphiquement un argument avec trois parties de base : les *éléments de preuves* qui supportent un ou plusieurs *arguments*, servant à leur tour à démontrer une *affirmation*. Dans le cas de l'Argumentaire de Sécurité, cette affirmation est souvent un objectif de sécurité précis.

Des notations graphiques ont été proposées pour aider la compréhension du développement d'un argumentaire, par exemple la notation CAE (*Claims-Argument-Evidence*) (Bishop et al. (2004)) et GSN (*Goal Structuring Notation*) (Weaver et Kelly (2004)). CAE reste simple et basique tandis que GSN prend en compte les hypothèses, les contextes dans lequel se situe le système, les liens avec les éléments de preuve et la hiérarchie avec les autres objectifs. La notation KAOS (*Knowledge Acquisition and autOmated Specification*) présentée dans Dardenne et al. (1991, 1993), repose sur le même principe de décomposition des objectifs. Cependant, comme le note Sabetzadeh et al. (2011), l'objectif dans GSN est de type «argumentaire» sans se calquer obligatoirement sur l'architecture du système à la

9. *A Safety Case is a structured argument, supported by a body of evidence, that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given environment*

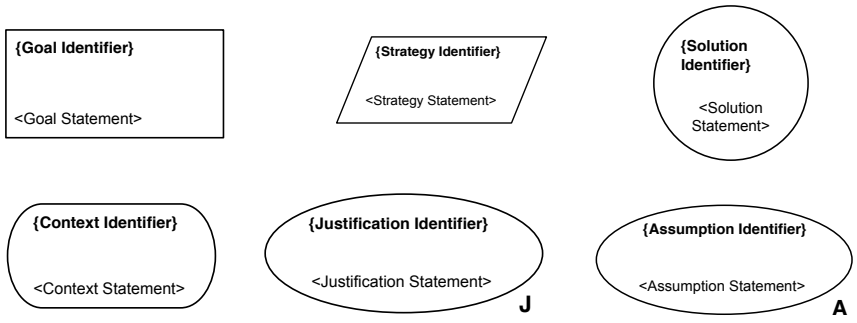


FIGURE 1.14 – Éléments de la notation GSN

différence de KAOS plus proche du système lui même, avec la possibilité par exemple de faire des décompositions d'argumentaire de type OR.

La notation GSN telle qu'elle est formalisée actuellement (GSN-Standard, 2011), est une représentation graphique d'un argument qui met en valeur la relation entre la revendication et les éléments de preuve qui la soutiennent suivant une stratégie précise et dans un contexte donné. Cette notation utilise principalement les symboles présentés sur la Figure 1.14 .

- Revendication (*Goal*) : représente une affirmation faisant partie de l'argumentaire
- Stratégie (*Strategy*) : explique la nature de l'inférence qui relie une affirmation aux éléments qui la soutiennent
- Élément de preuve (*Solution*) : donne référence aux documents servant de preuve de l'affirmation
- Contexte (*Context*) : précise le contexte de l'affirmation, contenant des informations ou des déclarations
- Justification (*Justification*) : présente une déclaration à partir d'un raisonnement
- Hypothèse (*Assumption*) : présente une déclaration intentionnellement sans fondement

Le lien entre les revendications et d'autres revendications, stratégies ou éléments de preuve symbolise une inférence et porte la signification « soutenue par » (flèche à tête pleine). Celui entre les revendications ou les stratégies et les contextes, les justifications, les hypothèses symbolise une information supplémentaire et porte la signification « dans le contexte de » (flèche à tête creuse).

Un exemple simple illustrant un raisonnement est donné Figure 1.15. Dans cet exemple la revendication que le logiciel est apte à l'usage repose uniquement sur des activités de test et de vérification formelle. Les tests effectués avec un oracle défini, supposé correct, montrent que le logiciel fournit des résultats cohérents. La vérification formelle du fonc-

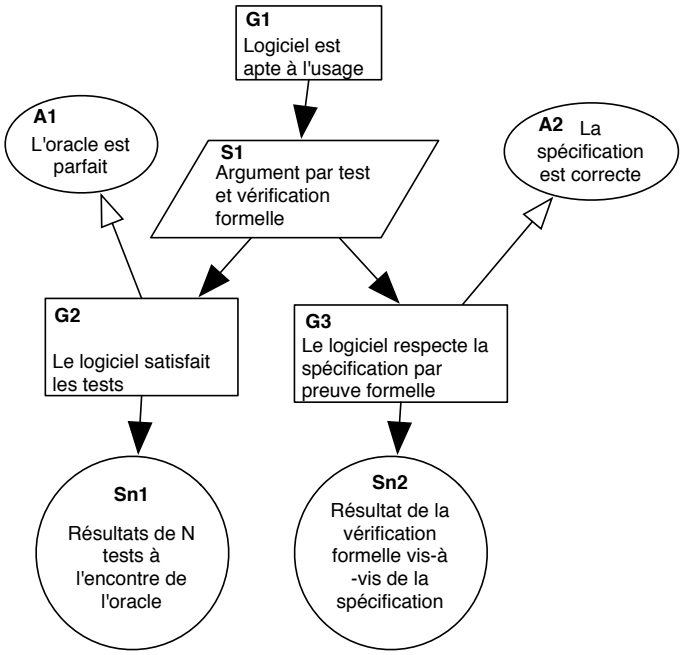


FIGURE 1.15 – Un exemple de l'argumentaire présenté avec GSN

tionnement du système est valide si elle repose sur une spécification correcte. Ces deux affirmations, ensemble, infèrent la revendication « Le logiciel est apte à l'usage ». Cet exemple simple illustre bien la limite de cette approche argumentaire, car rien ne garantit que cette stratégie (S1), est suffisante pour la revendication, si ce n'est l'expertise. Rappelons que nous nous plaçons dans le cadre où le système est trop complexe ou qu'il existe trop d'incertitudes pour appliquer des normes ou démontrer quantitativement que la revendication principale est vraie. La construction d'un argumentaire doit donc se baser sur le jugement d'experts.

Un exemple plus complexe extrait de GSN-Standard (2011) est donné Figure 1.16. Il représente un argumentaire démontrant la sécurité opérationnelle d'un système de contrôle. Cette revendication G1 est supportée par deux sous-revendications : « G2 : tous les dangers sont éliminés ou traités avec succès » et « G3 : la partie logicielle du système de contrôle a été développée en respectant les niveaux d'intégrité SIL, requis pour les dangers concernés ». La première sous-revendication, dans le contexte où la liste des dangers et la limite de la tolérabilité sont connues, est supportée par le traitement de chacun des dangers identifiés, l'inférence utilisée a donc pour hypothèse « A1 : tous les dangers sont identifiés ». L'affirmation du traitement de chacun des dangers est ensuite argumentée par le résultat des méthodes d'analyse de la sûreté de fonctionnement basées sur des arbres

de fautes (Sn2) et sur de la vérification formelle (Sn1). De manière similaire, la deuxième sous-revendication traite les systèmes de protection primaire et secondaire indépendamment et affirme que chaque système répond bien à son exigence de niveau d'intégrité (SIL) en suivant le processus de développement adéquat.

En plus de cette notation, GSN-Standard (2011) propose une méthode pour la construction des argumentaires qui repose notamment sur deux processus pour identifier les éléments de la structure GSN : la méthode descendante *Top - Down* (en partant de la revendication finale) et la méthode ascendante *Bottom - Up* (à partir des éléments de preuve). Il existe également des patterns GSN, dont sans doute le plus utilisé en sécurité (*safety*), est le *Hazard Avoidance Pattern* (Kelly et McDermid, 1997), présenté Figure 1.17, où le but de haut niveau G1 est décomposé en sous objectifs selon la stratégie S1. Dans ce cas, les n dangers doivent avoir été traités, pour pouvoir justifier G1. Si le traitement d'un danger a une confiance très faible, cela doit impacter fortement la confiance dans l'objectif G1. Une instanciation de ce pattern a été donnée dans le modèle GSN Figure 1.17.

1.7 Vue générale des contributions de la thèse

Comme nous l'avons présenté précédemment Section 1.1, malgré l'existence de normes pour certains domaines robotiques, la diversité des applications rend très peu probable l'application directe de ces normes, et par conséquent la certification ne peut se faire comme dans les autres domaines à sécurité critique. Ainsi, il est important de proposer des techniques permettant d'étudier la sécurité de tels systèmes, mais également de pouvoir la justifier auprès des organismes de certification.

L'approche que nous proposons ici s'inscrit dans le processus de gestion du risque standardisé présenté Chapitre 2. La vue globale de cette approche est présentée Figure 1.18. Dans l'activité d'analyse du risque nous proposons de combiner la technique HAZOP présentée Section 1.3 avec le langage de modélisation UML (*Unified Modeling Language*). Nous montrerons dans le Chapitre 2, comment à partir de trois diagrammes UML, il est possible d'effectuer grâce à la méthode HAZOP-UML, une identification des dangers opérationnels (c'est-à-dire, ceux liés à l'exécution d'une tâche, lors de la vie opérationnelle du système). À partir de cette liste de dangers, il est ensuite possible d'identifier et d'estimer les risques, en suivant les étapes classiques du processus de gestion du risque comme présenté dans la Section 1.2.

Puis pour l'étape d'évaluation du risque, et de décision d'acceptabilité, face à l'impossibilité d'obtenir des évaluations quantitatives (par exemple, il n'existe aucune information sur les taux de défaillance du logiciel, ou sur les taux d'erreur humain), nous proposons d'utiliser un argumentaire de sécurité, modélisé en GSN (présenté Section 1.6). Cependant,

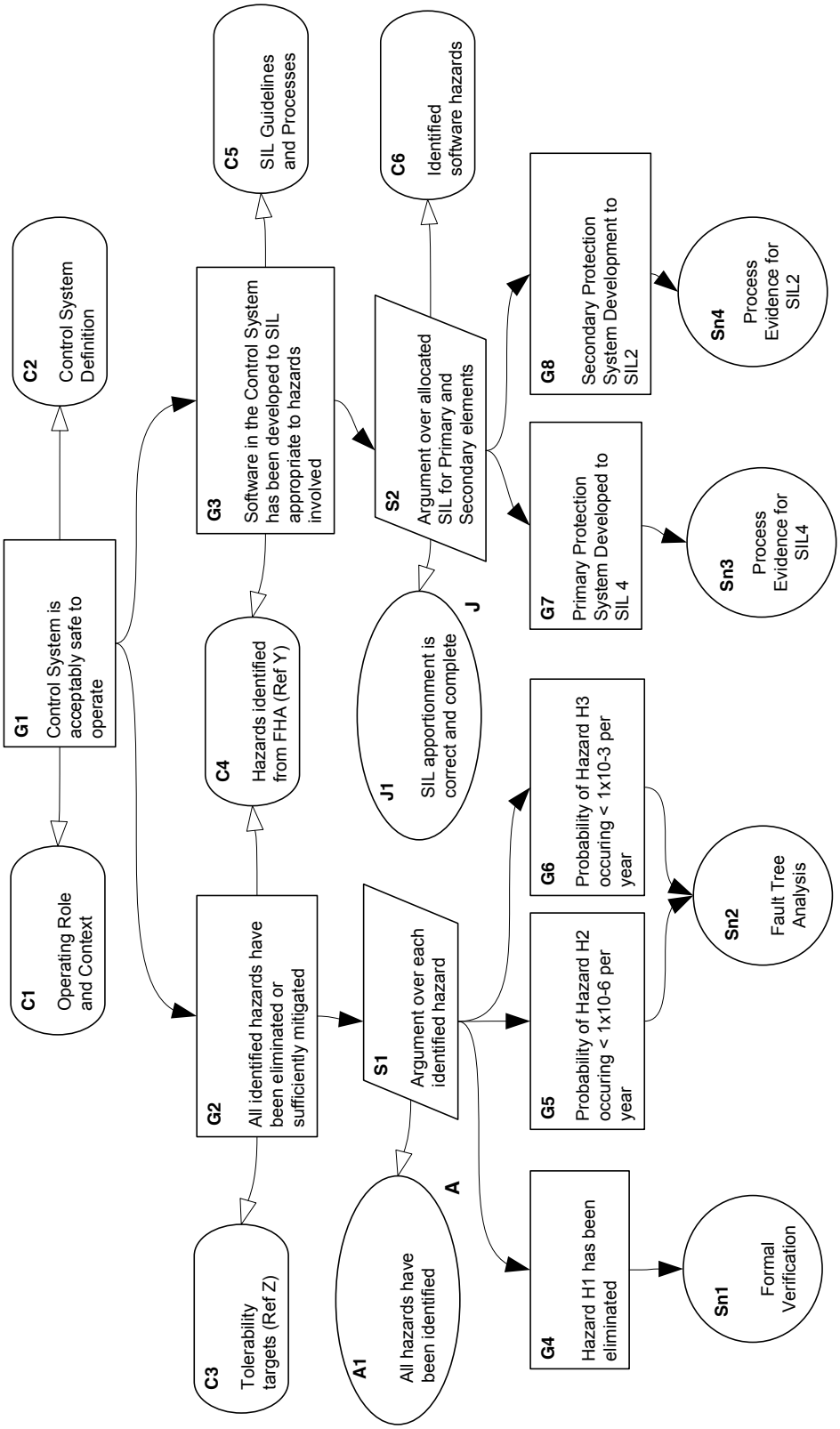


FIGURE 1.16 – Un exemple de l'argumentaire présenté avec GSN tiré de GSN-Standard (2011)

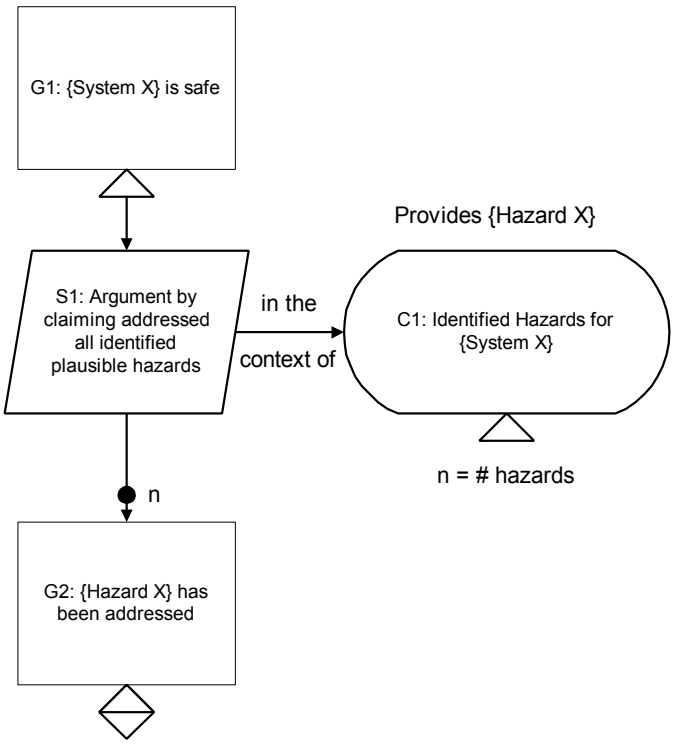


FIGURE 1.17 – GSN Hazard Avoidance Pattern

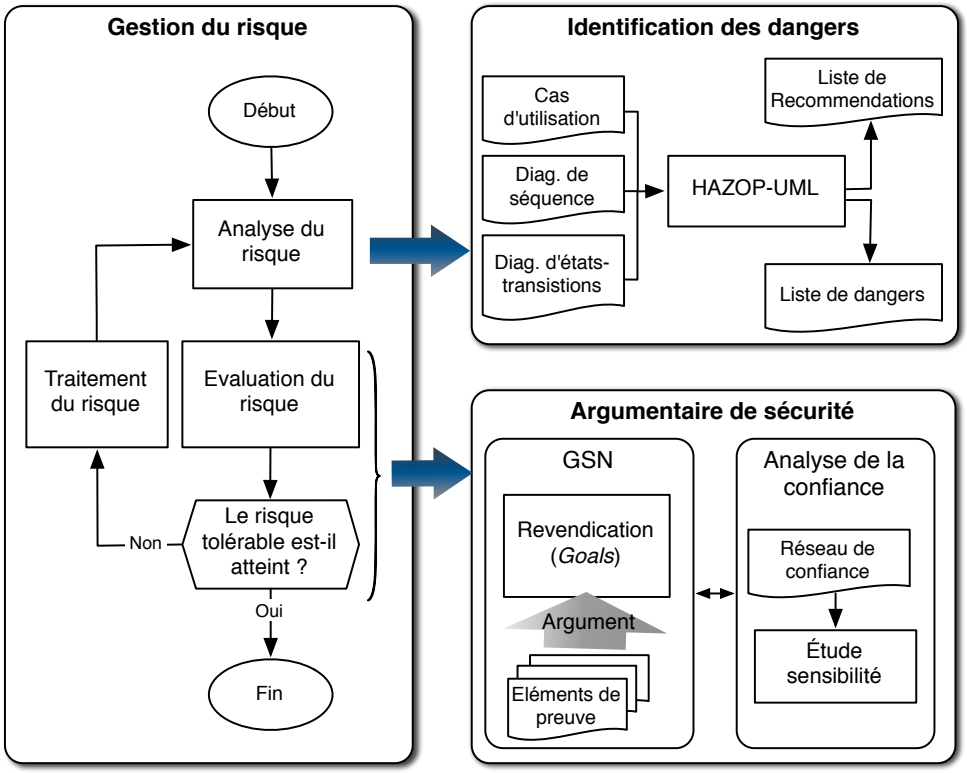


FIGURE 1.18 – Vue générale des contributions de la thèse au processus de gestion du risque

comme nous l'avons souligné auparavant, se pose le problème de la confiance que l'on peut placer dans ces argumentaires, fortement dépendants du niveau d'expertise de la personne réalisant l'étude. Nous proposons donc dans le Chapitre 3 une nouvelle approche permettant de construire un réseau de confiance associé à un argumentaire de sécurité. Ce réseau est basé sur les réseaux Bayésiens (Section 1.4), mais s'appuie également sur la fonction de croyance de D-S (Section 1.5). Ce réseau permettra notamment d'effectuer des analyses de sensibilité afin de détecter les faiblesses de l'argumentaire de confiance, et de l'améliorer.

Comme présenté sur la Figure 1.18, les deux contributions de cette thèse s'insèrent dans le processus de gestion du risque, contenant d'autres activités (comme l'estimation du risque par exemple). Nous présenterons dans le Chapitre 4 les résultats que nous avons obtenus lors de l'application de cette approche dans le cas concret de l'analyse d'un déambulateur robotisé développé dans le cadre du projet ANR MIRAS.

Chapitre 2

Analyse du risque basée modèle

2.1 Introduction

Les processus de développement de systèmes à sécurité critique ont rapidement intégré des activités de gestion du risque comme présenté dans le Chapitre 1. Une des premières activités consistait à comprendre le système du point de vue de ses fonctions, ses frontières, ainsi que l'utilisation attendue. Les étapes suivantes d'analyse, d'estimation et d'évaluation du risque, reposaient donc sur la qualité de cette compréhension. Conjointement au développement des techniques d'analyse du risque, des langages de modélisation des systèmes sont apparus, et c'est tout naturellement que de nombreux travaux ont porté sur la connexion entre les techniques de modélisation système et celles d'analyse du risque. Plus généralement, le concept d'analyse de sécurité basée sur les modèles (*mode-based safety analysis*) est apparu, et il existe aujourd'hui de nombreux travaux et outils sur ce sujet.

Après un état des lieux de ces travaux, ce chapitre présentera la méthode HAZOP-UML. Cette méthode, initiée au LAAS-CNRS avant le début de cette thèse, a été enrichie et finalisée lors de cette thèse. Parmi les points importants de notre contribution, nous montrerons comment les diagrammes d'états-transitions ont été intégrés à la méthode. La validation de cette approche sera donnée ultérieurement dans le Chapitre 4, où nous présenterons également comment HAZOP-UML s'insère complètement dans un processus de gestion du risque.

2.2 Utilisation des modèles dans l'analyse du risque

Les premiers travaux autour de l'analyse de risque (ou sécurité) basée modèle, se sont attachés à réaliser des ponts entre techniques de modélisation et techniques d'analyse de

risque comme les arbres de fautes (AdF), l'analyse des modes de défaillance et de leurs effets critiques (AMDEC). La méthode HIPS-HOPS (*Hierarchically Performed Hazard Origin and Propagation Studies*) puis l'outil associé, développé à l'université de Hull [1], illustre parfaitement cette transition en générant automatiquement des AdF et des AMDEC à partir de modèles du système (par exemple exprimés avec SIMULINK) et d'annotations de défaillance pour chaque composant). L'approche TOPCASED [2] s'insère également dans cette catégorie mais en proposant un ensemble de modèles tournés vers UML mais sans génération de données pour des outils d'analyse du risque.

Parallèlement, plusieurs projets de recherche européens comme ESACS (2001-2003) [3] dans le domaine des transports, suivi de ISAAC (2004-2007) [4] dans l'avionique, puis CESAR (2009-2012) [5] suivi de CRYSTAL (2013-2017) [6] dans les systèmes embarqués, ont eu au cœur de leurs problématiques l'analyse de sécurité basée modèle.

De manière générale, les activités réalisées au sein de ces méthodes peuvent se regrouper selon trois catégories (Blanquart, 2010) :

1. analyse de la propagation des fautes
   (a) *bottom-up* : effets d'une faute sur le système
   (b) *top-down* : recherche des fautes induisant un événement redouté
2. vérification d'exigences de sûreté de fonctionnement (ou plus précisément de sécurité)
3. quantification de la probabilité d'occurrence d'évènements redoutés

La proposition présentée dans ce mémoire s'inscrit dans la première catégorie. L'objectif initial étant d'identifier des scénarios dangereux, notamment lors des interactions homme-robot, la méthode HAZOP-UML s'apparente à de l'analyse de propagation de fautes (même si par faute on considèrera ici des défaillances ou des événements non prévus). Le choix d'UML s'est imposé par l'utilisation de plus en plus importante de cet langage de modélisation.

Les travaux les plus proches de notre approche ont été menés dans le cadre du projet (CORAS, 2014; Bjørn Axel Gran et Thunem, 2004). Dans ce projet une méthodologie utilisant les concepts d'analyse de risques et de modélisation orientée objet a été développée pour l'évaluation de risques des systèmes à sécurité critique vis-à-vis des malveillances.

1. http ://hip-hops.eu/
2. http ://www.topcased.org
3. http ://www.transportresearch.info/web/projects/project_details.cfm ?ID=2658
4. http ://ec.europa.eu/research/transport/projects/items/isaac_en.htm
5. http ://www.cesarproject.eu
6. http ://www.crystal-artemis.eu/

Dans notre cas, nous privilégions la sécurité-innocuité (*safety*) par rapport à la sécurité-confidentialité (*security*), mais l'objectif de nos études reste similaire à celui du projet CORAS. Cependant, nous n'avons pas les même revendications dans les diagrammes UML (focalisant sur HAZOP et Arbre de Fautes). Une différence majeure dans notre approche est la forte connexion entre les modèles UML et la technique HAZOP, qui n'existe pas dans le projet CORAS. Ainsi, ils utilisent la technique HAZOP sans aucun lien explicite vers les modèles UML (leurs mots-guide de HAZOP ne sont pas applicables aux modèles UML).

Notre approche d'analyse du risque est fondée sur la ré-interprétation des mots-guides de la technique HAZOP dans le contexte des différents modèles UML. La proposition de Lano et al. (2002), suivie par une étude systématique de Hansen et al. (2004), considère également l'interprétation des mots-guides de HAZOP dans la déviation des éléments de UML comme classe, association, rôle, message etc. Une autre approche similaire de (Gorski et Jarzebowicz, 2005) et (Jarzebowicz et Górski, 2006) présente aussi une analyse statistique de l'utilisabilité de la méthode. L'interprétation des mots-guides appliqués aux diagrammes UML statiques dans ces approches cherche à inspecter le modèle pour identifier les fautes de conception et non pas les déviations opérationnelles. Toutefois, pour les diagrammes UML dynamiques (diagramme de cas d'utilisation, de séquence, d'activité et d'états), l'interprétation de plusieurs mots-guides peut servir à l'examen des déviations relatives à sa phase opérationnelle. Ceci est le cas des études présentées par Johannessen et al. (2001) et de façon plus formelle par Allenby et Kelly (2001) qui se concentrent sur les diagrammes de cas d'utilisation. Ces travaux sur le diagramme de cas d'utilisation ont inspiré l'étude présentée par Srivatanakul (2005) qui s'est concrétisée par le développement d'une approche pour l'analyse de risques vis-à-vis des malveillances basée sur de nouvelles interprétations des mots-guides. Même si ces travaux sont orientés davantage aux risques liés à une utilisation malveillante des opérateurs, certaines interprétations restent applicables aux systèmes critiques en interaction avec des humains, en prenant en compte des fautes d'origine accidentelle.

Dans le cadre de nos travaux, nous avons combiné et étendu les résultats de ces études, en nous intéressant aux diagrammes de cas d'utilisation, de séquence et d'états-transitions pour examiner les déviations relatives à la phase opérationnelle. Nous accordons une attention particulière à l'intégration des techniques d'analyse des erreurs d'interaction humaine basées sur HAZOP comme présenté par Guiochet et al. (2004), qui sont au cœur des interactions humain-robot. En effet, la prise en compte de ces types d'erreur est un problème majeur dans les systèmes critiques (Stanton et al., 2006), cependant leur analyse n'est pas toujours corrélée avec la modélisation préliminaire du système.

2.3 Langage de modélisation unifié UML

UML (en anglais *Unified Modeling Language* ou Langage de Modélisation Unifié) est un langage de modélisation graphique. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ». Couramment utilisé dans les projets logiciels, il peut être appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique. Sur le site de l'OMG (OMG, Consulté le 13 mars 2010), UML2 propose treize types de diagrammes (contre neuf en UML1.3). UML n'étant pas une méthode, le choix de diagrammes à utiliser est laissé libre aux utilisateurs. De même, on peut se contenter de modéliser partiellement un système, par exemple certaines parties critiques.

Les 13 diagrammes UML (listés Figure 2.1) sont dépendants hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie. Ils peuvent être rassemblés selon deux catégories : les diagrammes structurels et les diagrammes comportementaux. Il existe six diagrammes structurels :

1. **Le diagramme de classes** représente les classes intervenant dans le système.
2. **Le diagramme d'objets** sert à représenter les instances de classes (objets) utilisées dans le système.
3. **Le diagramme de composants** il permet de montrer les composants du système d'un point de vue physique, tels qu'ils sont mis en œuvre (fichiers, bibliothèques, bases de données...)
4. **Le diagramme de déploiement** sert à représenter les éléments matériels (ordinateurs, périphériques, réseaux, systèmes de stockage...) et la manière dont les composants du système sont répartis sur ces éléments matériels et interagissent entre eux.
5. **Le diagramme de paquetages** un paquetage étant un conteneur logique permettant de regrouper et d'organiser les éléments dans le modèle UML, le diagramme de paquetages sert à représenter les dépendances entre paquetages, c'est-à-dire les dépendances entre ensembles de définitions.
6. **Le diagramme de structures composites** permet de décrire sous forme de boîte blanche les relations entre composants d'une classe.

En complément de ces diagrammes statiques, il existe également sept diagrammes comportementaux, dont quatre diagrammes d'interaction ou dynamiques :

1. **Le diagramme des cas d'utilisation (use-cases)** permet d'identifier les possibilités d'interaction entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire tous les usages que doit permettre le système.

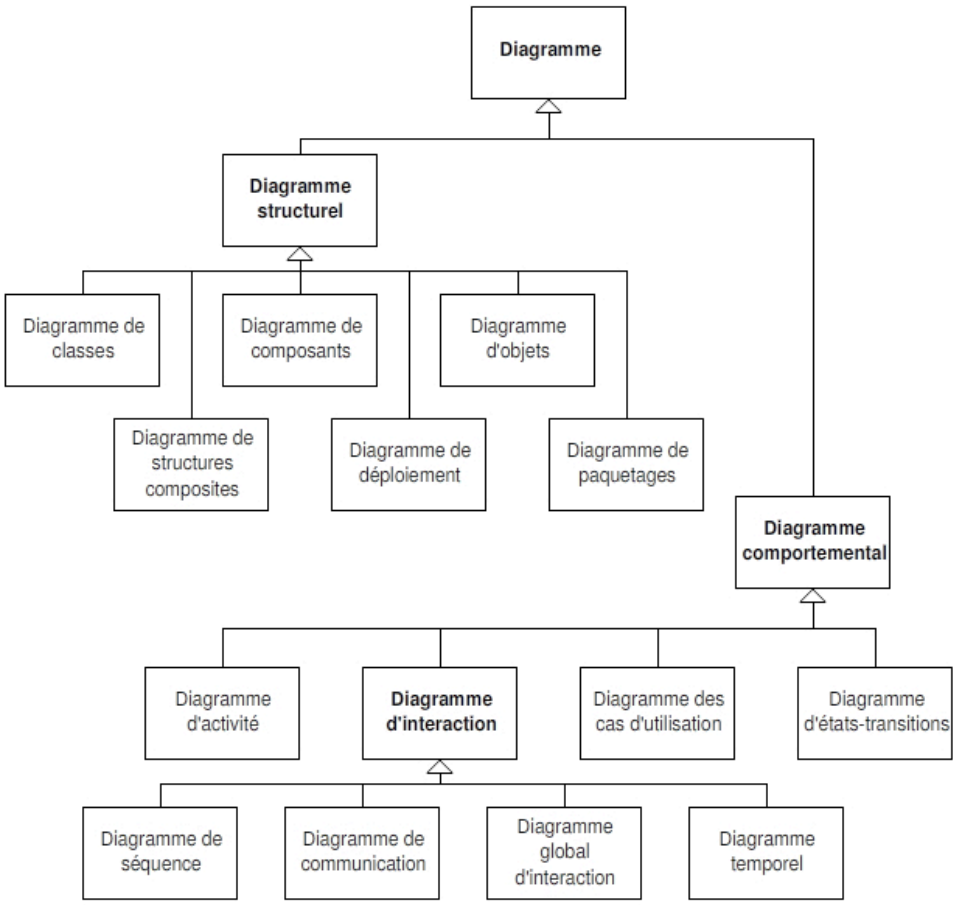


FIGURE 2.1 – Les 13 diagrammes d'UML2

2. **Le diagramme d'états-transitions** permet de décrire sous forme de machine à états finis le comportement du système ou de ses composants.
3. **Le diagramme d'activité** permet de décrire sous forme de flux ou d'enchaînement d'activités le comportement du système ou de ses composants.
4. **Le diagramme de séquence** est une représentation séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs.
5. **Le diagramme de communication** est une représentation simplifiée d'un diagramme de séquence se concentrant sur les échanges de messages entre les objets.
6. **Le diagramme global d'interaction** permet de décrire les enchaînements possibles entre les scénarios préalablement identifiés sous forme de diagrammes de séquence (variante du diagramme d'activités).
7. **Le diagramme temporel** permet de décrire les variations d'une donnée au cours du temps sous la forme d'un chronogramme.

Dans l'approche que nous proposons, l'objectif est de permettre d'identifier les dangers dès le début d'un projet. Or dans la plupart des méthodes associées à UML, les deux diagrammes utilisés en premier sont les diagrammes des cas d'utilisation et de séquence. Le diagramme d'état-transitions, est également utile lors des premières étapes d'un processus pour spécifier le comportement attendu d'un système robotique. Par la suite, nous nous intéressons à ces trois diagrammes en particulier (diagramme des cas d'utilisation, diagramme de séquence et diagramme d'état-transitions), en mettant l'accent sur les éléments qui sont fondamentaux pour notre approche.

2.3.1 Diagramme des cas d'utilisation

UML définit une notation graphique pour représenter les cas d'utilisation, cette notation est appelée diagramme de cas d'utilisation. Un cas d'utilisation est la description d'un objectif à atteindre par un acteur qui utilise le système (comme par exemple « Programmer la tâche » comme sur la Figure 2.2). Il correspond à une classe de scénarios. Un acteur est une entité externe qui interagit avec le système, comme une personne humaine ou un autre système. L'activité du système a pour objectif de satisfaire les besoins de l'acteur, c'est-à-dire d'exécuter le cas d'utilisation. Cependant, cette exécution peut également amener à des exceptions ou des scénarios alternatifs.

La figure 2.2 montre un exemple de diagramme de cas d'utilisation d'un robot. Deux acteurs ont été identifiés, Opérateur et Utilisateur, correspondant aux deux rôles différents vis-à-vis du système robotique (représenté par le cadre externe). Ces acteurs attendent de

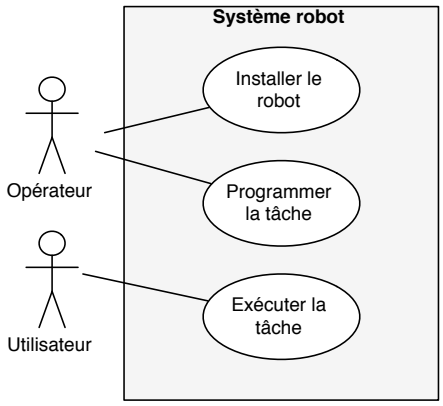


FIGURE 2.2 – Diagramme de cas d'utilisation d'un robot générique

la part du système les services suivants : installer le robot et programmer la tâche pour l'opérateur, et exécuter la tâche pour l'utilisateur. Ce petit exemple générique illustre la manière dont est utilisé le diagramme de cas d'utilisation dans la suite de ce manuscrit. Il est possible de trouver d'autres représentations, notamment certaines qui représentent le robot lui même en tant qu'acteur. Le système alors représenté par le cadre est le contrôleur de robot uniquement, c.à.d, tout sauf les parties mécaniques. Cependant, pour certains composants, la frontière entre mécanique et électronique est parfois difficile à établir. Nous avons préféré cette représentation plus simple, et moins ambiguë.

Un diagramme de cas d'utilisation n'existe pas sans une description textuelle pour chaque cas d'utilisation. Il est en effet très facile de lire un diagramme, mais beaucoup d'informations fondamentales n'apparaissent pas. Un exemple de fiche est donné dans la figure 2.3. Les différents éléments à fournir sont explicités dans la colonne de droite. Il est à noter que dans la ligne « Exigences non fonctionnelles », sont notées les exigences n'apparaissant pas sur les diagrammes des cas d'utilisation, et qui peuvent concerner toute exigence de sûreté de fonctionnement, performance, ou d'ergonomie. Nous utiliserons par la suite cette ligne pour exprimer des contraintes de sécurité valides tout au long de l'exécution du scénario nominal.

2.3.2 Diagramme de séquence

Les diagrammes de séquence permettent de représenter des interactions entre objets selon un point de vue temporel, l'accent est mis sur la chronologie des envois de messages. Ces diagrammes servent notamment à décrire un cas d'utilisation. La figure 2.4 résume les éléments principaux d'un diagramme de séquence. Une ligne de vie *(lifeline)*, part

Cas d'utilisation		*<nom du cas d'utilisation>*
Acteurs		*<acteurs liés au cas d'utilisation>*
Préconditions		*<préconditions au lancement du cas d'utilisation>*
Déroulement normal	Description détaillée	*<scénario nominal du cas d'utilisation>*
	Postconditions	*<post-condition en cas de déroulement attendu du cas d'utilisation>*
Déroulement alternatif	Exceptions	*<levée d'exception lors de l'exécution du scénario nominal>*
	Variantes	*<variante du scénario nominal>*
Exigences non fonctionnelles		*<Toute exigence liée à la sûreté de fonctionnement, aux performances, à l'ergonomie, etc. du cas d'utilisation>*

FIGURE 2.3 – Fiche de cas d'utilisation

d'un objet (par exemple « :Système») et descend verticalement en pointillé (le temps est représenté comme s'écoulant du haut vers le bas le long des « lignes de vie » des objets). Entre ces lignes de vie sont représentés des messages. La signature des messages apparaît sur chaque flèche ainsi que les paramètres associés. Il est possible de représenter, en utilisant le concept de message, des envois de signaux, des appels de méthodes, des déclenchements d'activités, etc. Dans notre cas, nous utiliserons les diagrammes de séquence pour représenter les interactions homme-robot. Elles sont en général de trois types :

— *indirectes* via des pendants [7] ou interfaces de commande informatiques ou matérielles
— *cognitives* par l'utilisation de gestes ou de paroles perçus par des capteurs du robot
— *physiques* par le contact direct entre l'homme et la structure physique du robot

Lors de la modélisation des interactions au début du processus de développement, il n'est parfois pas encore décidé quel type d'interaction sera mis en œuvre. L'utilisation des diagrammes de séquence est particulièrement adapté à cette situation, car on peut réduire un message à la simple spécification de l'interaction (et pas sa réalisation). Par exemple, si le robot doit percevoir le message « mettre le robot en pause », cela sera représenté par un message entre l'utilisateur et le robot, mais pourra être réalisé par la suite par un dispositif électronique (un bouton poussoir), ou par un geste spécifique de l'humain capté par le robot.

7. Le terme de « pendant » provient de la robotique industrielle où les robots sont contrôlés à partir de pupitres ou de télécommandes suspendues (nommées « pendants »)

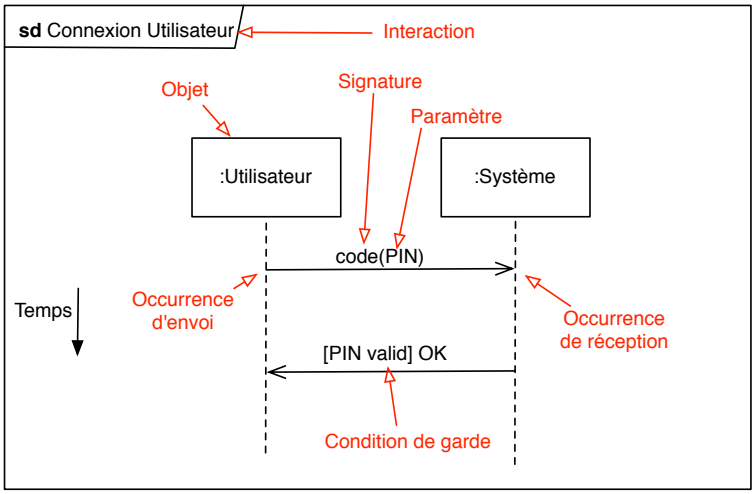


FIGURE 2.4 – Éléments principaux d'un diagramme de séquence

Il existe de nombreux autres concepts en UML pour les diagrammes de séquence, notamment les « fragments » d'interaction, permettant de rajouter sur les diagrammes des boucles, des branchements conditionnels, des contraintes de séquencement, etc. Nous n'utiliserons pas les fragments par la suite, car ils correspondent rarement à des concepts utilisés pour les diagrammes de séquence « système » (diagrammes où seuls les acteurs et le système sont représentés).

2.3.3 Diagramme d'états-transitions

Un diagramme d'états-transitions est un schéma utilisé pour représenter des automates déterministes. Il s'inspire principalement du formalisme des statecharts d'Harel (1987). La puissance d'expression de cette représentation permet de modéliser le fonctionnement global du système, mais peut également être utilisé pour spécifier et implémenter des algorithmes. Un état représente le système à un instant donné, souvent pendant l'exécution d'une action. En UML, il est défini par la valeur de ses attributs à un instant donné. Une transition relie deux états, elle est représentée par une flèche. En plus des états de départ (au moins un) et d'arrivée (nombre quelconque), une transition peut comporter les éléments facultatifs suivants :

— Un événement `event`
— Une condition de garde `guard`
— Une liste d'actions `action`

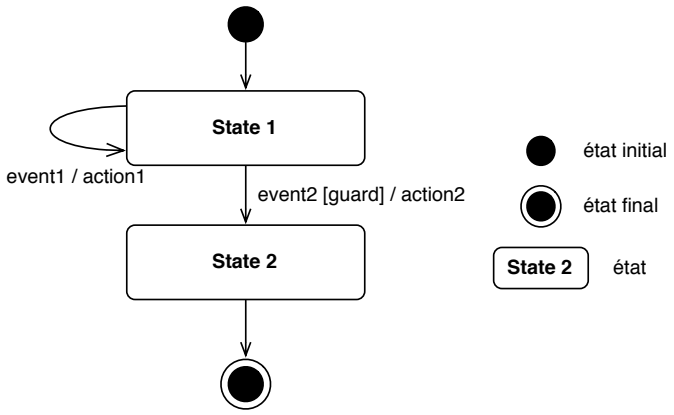


FIGURE 2.5 – Exemple de diagramme d'états-transitions

Une transition peut avoir une condition de garde (spécifiée par '[' <guard> ']' dans la syntaxe). Il s'agit d'une expression logique sur les attributs de l'objet, associée au diagramme d'états-transitions, ainsi que sur les paramètres de l'événement déclencheur. La condition de garde est évaluée uniquement lorsque l'événement déclencheur se produit. Si l'expression est fausse à ce moment là, la transition ne se déclenche pas. Si elle est vraie, la transition se déclenche et ses effets se produisent.

Les événements peuvent être de quatre types :

1. Événement de type signal (*signal*) : une occurrence asynchrone d'un événement externe à la machine à états (Ex : Bouton STOP enfoncé).
2. Événement d'appel (*call*) : réception de l'appel d'une opération par un objet. Les paramètres de l'opération sont ceux de l'événement d'appel.
3. Événement de changement (*change*) : un événement de changement est généré par la satisfaction (i.e., passage de faux à vrai) d'une expression booléenne sur des valeurs d'attributs.
4. Événement temporel (*after* ou *when*) : délai expiré $after(< durée >)$, ou arrivée d'un temps absolu $when(date =< date >)$

Ces éléments sont représentés de manière générique sur la figure 2.5. D'autres éléments sont parfois utilisés comme les états composites (disjoints ou concurrents) ou des représentations des points de décision ou de jonction. Comme pour les autres diagrammes, nous avons fait le choix de ne pas utiliser ces représentations complémentaires car :

— elles peuvent se représenter avec la notation classique,
— l'objectif est de faire des diagrammes les plus simples possibles pour communiquer avec les experts en sécurité.

2.4 Méthode d'analyse du risque HAZOP-UML

2.4.1 Introduction

Parmi les méthodes d'analyse des risques, beaucoup sont aujourd'hui adaptées pour s'appuyer sur des modèles du système, exprimés par exemple en UML. Cependant, très peu s'appuient sur des modèles de spécifications et encore moins en intégrant l'interaction humain-système. C'est dans ce contexte qu'une adaptation de la méthode HAZOP présenté Section 1.3 au diagramme de séquence, de cas d'utilisation et d'états-transition, est proposée. Cette méthode a été initiée et partiellement décrite dans Martin-Guillerez et al. (2010b,a), mais a été étendue et validée lors de ce travail de thèse.

Elle est basée sur une description en UML (*Unified Modeling Language*) des scénarios d'utilisation et des interactions humain-robot, puis effectue une analyse des déviations de ces scénarios en se basant sur une technique similaire à HAZOP (HAZard OPerability). Celle-ci permet de traiter chaque élément de ces modèles en appliquant des listes de mots-guides pour étudier les déviations de ces éléments de base puis en estimer les effets. Le fait de coupler HAZOP et UML présente trois apports importants par rapport aux techniques classiques d'analyse des risques comme l'AMDEC (Analyse des Modes de Défaillance, et de leurs Effets Critiques) et les Arbres de Fautes :

- les techniques de modélisation (représentation graphique des interactions et du système avec un sous-ensemble de la notation UML) permettent de communiquer avec des personnes non spécialistes de la conception (comme les patients ou les spécialistes médicaux) ;
- la cohérence avec le processus de développement est assurée car les modèles sont les mêmes que ceux utilisés lors de la conception ;
- c'est une approche « centrée utilisateur » car l'utilisateur reste toujours au cœur des modèles.
- l'analyse du risque HAZOP-UML peut être réalisée très tôt dans les premières étapes du développement du système, dès la description du besoin. À cette étape, la conception et les spécifications ne sont pas encore claires, seules les fonctions désirées du système et leur déroulement sont modélisables avec des diagrammes de cas d'utilisation et des diagrammes de séquences.

Il est important de noter que l'application de cette méthode ne se substitue pas aux techniques comme l'AMDEC ou les AdF, mais vient en amont de celles-ci, afin d'identifier clairement les dangers majeurs opérationnels auxquels seront confrontés les utilisateurs. S'il faut par exemple estimer la probabilité d'une combinaison de défaillances, il faudra ensuite passer par une technique adaptée comme les AdF.

Cas d'utilisation	*<Nom du cas d'utilisation>*
Description	*<Scénario nominal du cas d'utilisation>*
Préconditions	*<préconditions au lancement du cas d'utilisation>*
Postconditions	*<postconditions en cas de succès du cas d'utilisation>*
Invariants	*<Conditions à vérifier pendant l'exécution du cas d'utilisation>*

FIGURE 2.6 – Fiche de cas d'utilisation utilisée pour HAZOP-UML

La méthode HAZOP-UML proposée dans cette section, est basée sur le processus représenté dans la Figure 1.10. Un important travail a donc consisté à traduire pour les modèles UML ce que signifient les concepts d'éléments et d'attributs (au sens d'HAZOP), puis à déterminer des déviations génériques, c'est-à-dire des interprétations de l'application de chaque mot-guide à chaque attribut des diagrammes UML considéré.

2.4.2 Mots-guide appliqués au diagramme des cas d'utilisation

En UML, les diagrammes de cas d'utilisation font partie des diagrammes comportementaux. Ils permettent d'identifier les possibilités d'interaction entre le système et les acteurs, c'est-à-dire toutes les fonctionnalités que doit fournir le système. La première « entité » au sens de la méthode HAZOP que nous proposons de traiter ici est un cas d'utilisation. Il est donc important de déterminer quels sont les « attributs » d'un cas d'utilisation, sur lesquels seront appliqués les mots-guide.

Comme présenté précédemment Figure 2.3, pour chacun des cas d'utilisation, il est possible de déterminer les **préconditions**, et les **postconditions**. Nous proposons d'utiliser également le concept d'**invariant** (condition nécessaire pendant le déroulement de l'opération) qui reprend de manière plus précise la catégorie « Exigences non fonctionnelles ». Il est en effet apparu que lors de la modélisation en UML, les experts du domaine fournissent dès le début des invariants (notamment de sécurité), pour chaque cas d'utilisation. Ces invariants, généralement sous forme de contraintes (par exemple : la vitesse du robot ne doit pas dépasser 20cm/s) peuvent être issus d'experts, de standards, ou d'autres analyses de sécurité réalisées en parallèle. La Figure 2.6 présente la fiche d'un cas d'utilisation avec les éléments de base utilisés pour l'application d'HAZOP-UML.

Entity = Use Case		
Attribute	**Guideword**	**Interpretation**
Preconditions / Postconditions / Invariants	No/none	The condition is not evaluated and can have any value
	Other than	The condition is evaluated true whereas it is false The condition is evaluated false whereas it is true
	As well as	The condition is correctly evaluated but other unexpected conditions are true
	Part of	The condition is partially evaluated Some conditions are missing
	Early	The condition is evaluated earlier than required (other condition(s) should be tested before) The condition is evaluated earlier than required for correct synchronization with the environment
	Late	The condition is evaluated later than required (condition(s) depending on this one should have already been tested) The condition is evaluated later than required for correct synchronization with the environment

TABLE 2.1 – Liste des mots-guides pour l'HAZOP-UML appliquée aux cas d'utilisation

Ces éléments sont des paramètres qui, couplés avec certains mots-guides précisés dans la table 2.1, permettent d'étudier la déviation correspondante. Ces mots-guides et leur interprétation ont été déterminés précédemment et validés sur plusieurs cas d'étude. Nous avons conservé la version anglaise car c'est celle qui a été utilisée et validée lors de plusieurs cas d'étude (notamment projet PHRIENDS et MIRAS présentés dans le Chapitre 4). À titre d'exemple, imaginons un cas d'utilisation où la précondition est « Le robot est en mode débrayé » . L'application du mot-guide *Other than* pourrait donner comme déviation « Le robot n'est pas en mode débrayé alors que le contrôleur considère qu'il est ».

2.4.3 Mots-guide appliqués au diagramme de séquence

En complément des diagrammes de cas d'utilisation, les diagrammes de séquences présentent les interactions entre le système et les acteurs. Comme pour les cas d'utilisation, la table 2.2 permet de choisir les mots-guides adéquats lors de l'utilisation de la méthode HAZOP-UML.

Comme pour les cas d'utilisation, cette table résulte d'études préliminaires à cette thèse, mais a été finalisée dans le cadre du travail présenté ici.

Entity = Sequence Diagram		
Attribute	**Guideword**	**Interpretation**
Predecessors / successors during interaction	No	Message is not sent
	Other than	Unexpected message is sent
	As well as	Message is sent as well as another message
	More than	Message sent more often than intended
	Less than	Message sent less often than intended
	Before	Message sent before intended
	After	Message sent after intended
	Part of	Only a part of a set of messages is sent
	Reverse	Reverse order of expected messages
Message timing	As well as	Message sent at correct time and also at incorrect time
	Early	Message sent earlier than intended time
	Later	Message sent later than intended time
Sender / receiver objects	No	Message sent to but never received by intended object
	Other than	Message sent to wrong object
	As well as	Message sent to correct object and also an incorrect object
	Reverse	Source and destination objects are reversed
	More	Message sent to more objects than intended
	Less	Message sent to fewer objects than intended
Message condition	No/none	The condition is not evaluated and can have any value (omission)
	Other than	The condition is evaluated true whereas it is false, or vice versa (commission)
	As well as	The condition is well evaluated but other unexpected conditions are true
	Part of	Only a part of condition is correctly evaluated
	Late	The condition is evaluated later than required (other dependent condition(s) have been tested before) The condition is evaluated later than correct synchronization with the environment
Message parameters / return parameters	No/None	Expected parameters are never set / returned
	More	Parameters values are higher than intended
	Less	Parameters values are lower than intended
	As Well As	Parameters are also transmitted with unexpected ones
	Part of	Only some parameters are transmitted Some parameters are missing
	Other than	Parameter type / number are different from those expected by the receiver

TABLE 2.2 – Liste des mots-guides pour l'HAZOP-UML appliquée aux séquences

2.4.4 Mots-guide appliqués au diagramme d'états-transitions

Un diagramme d'états-transitions met en avant le comportement d'un système face à une situation donnée. Ce comportement se traduit par le passage d'un état à un autre état, ce sont donc les transitions et l'état de destination qui retiennent notre attention pour l'application de HAZOP-UML.

Pour éviter l'explosion combinatoire, nous avons choisi un sous-ensemble des éléments présents dans la notation mais qui permet de tout représenter :

1. les états,
2. les transitions,

3. les événements initiateurs,
4. les conditions,
5. les actions.

Les éléments comme les sous-états, les états concurrents, les points de décision ou les points de jonction ne sont pas retenus ici. En effet, pour chacun de ces éléments, la notation de base permet d'exprimer ces concepts. Par exemple, pour remplacer un point de décision permettant de simuler un choix (si/alors/sinon), nous proposons deux transitions partant du même état mais ayant des conditions d'activation différentes.

Comme pour les diagrammes de cas d'utilisation ou les diagrammes de séquences, il a été nécessaire de redéfinir les mots-guides adaptés à chaque élément. La Table 2.3 présente le résultat des interprétations génériques pour les mots-guides. Comme pour les cas d'utilisation et les diagrammes de séquence, certains mots-guides n'ont pas d'interprétation générique.

Par exemple, la combinaison *Action-LATE* correspond à un déclenchement en retard de l'action quand le système change de l'état. Cette situation peut être critique pour des systèmes de contrôle temps réel. Par conséquent, *LATE* fait partie des mots-guides proposés pour l'élément `Action`. En revanche, le mot-guide *PART OF* appliqué à l'élément `Event` ne peut pas être interprété. Nous éliminons donc cette combinaison.

2.4.5 Documents produits par HAZOP-UML

L'application de la méthode HAZOP-UML se présente sous forme de tableau et doit conduire à l'identification des dangers. Nous proposons d'utiliser le tableau comportant les éléments suivants (Figure 2.7) :

1. *Element* : l'élément UML dont la déviation est examinée.
2. *Attribute* : l'attribut considéré.
3. *Guideword* : le mot-guide appliqué.
4. *Deviation* : la déviation résultante de la combinaison du mot-guide et l'attribut.
6. *Use Case effect* : l'effet au niveau du cas d'utilisation.
7. *Real World Effect* : l'effet possible au niveau du système dans son environnement.
8. *Severity* : niveau de gravité du *Real World Effect*.
9. *Possible Causes* : causes possibles de la déviation (logiciel, matériel, humain, etc.).
10. *Integrity Level Requirement* : niveau d'intégrité de sécurité préliminaire. On peut par exemple utiliser les SIL de la norme IEC61508 (2010) cherchant à éviter la déviation avec un niveau de confiance suffisant (ceci mène à l'application des techniques de prévention des fautes ou d'élimination des fautes).

Entity = State-transition diagram		
Attribute	**Guideword**	**Interpretation**
Destination state	Other than	The transition leads to another state than expected
Transition	No/none	The transition is not triggered when intended
	Never	The transition is not triggered because the event never occurs or the condition is never met
Event	No/none	The transition is triggered while the event does not occur
	Other than	**transition not triggered** : the transition is not triggered when the event occurs **transition triggered** : the transition is triggered when another event occurs
Condition	No/none	The condition is not evaluated and can have any value (omission), the transition is triggered
	Other than	**transition not triggered** : the condition is evaluated false whereas it is true (commission), the transition is not triggered **transition triggered** : the condition is evaluated true whereas it is false, the transition is triggered
	As well as	The condition is well evaluated but other unexpected conditions are true, the transition is triggered
	Part of	Only a part of condition is correctly evaluated, the transition is triggered
	Early	The condition is evaluated sooner than required, the transition is triggered
	Late	The condition is evaluated later than required, the transition is triggered
Action	No/none	The transition is not triggered, there is no action
	Other than	The transition is triggered but an action other than intended takes place
	As well as	The transition is triggered, the action as well as an unexpected action take place
	Part of	The transition is triggered but only a part of action takes place
	Early	The transition is triggered but the action takes place sooner than correct synchronization with the environement
	Late	The transition is triggered but the action takes place later than correct synchronization with the environement
	More	The transitions is triggered but the result of the action, if quantifiable, is too high
	Less	The transitions is triggered but the result of the action, if quantifiable, is too low

TABLE 2.3 – Interprétation des mots-guides pour les diagrammes d'état-transition

11. *New Safety Requirements* : si la déviation ne peut pas être évitée, les nouvelles recommandations sont proposées (par exemple, l'ajout des techniques de tolérance aux fautes, nouvelles contraintes de régulation).
12. *Remarks* : explication de l'analyse, recommandations supplémentaires, etc.
13. *Hazard Number* : l'effet réel de la déviation peut être est un danger avec un numéro spécifique, facilitant la lecture des résultats de l'étude et leur référence dans les tables HAZOP.

La réalisation des tables HAZOP permet de synthétiser et d'établir une liste des dangers, comme par exemple dans la Figure 2.8, qui peuvent survenir lors de l'utilisation du système et une liste de recommandations (voir par exemple Figure 2.9), issues de la modélisation et de l'étude de la sécurité. Non seulement les dangers sont identifiés, il est possible de connaître leur popularité, c'est-à-dire le nombre de combinaisons menant à un danger donné.

2.5 Conclusion

La méthode d'identification des dangers présentée dans ce chapitre utilise les techniques bien connues que sont l'analyse du risque HAZOP et la modélisation en language UML. Ces dernières sont largement utilisées pour la conception des systèmes et l'étude du risque. Notre approche se limite à seulement trois types de diagrammes et propose une interprétation des mots-guides associés à chaque type de diagramme. L'avantage majeur de cette méthode est sa facilité d'application dès la première phase du processus du développement du produit quand seules les spécifications de haut niveau sont définies. Nous notons également que les diagrammes de séquences sont parfaitement adaptés pour étudier les systèmes robotiques en interaction avec l'humain. Cette méthode a également l'avantage d'utiliser les mêmes modèles UML que le processus de développement. Bien que la méthode fournisse des résultats importants du point de vue de la gestion du risque, elle ne constitue cependant qu'une seule étape qui est l'identification des dangers. Une spécificité de cette méthode est de n'identifier que les risques opérationnels (c.à.d ceux liés à la vie opérationnelle du système). Il est donc important de compléter cette approche par d'autres techniques comme l'Analyse Préliminaire des Risques ou les Arbres de Défaillances appliqués un peu plus tard dans le processus de développement.

La contribution de cette thèse a été une consolidation des mots-guides pour les diagrammes de séquence et des cas d'utilisation ainsi que le développement des mots-guides pour les diagrammes d'état-transition. Une autre contribution de cette thèse a été de comprendre comment l'approche HAZOP-UML s'intègre dans le processus global de la gestion du risque qui sera présentée dans le Chapitre 4.

Project: MIRAS
HAZOP table number: UC12
Entity: UC12

Use case description

Use case name: Se lever à partir du siège robot

Precondition: - Le patient est assis sur le siège
- Les batteries sont suffisamment chargées pour réaliser cette tâche + déverticalisation

Postcondition: - Le patient est debout face au robot
- Le robot est prêt à déambuler

Invariant: - Les indicateurs physiologiques indiquent un état acceptable
- Base du robot immobile

Date: 29/04/10
Prepared by: Quynh Anh DO HOANG
Revised by:
Approved by:

Line Number	Element	Guideword	Deviation	Use Case Effect	Real World Effect	Severity	Possible Causes	Integrity Level Requirements	New Safety Requirements	Remarks	Hazard Number
UC12 Ligne 1	Le patient est assis sur le siège (precondition)	No/none	Le patient n'est pas assis sur le siège du robot mais le robot pense que si	Le robot démarre la verticalisation	Le robot ne fait pas ce que le patient demande	Néant	Echec logiciel/matériel	Néant	Capteur de pression sur le siège		
UC12 Ligne 2		Other than	ref L1								
UC12 Ligne 3			Le patient est assis sur le siège mais le robot ne le détecte pas	Le robot est dans un mauvais mode	La base du robot peut bouger, provoquant la chute du patient	Sérieuse	Echec logiciel/matériel	SIL2 : SW/HW détection de position du patient	Le robot doit vérifier la position du patient avant d'effectuer la verticalisation		HN6
UC12 Ligne 4		As well as	**N/A**								
UC12 Ligne 5		Part of	Le patient est mal assis	Le robot passe en mode verticalisation du siège avec le patient mal assis	Déséquilibre/Chute du patient	Sérieuse	Echec logiciel/matériel Le robot ne détecte pas car le poids du patient est trop faible	SIL2 : SW/HW détection de position du patient	Ajouter un loquet pour bloquer le siège à l'horizontal Se renseigner sur le poids des patients		HN6
UC12 Ligne 6		Early	**N/A**								
UC12 Ligne 7		Late	**N/A**								

FIGURE 2.7 – Extrait d'une table HAZOP

Hazard Number	Description	Severity
HN4	Chute du patient sans alarme ou avec alarme tardive	Sévère
HN5	Problème physiologique du patient sans alarme ou avec alarme tardive	Sévère
HN6	Chute du patient provoquée par le robot	Sévère
HN7	Défaut de passage en mode sûr	Sévère
HN1	Posture incorrecte du patient pendant l'utilisation du robot	Sérieuse
HN2	Chute du patient pendant l'utilisation du robot	Sérieuse
HN3	Arrêt total du robot pendant l'utilisation (absence d'énergie)	Sérieuse
HN8	Le robot coince le patient	Sérieuse
HN9	Collision entre le robot et le patient	Sérieuse
HN10	Collision entre le robot et une personne autre que le patient	Sérieuse
HN11	Gêne du personnel médical pendant une intervention	Modérée
HN12	Déséquilibre du patient provoqué par le robot	Modérée
HN13	Fatigue du patient	Mineure

FIGURE 2.8 – Exemple d'une liste de dangers - résultat de la méthode HAZOP-UML

Les résultats issus de la méthode HAZOP-UML comme la liste des dangers et la liste des recommandations de sécurité servent à définir le contexte dans lequel un argumentaire de sécurité portant sur le système est construit.

Line Number	Severity	New Safety Requirements	Hazard Number	Origine
1	Modérée	Déterminer les délais de réaction entre la détection et l'activation du mode déverticalisation sur siège du robot. Garantir la réaction dans les délais corrects.	HN12	UC13.SD01 Ligne 29
2	Modérée	Etablir les délais de réactions pour assurer la synchronisation correcte entre la descente des poignées et la déverticalisation du patient	HN13	UC03.SD02 Ligne 57
3	Modérée	La force de l'actionneur ne doit pas "éjecter" le patient.	HN12,HN13	UC12.SD01 Ligne 19,30
4	Modérée	Maintenir le blocage des roues pendant un temps MIN sans dépasser un temps MAX. Vérification de la position de la position du patient avant le déblocage.	HN12,HN13	UC12.SD01 Ligne 62,89
5	Modérée	Etude de la posibilité d'un profil de la déverticalisation sur siège.	HN2	UC13.SD01 Ligne 1,2,3
6	Modérée	Le patient doit pouvoir forcer les vérins en s'appuyant sur le siège pour s'asseoir	HN6,HN12	UC13.SD01 Ligne 18,19,78,85
7	Modérée	Il faut s'assurer de la bonne synchronisation des actions du robot	HN6	UC13.SD01 Ligne 80
8	Sérieuse	Assurer que l'ordre envoyé au moteur correspond à la déverticalisation	HN12	UC03.SD02 Ligne 53
9	Sérieuse	Définir des limites sur la vitesse des poignées.	HN12	UC03.SD02 Ligne 55
10	Sérieuse	Coupler (si possible physiquement) les deux bras pendant la déverticalisation.	HN12	UC03.SD02 Ligne 59
11	Sérieuse	Forcer physiquement les limites en hauteurs des poignées (déterminer les valeurs maximum)	HN8	UC03.SD02 Ligne 91
12	Sérieuse	Validation du profil par opérateur humain.	HN8,HN12	UC03.SD02 Ligne 91,96
13	Sérieuse	Ajouter la précondition "le patient tient le robot par les deux poignées"	HN2	UC03bis Ligne 4 ; UC13 Ligne 4
14	Sérieuse	Le robot doit vérifier la position du patient avant d'effectuer la verticalisation	HN6	UC12 Ligen 3

FIGURE 2.9 – Extrait d'une liste de recommandations mentionnant la source des dangers identifiés - résultat de la méthode HAZOP-UML

Chapitre 3

Confiance dans un argumentaire de sécurité

3.1 Introduction

La question principale qui se pose lorsqu'un argumentaire est construit est la confiance que l'on peut lui accorder. En effet, comment garantir que la logique d'argumentation est valide, que les éléments de contexte ou de preuves sont suffisants, et dignes de confiance ?

L'approche générique consiste à identifier les éléments de l'argumentaire dans lesquels la confiance est limitée ou les incertitudes qui peuvent exister dans un argumentaire, à les modéliser, les estimer, puis exploiter cette estimation pour les traiter en conséquence. Plusieurs travaux regroupés dans la Figure 3.1 suivent cette logique. Cette figure décompose chaque approche en différentes étapes dans l'évaluation de la confiance :

- **Construction d'un argumentaire de sécurité**, souvent sous forme d'une notation graphique comme GSN.
- **Identification des incertitudes** qui peuvent exister dans un argumentaire.
- **Construction d'un argumentaire de confiance** prenant en compte ces incertitudes.
- **Évaluation de la confiance**, effectuée de manière soit qualitative, soit quantitative.
- **Aide à la décision** en se basant sur la confiance.

Chaque approche est présentée plus en détails par la suite.

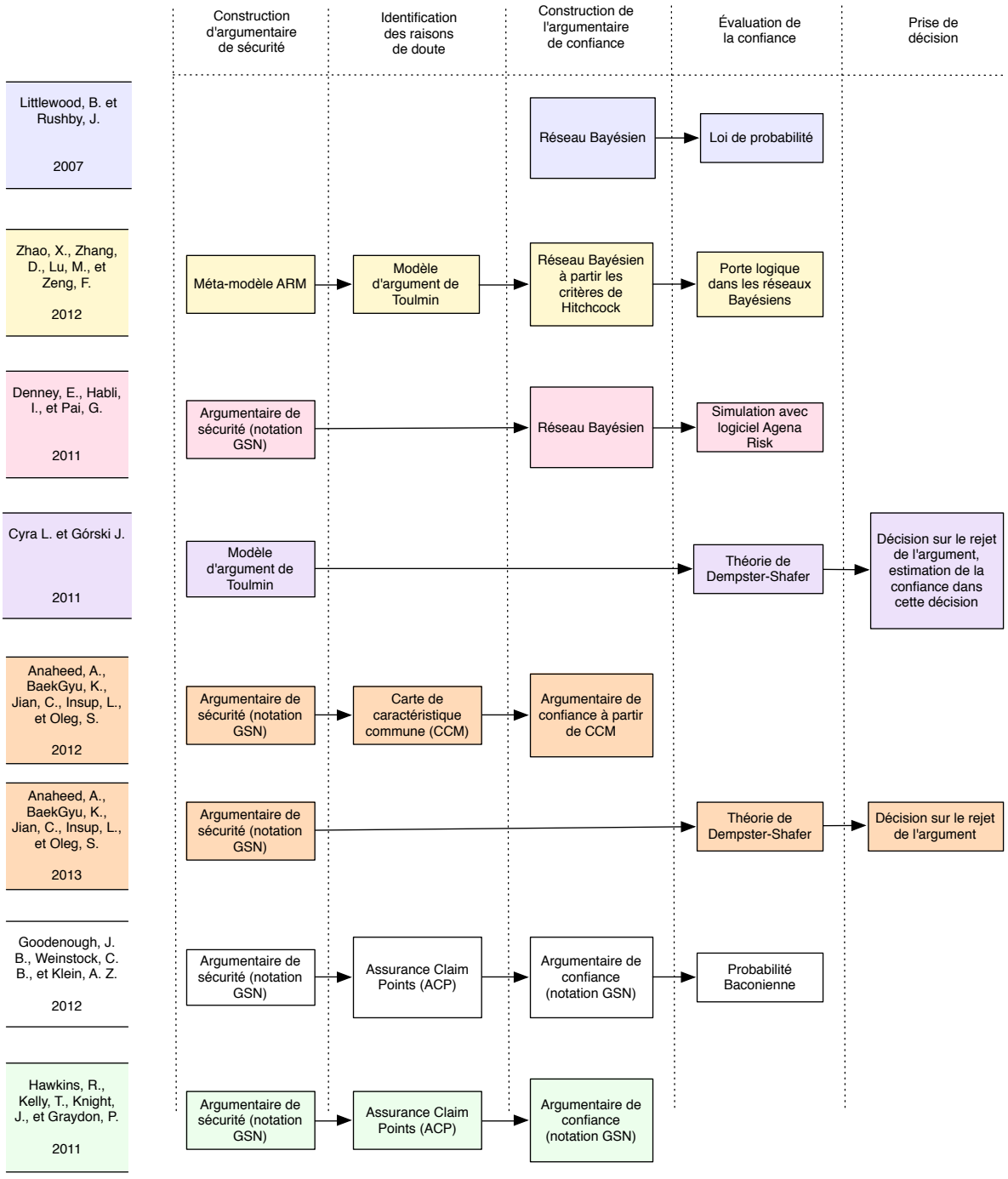


FIGURE 3.1 – Différentes approches sur l'évaluation de la confiance

3.2 Approches existantes pour l'évaluation de la confiance dans un argumentaire

3.2.1 Approches qualitatives

Créateurs de la notation graphique GSN, les chercheurs de l'Université de York prennent conscience du problème de la confiance souvent mêlée dans l'Argumentaire de Sécurité, rendant cette confiance complexe et difficile à évaluer. Une clarification est alors nécessaire pour distinguer l'Argumentaire de Sécurité et la confiance que l'on peut placer dans un tel argumentaire. Ils poursuivent donc leurs recherches dans cette direction, tout en revendiquant le fait de ne pas adopter une approche quantitative.

Hawkins et al. (2011) proposent alors d'ajouter dans un Argumentaire de Sécurité existant des arguments démontrant la confiance : un Argumentaire de Confiance. L'ensemble de ces 2 argumentaires constitue pour eux un Dossier d'Assurance de la Sécurité (*Safety Assurance Case*) comme présenté dans la Figure 3.2. La construction de l'Argumentaire de Confiance est directement liée à l'Argumentaire de Sécurité. L'étape « identification des incertitudes » dans l'Argumentaire de Sécurité consiste à identifier les incertitudes situées à des points précis de l'argumentaire. Ces points sont appelés ACP (*Assurance Claim Point*), et classés selon 3 types comme cela est présenté dans la Figure 3.3 :

— Assertion de type {**sous-revendication - revendication**} ACP1 relève l'incertitude sur l'inférence de l'argument.

— Assertion de type {**contexte - revendication**} ACP2 relève l'incertitude portant sur le contexte (pertinence, digne de confiance) dans lequel la revendication est annoncée.

— Assertion de type {**élément de preuve - revendication**} ACP3 relève l'incertitude portant sur l'élément de preuve (pertinence, digne de confiance).

En réunissant ces assertions, Hawkins et al. (2011) créent un deuxième argumentaire avec la même notation GSN, appelé argumentaire de confiance, dont l'objectif est de montrer que ces trois types d'incertitude sont insignifiantes ou négligeables. Par cette méthode, la confiance dans l'argumentaire original (argumentaire de sécurité) est représentée comme l'absence justifiée de tout doute possible. Les auteurs privilégient l'approche qualitative dans cette démarche, de l'inférence jusqu'aux éléments de preuve. Un exemple générique est présenté dans la Figure 3.4. Bien que cette approche donne des résultats satisfaisants sur un petit exemple, le nombre d'assertions à considérer, et donc d'ACP, est au minimum trois pour un simple argument. Pour un système plus complexe avec un Argumentaire de sécurité de plusieurs niveaux, la construction de l'Argumentaire de confiance pose le problème d'explosion combinatoire.

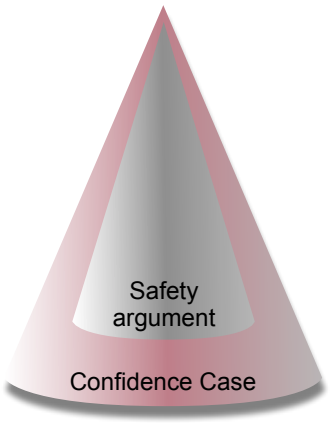


FIGURE 3.2 – Vue globale de l'Argumentaire de Confiance par rapport à l'Argumentaire de Sécurité (Hawkins et al., 2011)

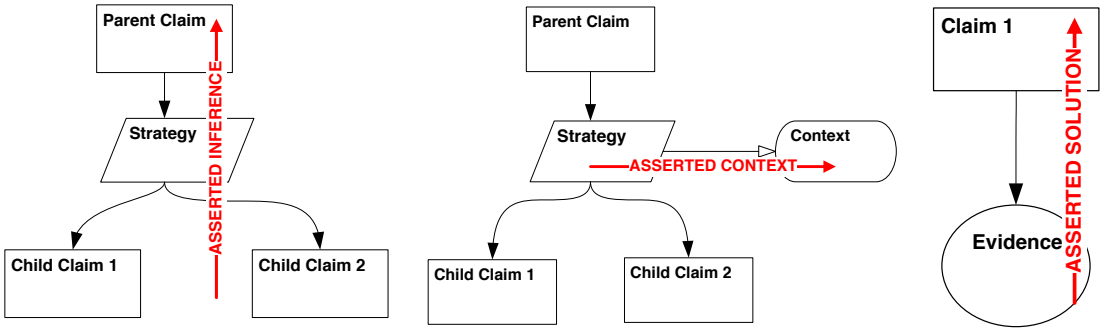


FIGURE 3.3 – Les trois types d'ACP : inférence ACP1 (a), contexte ACP2 (b) et solution ACP3 (c) (Hawkins et al., 2011)

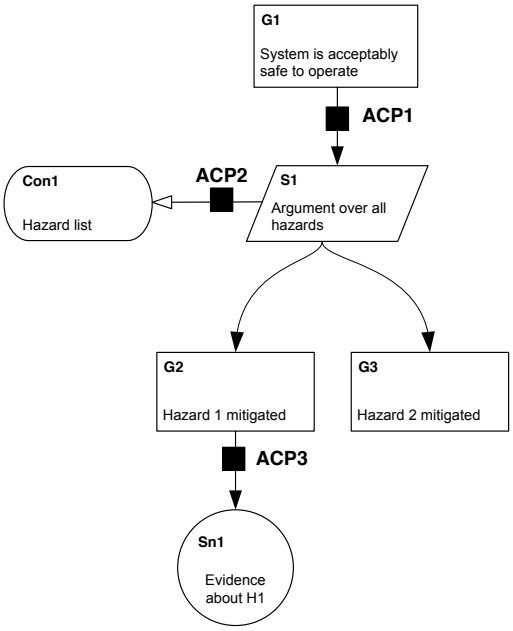


FIGURE 3.4 – Détermination des Assurance Claim Point (ACP) (Hawkins et al., 2011)

Une seconde approche qualitative est proposée par les chercheurs de l'Université de Pennsylvanie. Anaheed et al. (2012) ont repris les travaux de York sur les ACP, mais en proposant une méthode systématique pour identifier les ACP. L'approche commence également par la construction d'un argumentaire de sécurité avec la notation GSN, cependant l'identification des incertitudes est différente : elle fait appel à l'utilisation d'une carte commune de caractéristiques (« *Common Characteristic map* »). La carte proposée cible une seule caractéristique qui est l'aspect « digne de confiance » d'un contexte ou d'un élément de preuve et ne traite donc pas l'inférence de l'argumentaire. Dans le domaine de la sécurité des systèmes, les éléments d'un argumentaire peuvent être classés parmi les catégories suivantes : « *created artifact* », « *provided artifact* », « p*rocess result* », « *the use of mechanism* », « *the use of a tool* ». La carte représente un modèle d'argumentaire (voir Figure 3.5) avec les liens entre ces catégories et les éléments requis pour la caractéristique choisie.

Afin d'évaluer l'aspect « digne de confiance » d'un élément de l'argumentaire de sécurité, il faudra instancier la carte qui se transforme par la suite en une notation GSN. Dans cette structure nouvellement formée, toute branche sans feuille, c'est-à-dire, sans élément de preuve correspond à un défaut d'assurance, une incertitude, un manque de confiance. Il est

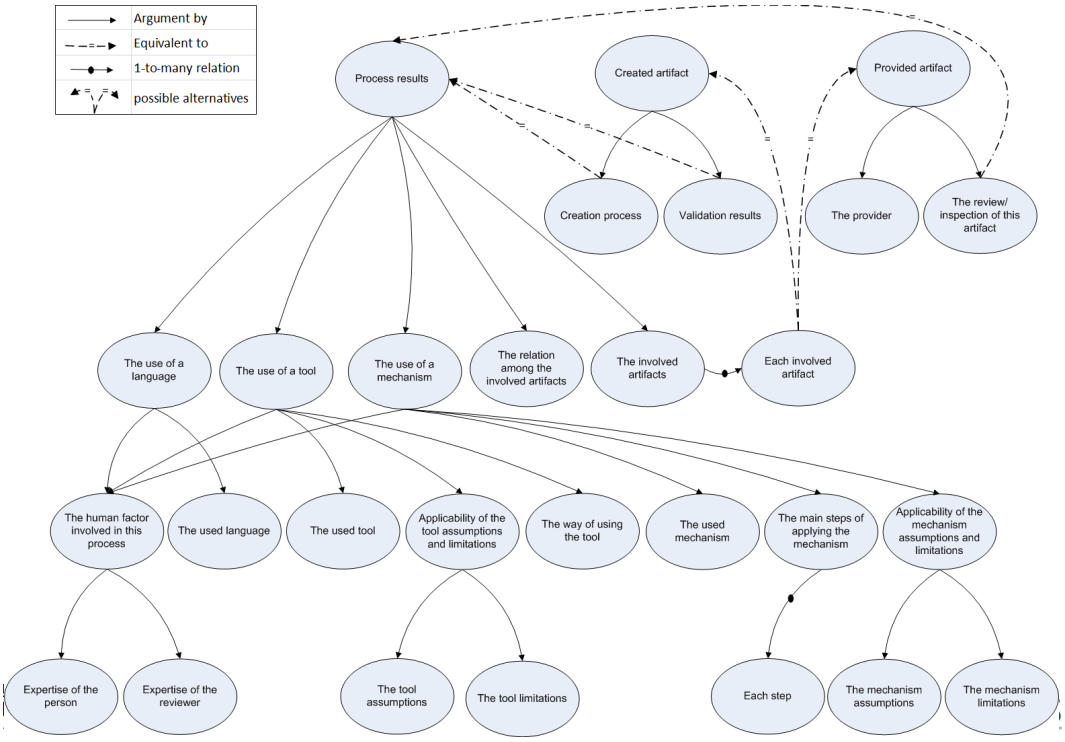


FIGURE 3.5 – Carte commune de caractéristiques

ensuite à la charge de l'analyste de mettre en place les éléments de preuve pour éliminer ces défauts d'assurance.

3.2.2 Approche quantitative : probabilité Baconienne

Ayant identifié le manque de confiance dans un Argumentaire de Sécurité, Goodenough et al. (2013) proposent d'expliquer jusqu'où l'on peut placer sa confiance dans un système. L'approche utilisée est fondée sur l'induction éliminatoire afin de justifier le niveau de confiance d'un Argumentaire de Sécurité.

La confiance, d'après ce point de vue, est liée au nombre de raisons de doute (« *defeater* ») identifiées et éliminées. Si n raisons sont identifiées parmi lesquelles seules i raisons sont éliminées (par un argumentaire ou un élément de preuve), la confiance s'exprime en probabilité Baconienne comme $i|n$ (lire « i sur n »). Ainsi, «ne pas avoir confiance» se traduit par le fait qu'aucune raison de doute, bien qu'identifiée, ne soit éliminée : $0|n$. Par analogie, « avoir totalement confiance » signifie que toute raison de doute est identifiée et éliminée : $n|n$. Notons cette différence avec la probabilité Pascalienne dans laquelle une

probabilité de 1 signifie « la revendication est toujours vraie » et 0 « la revendication est toujours fausse ».

Pour toute revendication révisable ou sujette à caution, il existe les raisons de doute (*defeater* en anglais). Ces raisons sont classées en 3 catégories :

1. « *a rebutting defeater* » fournit un contre exemple à la revendication,
2. « *an undermining defeater* » évoque des soupçons quant à la validité des éléments de preuve,
3. « *an undercutting defeater* » précise les circonstances dans lesquelles la conclusion est remise en cause même si les prémisses ou l'inférence restent valides.

Ainsi, identifier les raisons de doute et les éliminer (en démontrant qu'elles ne peuvent pas exister) est l'idée fondamentale de cette approche. Bien qu'il soit utile de connaître le nombre de raisons de doute non éliminées, il est cependant difficile d'utiliser ce chiffre pour prendre une décision. Ce nombre ne prend pas en compte l'importance ou le poids d'une raison par rapport à une autre.

3.2.3 Approche quantitative : réseaux Bayésiens

Bloomfield et Littlewood (2003); Littlewood et Rushby (2012), ont étudié également le problème de confiance dans un type d'argumentaire particulier : la diversification, l'exemple considéré étant un système qui repose sur deux architectures redondantes (une complexe mais moyennement fiable, une simple mais très fiable) pour assurer sa fonction. Le but de ces recherches est de mesurer le gain de confiance par rapport au même système utilisant une seule architecture. Ils en concluent qu'un gain de confiance est possible mais il est difficile d'estimer ce gain sans les hypothèses simplificatrices comme par exemple l'indépendance conditionnelle des événements étudiés. Une autre étude de Littlewood et Wright (2007) portant sur la même architecture propose une transformation en réseau Bayésien afin d'utiliser les probabilités conditionnelles et de conduire des études analytiques. Cependant, même avec les simplifications faites, le réseau Bayésien obtenu s'avère complexe pour comprendre et interpréter les résultats, montrant certains cas où la confiance peut diminuer au lieu d'augmenter.

Dans le même objectif d'estimer la confiance d'un Argumentaire de sécurité, Denney et al. (2011) proposent une approche quantitative en passant par un réseau Bayésien. En se basant sur la structure GSN de l'Argumentaire de Sécurité, les sources d'incertitude, aléatoires (mesurables par des statistiques) ou épistémiques (déterminées par un avis subjectif de l'expert), sont identifiées. Bien que l'identification des incertitudes ne soit pas présentée en détail, ces dernières sont traitées comme des variables aléatoires discrètes ayant des relations causales entre elles. Elles forment ainsi un réseau Bayésien comme celui de la

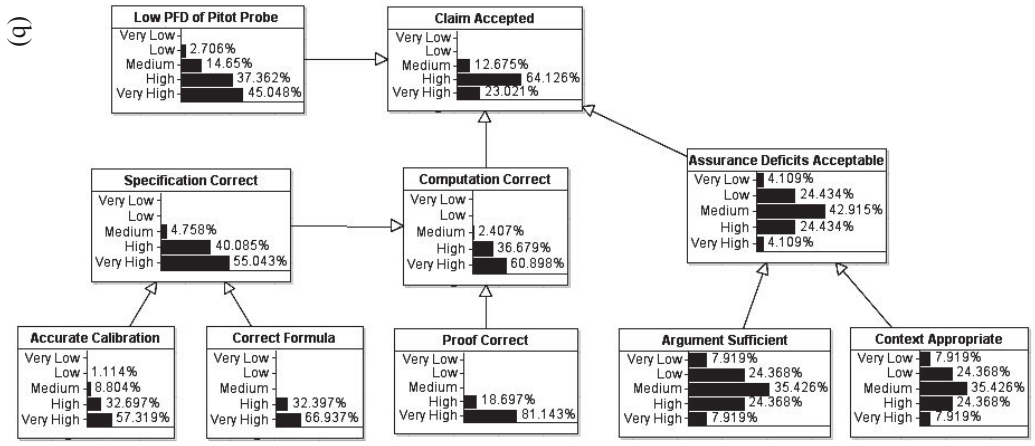


FIGURE 3.6 – Réseau Bayésien des incertitudes d'un Argumentaire de Sécurité présenté par de (Denney et al., 2011)

Figure 3.6, permettant l'estimation de la confiance dans les nœuds dont la confiance est non observable.

Dans cet exemple, chacun des nœuds du réseau Bayésien se compose de 5 états (Very Low ; Low ; Medium ; High ; Very High), qui correspondent à des niveaux de vérité dans la revendication (le nœud). Ainsi, pour le nœud «Specification is correct», la distribution sur les 5 valeurs peut être interprétée comme : 55% des experts pensent que le niveau de vérité est *very high*, 40% pensent que le niveau est *high* et 4,7% pensent que c'est *medium*. Les auteurs utilisent les travaux de (Fenton et Neil, 2012) et se basent sur la loi normale tronquée (un exemple est donné Figure 3.7) pour la distribution de la confiance sur ces 5 valeurs. Ils utilisent cette loi normale, pour passer des 5 valeurs discrètes à des valeurs continues en attribuant des intervalles continus à chaque valeur discrète, ([0 - 0,2[; [0,2 - 0,4[; [0,4 - 0,6[; [0,6 - 0,8[; [0,8 - 1]), et faire les calculs de propagation de la confiance. Ces calculs se basent sur la moyenne μ et l'écart-type σ de cette loi Normale, et des formules intégrant le poids de chaque élément de preuve du GSN.

Les paramètres de la loi Normale Tronquée des feuilles du GSN (*Solution*) sont fournis soit à partir des mesures statistiques soit à partir des avis d'experts qui décideront de manière appropriée la distribution du niveau de vérité. Ainsi, en partant des nœuds feuilles, les niveaux se propagent jusqu'au nœud principal *Claim Accepted*. Cependant, les règles de transformation des nœuds du GSN en réseau Bayésien ne sont pas précisées et les formules de propagation sont purement empiriques. Il manque également une définition précise de ce que les auteurs entendent par confiance. On peut l'interpréter ici comme

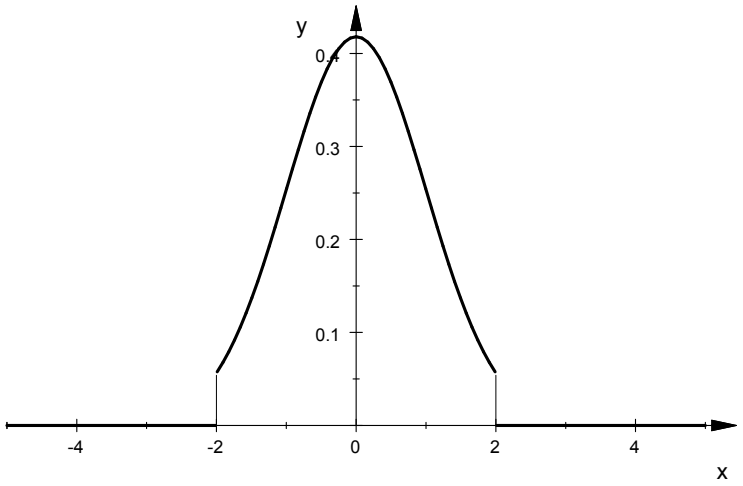


FIGURE 3.7 – Loi Normale Tronquée en 2

la variance de la loi normale, associée à un état. Ainsi, une faible variance (loi normale très étroite) avec une moyenne entre *high* et *very high* devrait être considéré comme un nœud «de confiance». Malgré ces critiques, leur travail était précurseur dans l'évaluation quantitative de la confiance d'un argumentaire, et l'utilisation d'un outil de réseau Bayésien pour propager la confiance est un avantage important de cette approche. Nous avons étudié en détail les possibilités de cette utilisation des travaux de Fenton dans le rapport technique disponible en ligne : (Do Hoang et al., 2013).

Les travaux présentés dans Zhao et al. (2012) se sont aussi intéressés au problème de l'évaluation de la confiance dans un argumentaire afin de s'en servir de base pour prendre des décisions. Leur approche comporte également deux parties : un argumentaire de sécurité et un argumentaire de confiance quantitatif. Les auteurs traitent un argumentaire de sécurité, que ce soit de type GSN ou CAE comme une instance d'un méta-modèle ARM (*Argumentation Metamodel*) proposé par l' OMG-ARM (2013). Cela permet de convertir chaque argument (composant d'une revendication, des hypothèses, des éléments de preuve,etc.) en un modèle d'argument de Toulmin (voir Section 1.6). Chacun des arguments dans le modèle de Toulmin (équivalent à une revendication du modèle ARM) est ensuite transformé en un nœud d'un réseau Bayésien, supporté par les quatre critères de Hitchcock (2005) (prémisse justifiée, information adéquate, justification applicable, hypothèse justifiée) ou par la mesure 5M1E (considérant les facteurs : Humain, Machine, Méthode, Matériel, Mesure et Environnement). Comme la revendication d'un argument devient un élément de preuve d'un autre argument, ces réseaux Bayésiens se connectent entre eux et forment

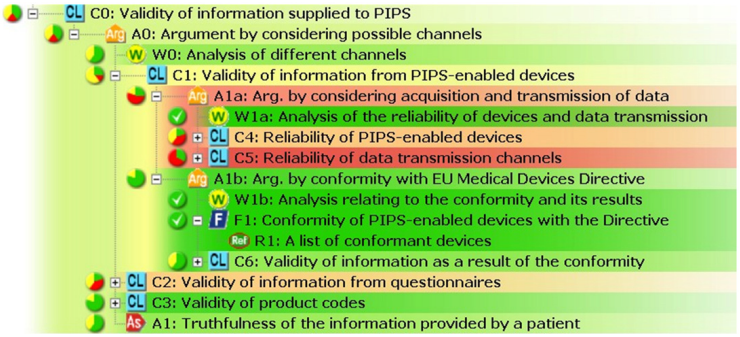


FIGURE 3.8 – Le visuel de l'évaluation de la confiance dans un argumentaire (Cyra et Górski, 2011)

ainsi un réseau global qui représente tout l'argumentaire du départ. Dans ce réseau Bayésien, la confiance dans les nœuds élémentaires prend des valeurs binaires (les avis d'expert ou les résultats de test). Cette confiance se propage vers des nœuds supérieurs à l'aide des fonctions de type *NOISY-OR* ou *NOISY-AND* jusqu'au nœud principal qui représente la confiance dans la revendication de l'Argumentaire. L'approche proposée traite le modèle argumentaire à partir du méta-modèle ARM qui est plus général que GSN ou CAE et donc applicable à ces deux derniers également. Cependant, le réseau Bayésien obtenu rencontre des problèmes d'explosion combinatoire en cas d'un grand argumentaire et l'attribution des probabilités repose sur l'expertise des experts.

3.2.4 Approche quantitative : théorie de Dempster-Shafer

La théorie de Dempster–Shafer présentée en Section 1.5 a été utilisée pour estimer quantitativement la confiance. Cyra et Górski (2011) proposent non seulement une méthode pour passer en revue l'argumentaire de sécurité du départ mais aussi d'émettre une décision et y placer une confiance. La révision des éléments de l'argumentaire est formalisée en se basant sur cette théorie. L'approche calcule alors 2 critères : la décision d'accepter l'argument et la confiance dans cette décision.

Une structure argumentaire est souvent difficile à analyser pour les non experts. Pour la rendre plus accessible, les auteurs proposent une nouvelle méthode pour évaluer les arguments et communiquer les résultats avec d'autres collaborateurs, appelée VAA (*Visual Assessment of Argument*). Chaque élément de l'argumentaire se voit attribuer une couleur représentant la confiance en cet élément (Figure 3.8).

La méthode consiste en quatre étapes :

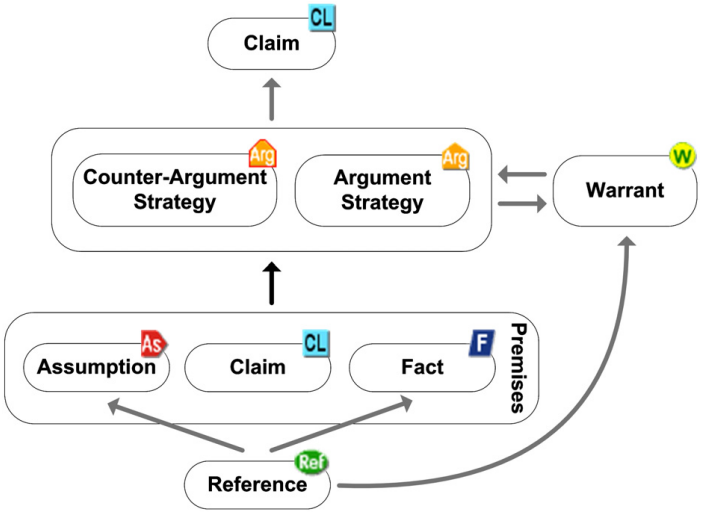


FIGURE 3.9 – Le visuel de l'évaluation de la confiance dans un argumentaire (Cyra et Górski, 2011)

— Étape 1 : Présentation de l'argumentaire qui est basé sur le modèle d'argument de Toulmin (Figure 3.9).
— Étape 2 : Détermination du type d'argument.
— Étape 3 : Évaluation des fonctions de croyance et de la plausibilité du raisonnement plausible de l'argument résultant, à partir de ceux de ses justifications, ses prémisses et ses hypothèses. La formule de calcul varie en fonction du type d'argument déterminé en étape 2.
— Étape 4 : Utilisation d'un barème d'évaluation Décision/Confiance pour estimer la confiance dans l'argument (Figure 3.11).

La détermination du type d'argument de l'étape 2 distingue les différents types présentés dans la Figure 3.10. La classification est fondée sur leur nature afin de les traiter de manière appropriée. La méthode distingue deux principaux types d'argument, divisés en sous-types :

Type 1 : argument dont la falsification d'une seule prémisse mène à la réfutation de la conclusion car aucune inférence ne peut en découler.
— Type 1.1 : la falsification d'une prémisse rejette la conclusion (similaire à une condition nécessaire et suffisante).
— Type 1.2 : la falsification d'une prémisse rejette l'inférence (similaire à une condition suffisante).

Type 2 : la falsification d'une prémisse diminue le support pour la conclusion.

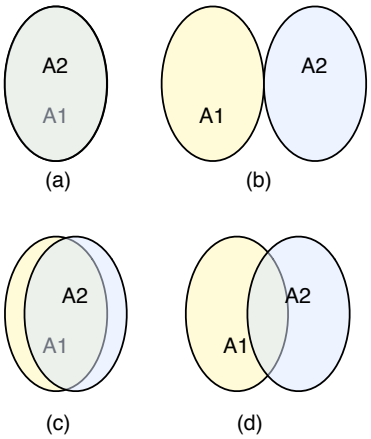


FIGURE 3.10 – Les différents types d'arguments de Type 2 : (a) Argument alternatif, (b) Argument complémentaire, (c) et (d) Argument mixte, tirés de (Cyra et Górski, 2011)

- Type 2.1 : chacune des prémisses supporte une partie de la conclusion (argument complémentaire).
- Type 2.2 : les prémisses s'utilisent dans quelques arguments qui supportent la conclusion de manière indépendante (argument alternatif).
- Type 2.3 : chacune des prémisses supporte une partie, non nécessairement disjointe, de la conclusion (argument mixte).

Une deuxième partie de la méthode utilise la théorie de Dempster-Shafer pour évaluer la confiance dans la décision d'acceptabilité ou non d'un argumentaire. Pour chaque assertion s, la théorie de Dempster-Shafer fait appel à une fonction de croyance et de la plausibilité du raisonnement, comprises entre 0 et 1 (voir Section 1.5). Dans cette méthode, elles sont définies de manière formelle :

$Bel(s) \in [0,1]$ est la fonction de croyance représentant le degré de croyance qui supporte directement s. $Pl(s) \in [0,1]$ est la plausibilité du raisonnement représentant la borne supérieure que la confiance dans s peut atteindre si les nouveaux éléments de preuve sont ajoutés.

Une fonction de décision $Dec(s)$ exprime le rapport de la croyance et la confiance dans l'assertion s, sans chercher à distinguer si elle est à accepter ou rejeter.

$$Conf(s) = Bel(s) + 1 - Pl(s) \text{ avec } Conf(s) \in [0,1]$$

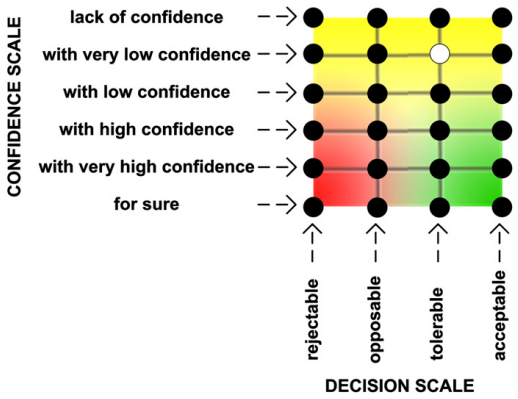


FIGURE 3.11 – Le barème d'évaluation : décision et confiance (Cyra et Górski, 2011)

$$Dec(s) = \begin{cases} \frac{Bel(s)}{Bel(s)+1-Pl(s)} & Bel(s)+1-Pl(s) \neq 0 \\ 1 & Bel(s)+1-Pl(s) = 0 \end{cases} \quad \text{avec } Dec(s) \in [0,1]$$

Les fonctions $Conf(s)$ et $Dec(s)$ permettent de placer l'assertion sur un barème d'évaluation tel que présenté dans la Figure 3.11, exprimant la décision (accepter/rejeter) concernant s et la confiance dans cette décision.

Les deux échelles décisions/confiance que présente la Figure 3.11 situent l'assertion s en fonction de la décision (acceptation/rejet) et la confiance dans cette décision.

Un deuxième volet des travaux de Anaheed et al. (2013) ne s'appuie pas sur les résultats précédents des auteurs, mais explore l'utilisation de la théorie de Dempster-Shafer en s'appuyant sur les travaux de Cyra et Górski (2011). Leur objectif est également une étude quantitative de la suffisance dans un argumentaire, représenté par la suffisance du nœud principal. Pour un nœud de l'argumentaire de sécurité exprimé en GSN, les auteurs attribuent 2 valeurs possibles : insuffisant, ou suffisant. Dans la théorie de D-S cela revient à avoir un discernement $\Omega = \{Suffisant, Insuffisant\}$, et $2^{\Omega} = \{\emptyset, \{Suffisant\}, \{Insuffisant\}, \{Suffisant, Insuffisant\}\}$ l'ensemble des parties de Ω. Chaque partie se voit attribuer une valeur m comprise dans l'intervalle $[0;1]$ représentant « le degré de croyance » ou « la masse ». L'approche sert à estimer la masse m du nœud principal, c'est-à-dire, à calculer son degré de croyance à partir des degrés de croyance des autres nœuds qui le supportent. Les auteurs distinguent quatre formules de calcul correspondant aux quatre types d'argument suivants :

Alternatif : cas où plusieurs éléments de preuve soutiennent indépendamment la revendication. Chaque élément soutient la totalité de la revendication.

Disjoint : cas où plusieurs éléments de preuve soutiennent la revendication de manière complémentaire. Chaque élément soutient une partie différente de la revendication.

Chevauchement : cas où plusieurs éléments de preuve soutiennent la revendication de manière complémentaire mais avec une partie de la revendication en commun.

Endiguement : cas où plusieurs éléments de preuve soutiennent la revendication mais l'un recouvre totalement l'autre élément de preuve.

Bien que similaire aux travaux de Cyra et Górski (2011), cette approche distingue les types d'arguments mixtes (chevauchement ou endiguement) des types d'arguments précédents étudiés (alternatif ou disjoint/complémentaire) pour couvrir plus de possibilités. Cependant, le degré de croyance que représente la masse m dans la « suffisance » d'un nœud dans l'argument ne permet pas de représenter l'incertitude dans l'inférence entre le noeud et ses descendants (ou ancêtres). Il nous semble également que la notion de «suffisant/insuffisant» est ambigüe pour estimer la confiance. En effet, on peut imaginer par exemple, qu'effectuer des tests serait «suffisant» pour justifier de la sécurité d'un système, mais que la réalisation de ces tests a été effectuée dans de mauvaises conditions, et qu'on a donc une faible confiance dans les conclusions de ces tests.

3.3 Vers une nouvelle approche d'évaluation de la confiance d'un argumentaire

Comme présenté dans les travaux précédents, la confiance dans un argumentaire peut être évaluée de manière qualitative ou quantitative. Dans cette section, nous proposons une contribution à l'évaluation quantitative de la confiance dans un argumentaire.

3.3.1 Approche générale

À la différence des travaux précédents, notre objectif n'est pas d'interpréter une valeur de la confiance globale pour prendre une décision mais une analyse de la sensibilité du modèle d'argumentaire, c.à.d. d'identifier pour un argumentaire donné, les éléments ayant le plus d'impact sur la confiance globale afin de consolider un argumentaire.

A partir d'un argumentaire exprimé en GSN, nous souhaitons donc construire un modèle de confiance, avec une estimation quantitative et des règles de calcul automatiques. L'objectif est donc de transformer un argumentaire en réseau de confiance, puis d'effectuer une étude de sensibilité. Le processus que nous proposons suit les étapes suivantes :

— Représenter l'argumentaire en notation graphique GSN.

— Transformer le graphe GSN en un réseau de confiance en suivant des règles de transformation.
— Attribuer la confiance à chacun des nœuds et calculer la confiance globale de l'argumentaire.
— Faire varier les confiances dans les éléments de base afin d'observer leur influence sur la confiance globale de l'argumentaire.

Le processus proposé est représenté sur la Figure 3.12. Les outils proposés pour la modélisation du réseau de confiance (réseau bayésien) et l'analyse de sensibilité (Graphe de Tornado) seront justifiés et présentés dans les sections suivantes. Une telle approche permet de réaliser, pour un argumentaire donné, les études suivantes :

— Consolider la confiance ou l'indépendance vis-à-vis des éléments les plus influents.
— Analyser la contribution (la sensibilité) de plusieurs éléments à la confiance.
— Comparer la confiance entre plusieurs argumentaires pour le même système, chaque argumentaire représente une version différente du système.
— Comparer la confiance dans différents systèmes.
— Comparer, pour un même système, la confiance obtenue par différents argumentaires de sécurité (différentes stratégies ou différentes architectures)

Les actions possibles sont alors de concentrer les efforts pour réduire les points "sensibles" de l'argumentaire, de les compléter, de les remplacer, etc. Dans cette approche, l'objectif est alors d'obtenir une confiance la plus forte possible, en ayant réduit les incertitudes liées aux points sensibles de l'argumentaire. On peut comparer cette approche avec le principe de gestion du risque ALARP (As Low As Reasonably Practicable), c.à.d, d'augmenter la confiance jusqu'au point où le gain de confiance ne justifie pas les coûts engendrés par la modification de l'argumentaire.

3.3.2 Mesure de confiance

Il est dans un premier temps fondamental de définir ce que nous souhaitons mesurer : la confiance. En effet, comme présenté dans la section précédente, très peu d'articles donnent une définition précise de ce qui est considéré comme la confiance. D'après les dictionnaires de langue Française, la confiance est définie comme le «sentiment d'assurance que l'on a dans quelqu'un ou quelque chose» (Rey, 1991). Ce sentiment d'assurance peut être quantifié par une probabilité. Cependant, Gacogne (2003) propose d'utiliser la notion de confiance comme un chapeau pour différentes théories comme la théorie des probabilités ou celle de Dempster-Shafer. Une *probabilité* serait une des mesures de la confiance dans un événement, tout comme la *croyance* dans la théorie de Dempster-Shafer (présentée dans la Section 1.5). A partir d'une définition d'axiomes simples pour la confiance, l'auteur liste pour chaque

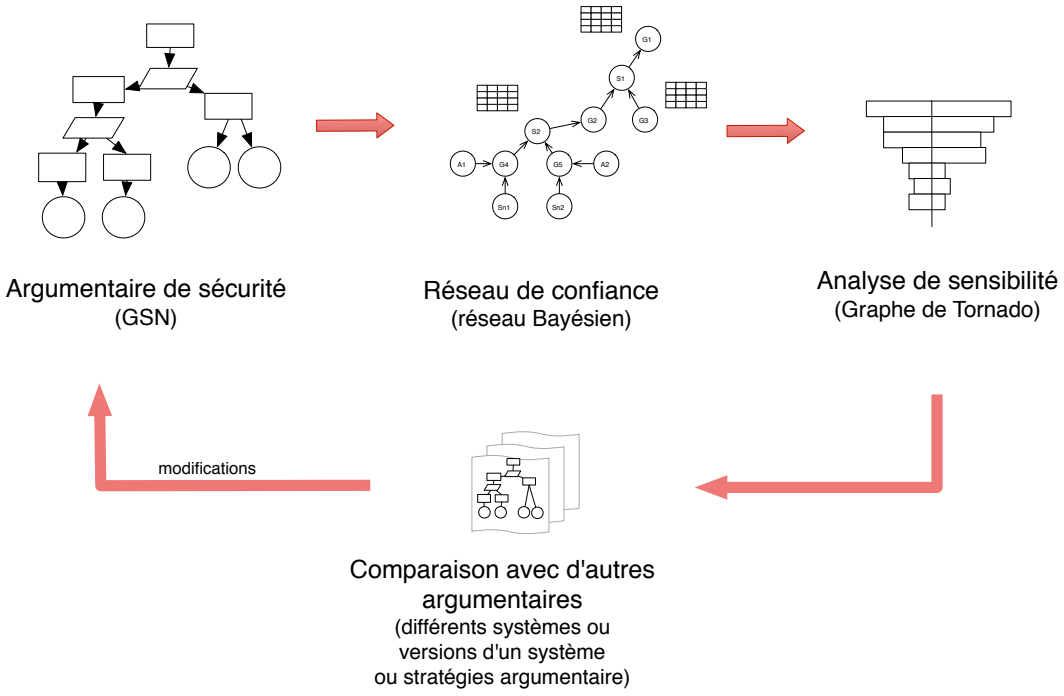


FIGURE 3.12 – Vue générale de l'approche proposée

théorie les axiomes supplémentaires. Parmi les axiomes de base de la définition de confiance, on retrouve l'axiome suivant : Si on note par g une mesure de la confiance et A, B deux événements, on a :

$$g(A \cup B) \geq max(g(A), g(B)) \tag{3.1}$$

$$g(A \cap B) \leq min(g(A), g(B)) \tag{3.2}$$

À titre d'exemple, pour la probabilité, qui est une mesure de confiance selon l'auteur, l'axiome suivant est ajouté : $A \cap B = \emptyset \Rightarrow P(A \bigcup B) = P(A) + P(B) = g(A \cup B)$ ce qui vérifie bien l'équation 3.1. Cette approche présente l'intérêt de faire explicitement apparaître le concept d'incertitude, que l'on retrouve dans les argumentaires de sécurité. Comme Cyra et Górski (2011), nous avons donc choisi de nous baser sur cette théorie. Cependant, contrairement à ces auteurs, nous considérons que dans le contexte de la construction d'un argumentaire, tous les éléments de preuve d'un argumentaire sont observés. Nous écarterons donc le fait qu'ils soient certainement faux. Par exemple, si on met dans un argumentaire un élément de preuve comme « Les tests sont concluants », pour exprimer que les tests ont été faits, et ne révèlent aucune faute, alors on peut exprimer la confiance que l'on a dans l'événement « Les tests sont concluants ». Cette confiance va varier selon la ou les personnes ayant réalisé les tests, les outils utilisés, etc. Sans affiner ces facteurs de confiance, on souhaite donc assigner une valeur de confiance à cet élément, qui est un fait observé. Dans notre étude nous considèrerons donc que nous sommes dans un cas particulier de Dempster-Shafer, où la valeur de *disbelief* est nulle. Nous nous plaçons dans le cadre de ce que la théorie D-S nomme les fonctions à support simple (voir l'équation 1.7).

Ainsi, la proposition A est décrite uniquement avec des masses pour la croyance et l'incertitude comme présenté sur la Figure 3.13. On retrouve comme pour la théorie des probabilités une seule grandeur sur un intervalle de 1, mais l'interprétation est différente car nous aurons :

$$\begin{cases} m(A) = Bel(A) = g(A) \in [0,1] : & confiance \\ m(A, \overline{A}) = 1 - g(A) \in [0,1] \ : & incertitude \\ m(\overline{A}) = 0 \end{cases} \tag{3.3}$$

Nous proposons la notation suivante : $g(A)$ désigne la confiance que nous plaçons dans un élément X (un élément de preuve, un contexte, une revendication...). Il est important de bien séparer la notion de confiance et de probabilité d'occurrence d'un élément de l'argumentaire. La confiance $g(B)$ dans un élément B est différente de la probabilité d'occurrence $p(B)$ de l'événement B. Par exemple, un test effectué démontre que le système X est apte à

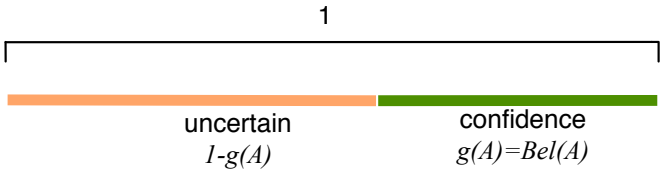


FIGURE 3.13 – Confiance et incertitude

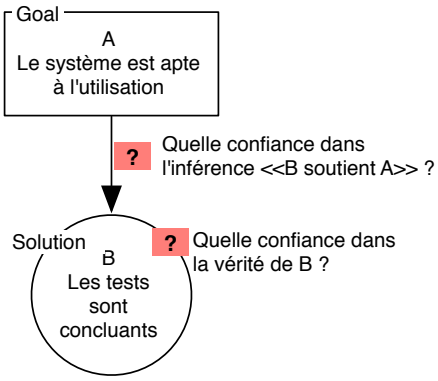


FIGURE 3.14 – Une inférence simple en GSN et les points de confiance associés

l'usage. La confiance *g(X)* dans le système représente à la fois la confiance dans le résultat de test et dans la pertinence du test pour démontrer la sécurité du système. Cependant, la probabilité d'occurrence *p(X)* de l'événement « le système est apte à l'usage » représente le fait que le système est effectivement apte à l'usage. Si un mauvais test a été effectué (avec un bon résultat) sans que l'utilisateur ne se rende compte, la confiance *g(X)* de l'utilisateur dans le système peut être élevée mais en réalité la probabilité *p(X)* que le système soit apte est relativement faible (à cause du mauvais test).

Dans le cadre d'un argumentaire, on est confronté à des assertions du type B soutient A, comme par exemple : B= « Les tests sont concluants », et A=« Le système est apte à l'utilisation ». La notion de confiance établie précédemment va donc s'appliquer à l'élément B, pour exprimer le degré de crédibilité de B, mais également au fait que l'inférence de B vers A est crédible ou non. A la différence des travaux menés à l'université de York (Hawkins et al., 2011), qui propose trois types d'incertitudes (voir les ACP Figure 3.3), nous ne considèrerons dans notre étude que deux types d'incertitude présentées Figure 3.14 :

— incertitude dans l'élément B
— incertitude dans l'inférence B menant vers A

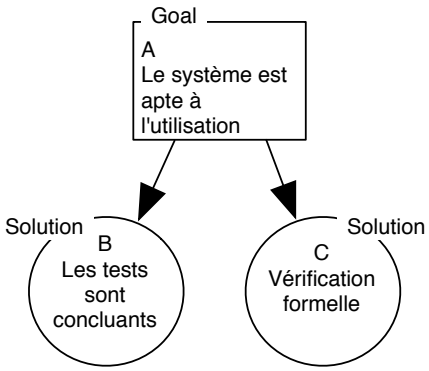


FIGURE 3.15 – Exemple d'argumentaire alternatif

Le fait de considérer uniquement ces deux types d'incertitudes, et notamment l'incertitude sur l'inférence, nous a détourné des travaux basés sur D-S, où il est en général possible de combiner ces croyances, mais pas de faire apparaître explicitement une confiance sur cette combinaison. Nous avons également noté que la notion du poids d'un prémisse dans une combinaison est également peu lisible dans ces travaux. Comme nous le verrons dans les sections suivantes, nous utiliserons pour cela un réseau de confiance, similaire à un réseau Bayésien, mais en gardant la définition de l'incertitude de la théorie de D-S.

3.3.3 Types d'argument

Sur la base de cette définition de la confiance, nous considérons que tout type d'argumentaire prendra compte explicitement les deux types d'incertitude et les points de confiance associés distingués dans la Figure 3.14. Dans le cas d'un argumentaire composé de plusieurs prémisses, comme par exemple : B et C soutiennent A, ces deux types de confiance seront également présents. En revanche, selon le type d'inférence, le calcul de la confiance dans une inférence sera différent. Prenons le cas de l'argumentaire diversifié étudié par Littlewood et Wright (2007), où l'objectif de haut niveau « Le système est apte à l'utilisation (A) » , est supporté par deux éléments de preuve : « Les tests sont concluants (B) » et « Le système est vérifié formellement (C) » . Dans leur article, les auteurs considèrent finalement que ces deux éléments sont dépendants (notamment de la spécification du système), mais sur le principe, les deux arguments amènent chacun une confiance supplémentaire dans A. C'est la même idée que l'on retrouve dans l'article de Cyra et Górski (2011), sous la dénomination «argument alternatif» : même si l'un des arguments s'avère trop incertain, le fait qu'il y ait un autre argument amène tout de même un certain niveau de confiance dans A (comme dans l'exemple de la Figure 3.15).

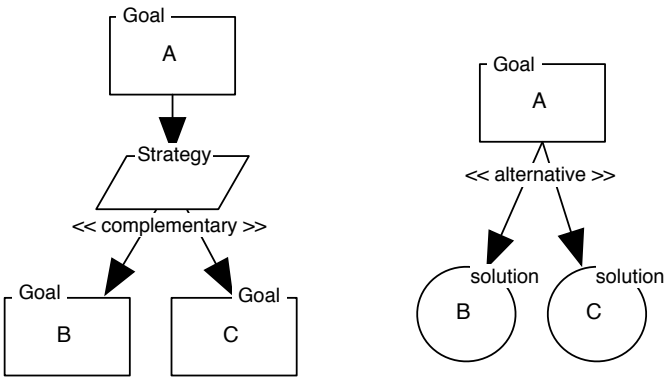


FIGURE 3.16 – Exemple d'utilisation de la notation pour une argumentation alternative ou complémentaire

Une autre forme d'argumentation peut être identifiée en prenant le cas du patron GSN « Hazard Avoidance Pattern » proposé par Kelly et McDermid (1997) et qui a été présenté dans la Section 1.6, où le but de haut niveau G1 est décomposé en sous objectifs selon la stratégie S1 : « Tous les dangers potentiels ont été traités » . Dans ce cas, les n dangers doivent avoir été identifiés (contexte), et chaque sous-objectif atteint («Danger X traité») pour pouvoir justifier G1. Si une confiance très faible est attribuée au traitement d'un danger présenté, cela doit impacter fortement la confiance dans l'objectif G1.

Cyra et Górski (2011) proposent de nommer ce type d'argumentaire « complémentaire » . Ils ont également d'autres types d'argumentaires (notamment les arguments nécessaires et suffisants, etc.), mais nous proposons dans ce manuscrit de simplifier l'approche et de ne considérer que deux types de combinaisons d'argumentation, ainsi qu'une annotation sur les GSN comme présenté dans la Figure 3.16 :

1. alternatif (noté *«alternative»*) : les prémisses supportent la conclusion, si l'une des prémisses est de faible confiance, cela n'annule pas la confiance dans la conclusion.
2. complémentaire (noté *«complementary»*) : si l'une des prémisses est de faible confiance, cela abaissera la confiance dans la conclusion.

Ces deux types d'argumentation sont détaillés dans les sections suivantes 3.3.5 et 3.3.6.

3.3.4 Argumentation simple

La première argumentation à considérer est l'inférence simple, du type « A est soutenu par B » . Cette inférence simple peut être utilisée dans un GSN comme présenté dans la Figure 3.17, soit dans le cadre d'une solution à un objectif, ou même pour un sous-objectif

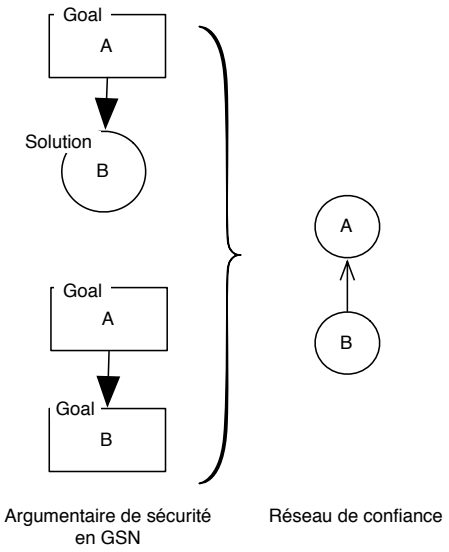


FIGURE 3.17 – Transformation GSN vers un réseau de confiance pour l'inférence simple

soutenant un autre objectif. Dans ce dernier cas, il s'agit souvent d'une reformulation de l'objectif de plus haut niveau. À titre d'exemple, prenons le cas où l'on souhaite démontrer qu'un système est sûr (objectif de haut niveau), une façon de faire (pour des systèmes simples), est de décomposer cet objectif en un objectif plus simple du type «Application de la norme X». Dans ce cas, un document décrivant comment la norme a été suivie est le seul élément de preuve (*Solution*). Le GSN serait alors «G1 : Système sûr» soutenu par « G2 : La conception a suivi la norme X » soutenu par la solution « Sn1 : *Checklist* de suivi de norme X » . On propose de transformer ces modèles GSN en un réseau de confiance, équivalent mathématiquement à un réseau Bayésien. En effet, comme expliqué précédemment, le fait d'avoir simplifié l'utilisation de la théorie de D-S à des fonctions à support simple ramène à un cas de calcul équivalent aux réseaux Bayésiens. En revanche, l'interprétation sera différente.

Le tableau ci dessous, spécifie 2 cas : aucune confiance en B, $g(B) = 0$, et confiance maximum en B, $g(B) = 1$.

g(B)	0	1
g(A)	0	p

Nous proposons de compléter la ligne $g(A)$ de ce tableau, en évaluant la confiance que l'on a dans A selon les deux cas précédents. De manière évidente, il s'avère que lorsqu'on

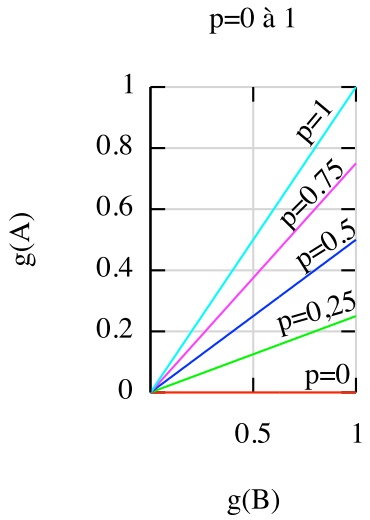


FIGURE 3.18 – Inférence Simple, $g(A)$ en fonction de $g(B)$, pour p variant de 0 à 1

a pleinement confiance en B ($g(B) = 1$), alors la confiance en A ($g(A)$), n'est pas égale à un, car cela dépend aussi de la confiance dans l'inférence « A est soutenu par B » , que l'on caractérise par une valeur p dans l'intervalle $[0, 1]$. L'estimation des confiances en B, $g(B)$, et en l'inférence, p, est un sujet important et complexe, qui fait appel à de nombreux domaines (statistiques, facteurs humain, etc.), et qui ne sera pas abordé dans ce manuscrit. Ce que nous proposons ici est un outil mathématique permettant de propager et de combiner ces confiances.

Si on calcule la confiance totale en utilisant les mêmes règles de calcul que pour un réseau Bayésien comme présenté Section1.4, on obtient : $g(A) = p * g(B)$. Dans ce cas, on observe une croissance linéaire de la confiance en A par rapport à la confiance en B, avec une pente spécifiée par p. Les confiances $g(A)$ et $g(B)$ étant définies sur $[0, 1]$, la valeur maximum de la confiance en A sera bornée par p. Cette évolution est illustrée sur la Figure 3.18.

Cette approche que nous proposons est équivalente à celle présentée par Simon et al. (2008), où des réseaux de fonctions de croyances correspondent à une transposition des réseaux Bayésiens à la théorie de l'évidence. Ils sont définis par des graphes orientés sans circuit permettant de modéliser à la fois de l'incertain aléatoire et de l'incertain épistémique par l'utilisation de Tables de Masses Conditionnelles (TMC) en lieu et place des Tables de Probabilités Conditionnelles (TPC). Cependant, nous nous plaçons dans le contexte particulier de la confiance avec des fonctions simples, les contraintes que nous proposons

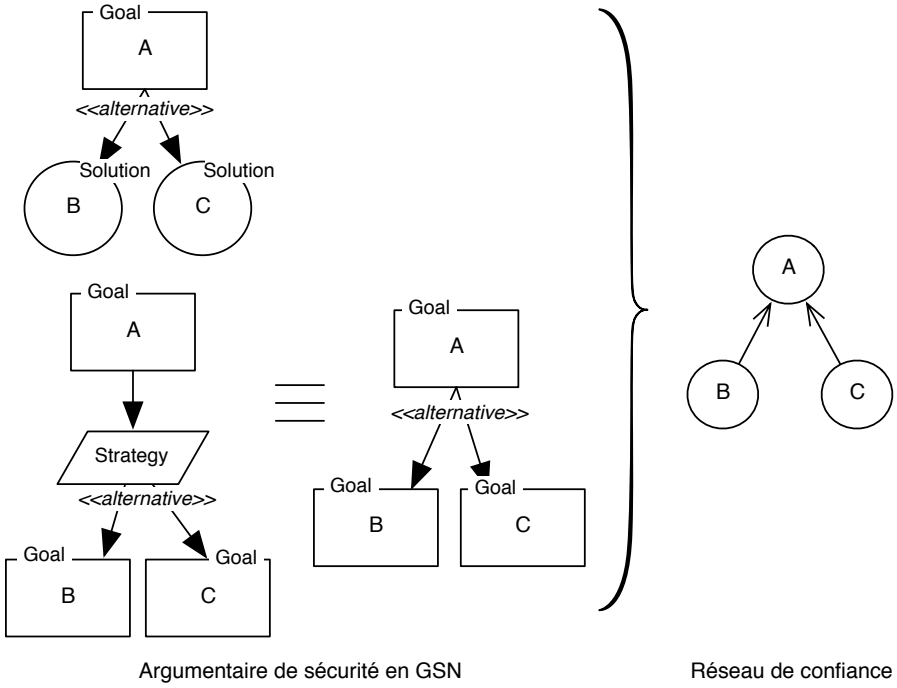


FIGURE 3.19 – Inférence de type argumentation alternative

par la suite pour la propagation de la confiance sont spécifiques au cas particulier des modèles GSN.

3.3.5 Argumentation alternative

L'argumentation alternative reflète les cas où l'on ajoute des arguments indépendants pour augmenter le niveau de confiance dans l'objectif principal. À titre d'exemple, prenons le cas où pour argumenter qu'un système est sûr on utilise des techniques de test, couplées avec des techniques de vérification formelle comme dans la Figure 3.15. Dans le cas où celui qui évalue un argumentaire n'exige pas que les 2 techniques soient réalisées, alors on est en présence d'une argumentation alternative. Il est possible de retrouver ce type d'argumentation au niveau des *Solutions*, mais également au niveau des *Goal*, comme cela est présenté Figure 3.19. Notons que la notion de stratégie en GSN n'est là que pour expliciter les choix d'argumentation qui ont été choisis. Pour cela, nous considérons que le réseau de confiance équivalent est simplement composé des nœuds liés aux *Goal* ou aux *Solution*. De la même manière que pour l'inférence simple, il est possible de compléter une matrice de confiance, en considérant les cas extrêmes de la confiance. Afin de proposer

une solution générique avec le moins de valeurs à déterminer pour ce genre de table nous avons opté pour l'utilisation du *Noisy-Or*, tel qu'il est présenté Section 1.4 : il s'agit d'un OU mais avec une probabilité p (au lieu de 1) lorsque $g(B) = 0$. On considère donc que lorsque une des deux prémisses est seule digne de confiance, alors on n'obtient pas une confiance maximum (c.à.d, égale à un), mais une confiance p. Cela est interprété de la manière suivante : p reflète la confiance dans A lorsque l'on n'a aucune confiance dans C et pleinement confiance dans B. Le raisonnement est le même pour q. La matrice de confiance est alors :

g(B)	0		1	
g(C)	0	1	0	1
g(A)	0	q	p	1-(1-p)(1-q)

Par rapport au *noisy-or* avec *Leak* présenté Section 1.4, nous n'utilisons pas de *leak* pour le cas où $g(B) = g(C) = 0$. En effet, dans le cas où les deux confiances sont nulles, il n'y a pas de justification à ce qu'il existe un certain degré de confiance en A, même si on a pleinement confiance dans les inférences, c.à.d, avec $p = q = 1$. Le calcul donne pour la confiance totale :

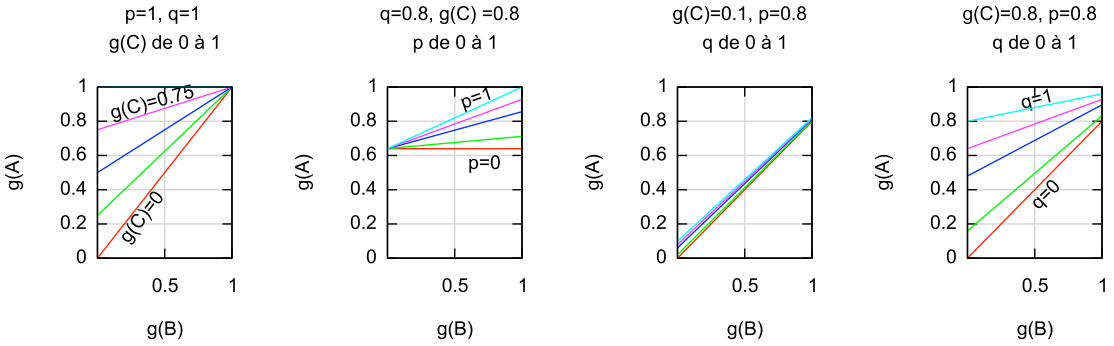
$$g(A) = p * g(B) + q * g(C) - g(B) * g(C) * p * q \tag{3.4}$$

La Figure 3.20 présente l'évolution de la confiance dans A en fonction de p et $g(B)$. La validation que nous proposons se base sur l'étude qualitative des variations de $g(A)$ en fonction de $g(B)$, $g(C)$, p et q. Pour cette figure et les suivantes, les courbes correspondent à 5 valeurs, $\{0; 0,25; 0,5; 0,75; 1\}$, de la variable étudiée. Par soucis de clarté, seules les valeurs limites seront annotées sur les diagrammes. Nous détaillons quelques observations, montrant que le modèle choisi est cohérent avec la notion de confiance dans le cadre d'une argumentation alternative :

— La figure (a) illustre l'indépendance entre $g(B)$ et $g(C)$. En effet, la confiance $g(A)$ tend vers 1, dès que $g(B)$ tend vers 1, quelle soit la valeur de $g(C)$.

— La Figure (b) montre l'influence de p sur $g(A)$. Pour $p = 0$, cela signifie que la confiance dans l'inférence «A est soutenu pas B» est nulle, et que donc la confiance ne dépend pas de la confiance dans B, ce qui est confirmé par le fait que $g(A)$ est constant pour $p = 0$.

— La Figure (c) montre que pour une valeur très faible de $g(C)$, le fait de faire varier q (la confiance de l'inférence A est soutenue par C) ne change pratiquement pas le résultat sur $g(A)$.



(a) (b) (c) (d)

FIGURE 3.20 – Graphique pour l'inférence de type argumentation alternative

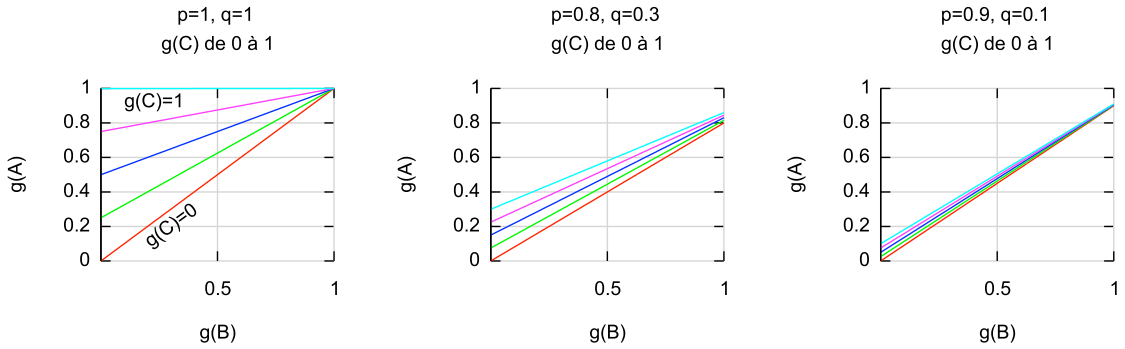
— La Figure (d) est le pendant de la (c) puisque au lieu d'avoir $g(C)$ faible on a $g(C) = 0,8$. On note dans ce cas que la variation de q a un fort impact sur l'évolution de $g(A)$. Ce graphe montre également que dès que $g(B)$ croît, alors pour $q > 0$, cela augmente la confiance dans A (et permet d'atteindre une confiance de plus 0,8).

Une autre justification consiste à relever les variations de g(A) aux limites (on note $g(A) \to 1$ pour «$g(A)$ tend vers 1» :

— Si $p = q = 1$ alors
  — si $g(B) \to 0$ alors $g(A) \to g(C)$
  — si $g(B) \to 1$ alors $g(A) \to 1$
  — si $g(C) \to 0$ alors $g(A) \to g(B)$
  — si $g(C) \to 1$ alors $g(A) \to 1$
— Si $q << 1$ alors
  — si $g(B) \to 0$ alors $g(A) \to q.g(C)$
  — si $g(B) \to 1$ alors $g(A) \to p + q.g(C)$
  — si $g(C) \to 0$ alors $g(A) \to p.g(B)$
  — si $g(C) \to 1$ alors $g(A) \to p.g(B) + q$

Pour $p << 1$ on obtient les tendances symétriques. Ces tendances sont observées sur la Figure 3.21 (a) et (b). Le graphe (c) illustre le fait que lorsque q tend vers 0 alors la confiance ne dépend plus que de $p.g(B)$. Au regard de ces observations, nous avons adopté ce choix de formule pour l'argumentation alternative. Notons qu'il est possible de généraliser la formule à n nœuds :

$$g = \sum_i g_i * p_i - \prod_i g_i * p_i \tag{3.5}$$



(a) (b) (c)

FIGURE 3.21 – Graphiques pour l'inférence de type argumentation alternative

Cependant, comme nous le verrons ultérieurement, seule la matrice de confiance suffira à renseigner le réseau de confiance.

3.3.6 Argumentation complémentaire

L'argumentation complémentaire intervient quand un ensemble de *Solution* ou de *Goal* sont requis simultanément. L'affaiblissement dans la confiance d'un des éléments doit faire chuter la confiance globale. Cependant, comme pour l'argumentation alternative, on peut associer des poids à chaque branche de l'argumentaire qui reflète son importance relative dans l'argumentaire. Par exemple, si l'objectif de haut niveau d'un argumentaire est «Tous les risques ont été traités», alors on portera une attention plus forte sur les sous-objectifs dont les risques sont les plus critiques. Cette attention devrait se traduire par une exigence plus forte vis-à-vis de la confiance dans le traitement de ces risques.

Comme pour l'argumentation alternative, de nombreux modèles de calcul existent, mais aucun de ceux disponibles avec les réseaux Bayésiens ne nous a fourni de solution acceptable permettant de décrire les tendances souhaitées. Nous avons alors adopté un raisonnement par l'incertitude dans le cadre de la théorie de croyance. On considère les propriétés de la croyance définies par l'équation 3.3 :

$$\begin{cases} m(A) = g(A) \ : \ confiance \\ m(A, \overline{A}) = 1 - g(A) \ : \ incertitude \\ m(\overline{A}) = 0 \end{cases}$$

En raisonnant par les incertitudes on obtient alors le tableau suivant en utilisant un *Leaky Noisy-OR* tel que présenté dans la section 1.4 en notant l'incertitude résiduelle $v = 1 - l$, et avec la condition supplémentaire $m(A, \overline{A}) = 1$ quand $m(B, \overline{B}) = m(C, \overline{C}) = 1$ (pour une incertitude maximum dans B et C, on fixe une incertitude maximum pour A) :

$m(B, \overline{B})$	0		1	
$m(C, \overline{C})$	0	1	0	1
$m(A, \overline{A})$	$1 - v$	$1 - v.(1 - q)$	$1 - v.(1 - p)$	1

En passant par la relation $g(X) = 1 - m(X, \overline{X})$, le tableau précédent devient :

$g(B)$	1		0	
$g(C)$	1	0	1	0
$g(A)$	v	$v.(1 - q)$	$v.(1 - p)$	0

L'expression de g(A) devient alors (non simplifiée) :

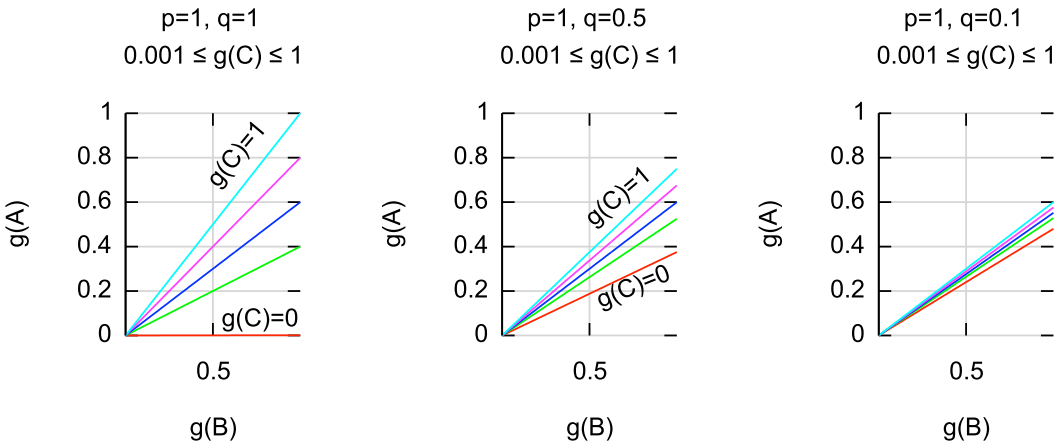
$$g(A) = g(B) * g(C) * v + g(B) * (1 - g(C)) * v * (1 - q) + (1 - g(B)) * g(C) * v * (1 - p) \quad (3.6)$$

La principale différence avec les autres travaux de recherche utilisant des *Noisy-AND* se trouve dans l'interprétation des paramètres de ces tableaux. Dans notre cas, p et q vont représenter l'importance (le poids) de B et C dans leur contribution à la confiance dans A. La somme de ces paramètres va également représenter la confiance globale dans l'inférence liée à l'argument complémentaire. Nous proposons d'utiliser la valeur de leak qui correspond au cas où $g(B) = g(C) = 1$ pour représenter cette confiance. Afin de normaliser, nous proposons donc dans le cas de 2 nœuds parents d'utiliser la formule : $v = (p + q)/2$. Dans le cas où p et q sont inférieurs à 1, la confiance dans les inférences impliquant B et C n'est pas maximale, alors v est inférieur à 1. La généralisation de cette contrainte dans le cas de l'argumentation complémentaire avec n parents est :

$$v = \frac{1}{n} \sum_{i=1}^{n} p_i \quad (3.7)$$

Les valeurs dans la table de confiance pour un nœud X ayant n parents sont :

$$\begin{cases} g(X|\overline{Y_1}, ..., \overline{Y_k}) = v. \prod_{i=1}^{k} (1 - p_i) \quad avec \quad k < n \\ g(X|\overline{Y_1}, ..., \overline{Y_n}) = 0 \end{cases} \quad (3.8)$$



(a) (b) (c)

FIGURE 3.22 – Argumentation complémentaire, l'influence de g(C)

où p_i représente le poids de Y_i dans l'argumentation.

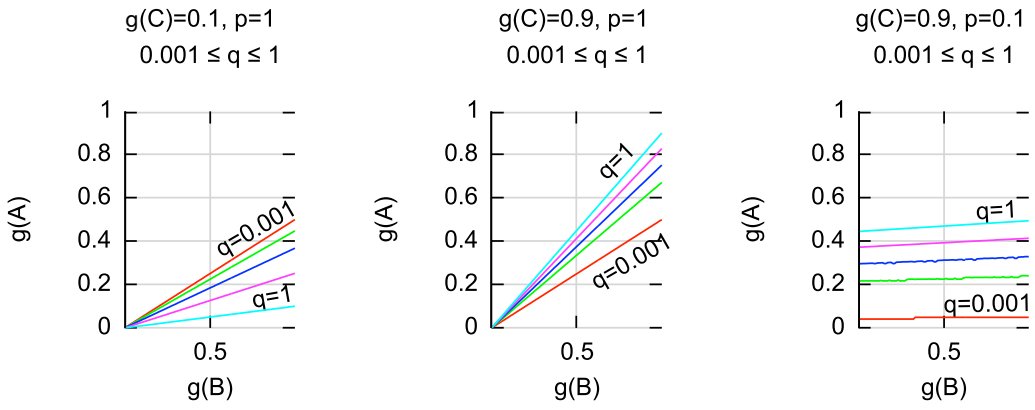
Nous considérons par la suite que tout élément ayant une confiance nulle n'est pas pris en compte dans l'argumentation car un tel élément sera enlevé de l'argumentaire de sécurité. Les Figures 3.22 et 3.23 illustrent le résultat dans le cas de deux parents B et C.

Dans la Figure 3.22, q prend respectivement les valeurs $q = 1$ (a), $q = 0,5$ (b) et $q = 0,1$ (c) pour $p = 1$. La figure (a) illustre en premier lieu le fait qu'une confiance nulle dans $g(B)$ ou $g(C)$ à poids équivalent, induit une confiance nulle dans A. Dans les cas (b) et (c), où le poids de $g(C)$ est plus faible, on obtient une confiance non nulle (même pour $g(C) = 0$). On observe également sur les 3 diagrammes, que les droites tendent à se confondre en une seule droite limite, $g(A) = 0.5 * g(B)$, ce qui illustre bien que lorsque q diminue, l'influence de $g(C)$ diminue.

Les Figures 3.23 (a) et (b) illustrent également cette tendance dans le cas où $g(C)$ est faible (0,1) puis élevée (0,9). Dans (c), l'influence de $g(B)$, notée p, est faible. La confiance $g(A)$ reste quasiment insensible à l'évolution de $g(B)$, les lignes sont presque horizontales, ce qui confirme également la tendance que nous souhaitons observer.

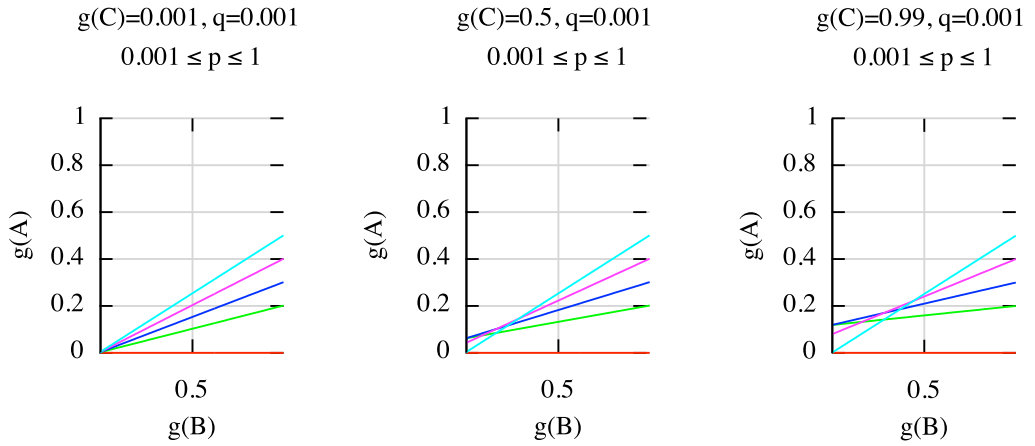
Comme pour l'argumentation alternative, une autre justification consiste à relever les limites attendues de $g(A)$:

- Si $p = q = 1$ alors
  - si $g(B) \to 0$ alors $g(A) \to 0$
  - si $g(B) \to 1$ alors $g(A) \to g(C)$
  - si $g(C) \to 0$ alors $g(A) \to 0$



(a) (b) (c)

FIGURE 3.23 – Argumentation complémentaire, l'influence de q



(a) (b) (c)

FIGURE 3.24 – Argumentation complémentaire, le cas pour les confiances basses

- si $g(C) \to 1$ alors $g(A) \to g(B)$

— Si $p >> q$ (C a un poids faible donc g(C) a peu d'influence) alors
- si $g(B) \to 0$ alors $g(A) \to 0$
- si $g(B) \to 1$ alors $g(A) \to p.g(B)/2$
- si $g(C) \to 0$ alors $g(A) \to p.g(B)/2$
- si $g(C) \to 1$ alors $g(A) \to p.g(B)/2$

Les tendances observées en utilisant notre *Noisy-AND* modifié, correspondent aux valeurs présentées ci-dessus. Seul le cas où les confiances sont faibles ne correspond pas au comportement attendu : lorsque q est faible et $g(B)$ tend vers 0, on devrait obtenir une limite de 0, or comme le montre les Figures 3.24 (b) et (c), $g(A)$ n'est pas nul pour des valeurs supérieures à 0 de p (la Figure (a) permet de vérifier que lorsque $g(C) = 0$ la propriété est tout de même vérifiée). On peut également observer ces limites en utilisant la formule 3.6. Cependant, les Figures (b) et (c) présentées ici, illustrent bien le fait que l'on obtient cette déviation de comportement uniquement pour des confiances basses. Bien que ce modèle ne soit pas parfait, il fournit des résultats satisfaisant pour la plupart des situations qui sont pertinentes en pratique. La recherche d'un modèle plus optimal fera partie de notre prospective.

En utilisant l'équation 3.8 pour plusieurs parents, on donne ci dessous l'exemple du cas à 3 parents (B,C,D) :

$g(B)$	1				0			
$g(C)$	1		0		1		0	
$g(D)$	1	0	1	0	1	0	1	0
$g(A)$	v	v.(1-r)	v.(1-q)	v.(1-q).(1-r)	v.(1-p)	v.(1-p).(1-r)	v.(1-p).(1-q)	0

Les poids de B, C, et D étant respectivement p, q, r, avec $(p+q+r)/3 = v$. Au regard de la complexité de l'expression de g(A), un outil permettant de ne saisir que les tables est requis pour effectuer les calculs (comme nous le verrons nous utilisons des réseaux avec plus d'une dizaine de nœuds parents pour un nœud fils). Ce point sera abordé Section 4.6.2.

3.3.7 Argumentation mixte

Les deux cas simples présentés dans les sections précédentes permettent de couvrir la notion de *Context* présente dans les GSN. En effet, la notion de contexte telle que présentée Section 1.6, est en réalité un élément obligatoire de l'argumentaire, c'est donc une argumentation de type complémentaire que l'on obtient entre le contexte et les autres éléments. Par exemple, comme dans la Figure 3.25, si un élément de type *Context* est

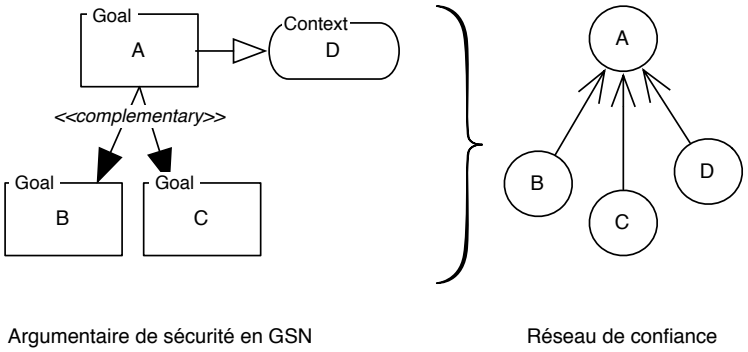


FIGURE 3.25 – Inférence mixte1

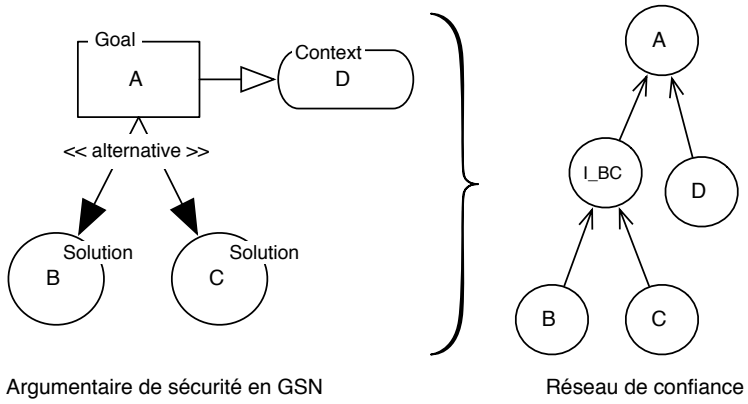


FIGURE 3.26 – Inférence mixte2

ajouté à un *Goal*, qui est soutenu par deux sous *Goal*, via une argumentation de type complémentaire, alors cela reviendra à construire un réseau de confiance avec des tables de type argumentation complémentaire.

Dans le cas d'une argumentation alternative, comme dans la Figure 3.26, il faut introduire un nœud intermédiaire qui sera du type alternatif, puis considérer un argumentaire de type complémentaire entre ce nœud intermédiaire et le nœud D.

3.3.8 Étude de sensibilité

Une fois l'argumentaire de sécurité et le réseau de confiance réalisé, nous proposons de réaliser une étude de sensibilité. Ceci consiste à identifier quels éléments de l'argumentaire ont le plus d'influence (négative ou positive) sur la confiance globale. Comme présenté précédemment Section 3.3.1, l'objectif est de pouvoir consolider un argumentaire.

Nous proposons d'utiliser le graphe de tornade (*tornado graph*). Il s'agit d'un outil statistique simple qui permet de satisfaire cet objectif en montrant les variables dont l'incertitude a le plus d'impact sur le résultat souhaité. Dans notre cas le résultat souhaité est la confiance dans l'argumentaire mais cette technique est utilisée dans de nombreux autres domaines. L'interprétation générale d'un tel graphe est simple dans le contexte de la confiance, car il présente visuellement l'impact potentiel de la variation d'un paramètre sur la confiance globale.

La technique consiste à partir d'une fonction $f(x_1, ...x_n)$, dont on a estimé les valeurs $X_1, ..., X_n$ de référence des variables x_i, à calculer pour chaque $x_i \in [X_{min}, X_{max}]$, les valeurs $f(X_1, ..., X_{i-1}, X_{min}, X_{i+1}, ...X_n)$ et $f(X_1, ..., X_{i-1}, X_{max}, X_{i+1}, ...X_n)$. Les valeurs X_{min} et X_{max} étant les valeurs minimum et maximum possibles de la variable x_i. Ainsi pour chaque x_i on obtient un intervalle de la fonction f. Le graphe de tornade permet de visualiser, de façon triée, l'ensemble des ces intervalles.

Nous prendrons l'exemple de l'argumentaire alternatif à 2 nœuds parents et un nœud enfant, présenté dans la Section 3.3.5. Si l'on prend des valeurs arbitraires q=0,7 et p=0,9 on obtient la table suivante :

g(B)	0		1	
g(C)	0	1	0	1
g(A)	0	0,7	0,9	0,97

Plaçons nous dans le cas où une première estimation de la confiance dans les nœuds parents B et C donne $g(B) = 0,8$ et $g(C) = 0,7$. En utilisant la table pour calculer la confiance totale on obtient $g(A) = 0,8572$. À partir de cette valeur, on peut calculer les variations de $g(A)$ en ne faisant varier que p entre 0 et 1 (son domaine de variation possible), puis faire de même pour les variables q, $g(B)$ et $g(C)$.

Par exemple, en gardant toutes les valeurs précédentes de p, q et $g(C)$, et en ne faisant varier que $g(B)$:

— si g(B)=0 alors g(A) = 0,49

— si g(B)=1 alors g(A) = 0,949

Une fois ces valeurs calculées, on établit le graphe de la Figure 3.27. En abscisse est représentée la valeur de $g(A)$. Pour chaque variable, $g(B)$, p, $g(C)$ et q, on trouve une représentation de la valeur minimum et maximum de g(A), avec une partie qui part de la valeur de référence $g(A) = 0,8572$ vers une confiance plus faible, et une autre vers une confiance plus importante. Sur ce graphe, les histogrammes ont été classés pour faire apparaître les paramètres du plus influant au moins influant. Ainsi, on peut noter que c'est g(B) qui aura l'influence la plus importante à la fois dans la baisse de la confiance, mais

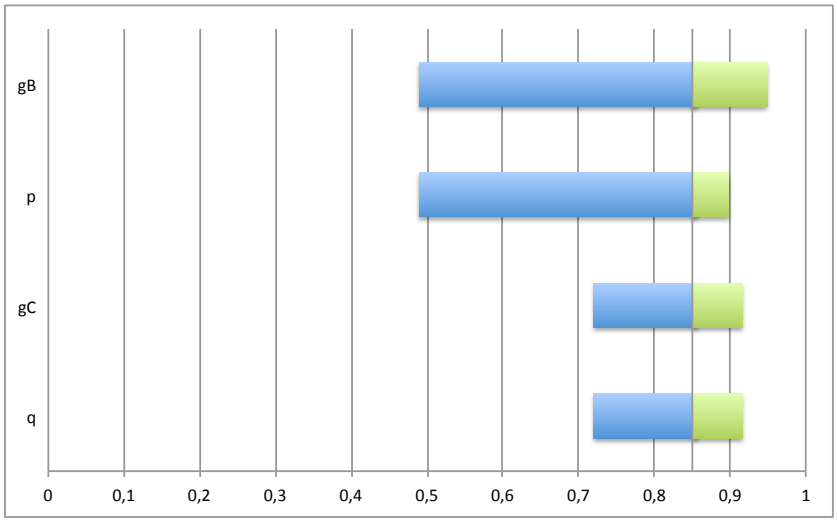


FIGURE 3.27 – Graphe de Tornade avec g(B), g(C) , p et q pour une argumentation alternative

aussi pour augmenter la confiance. C'est donc sur cet élément que devront de porter les efforts si l'on souhaite améliorer la confiance globale. Cependant il est important de noter les limites d'une étude de sensibilité en utilisant le graphe de Tornade. Par exemple il est impossible d'identifier l'influence d'une variation simultanée de deux ou plus variables. Il est également difficile sur cet exemple de savoir s'il sera plus facile d'augmenter la confiance grâce à $g(B)$ ou de p. Ce travail doit être effectué par les experts. Le modèle que l'on propose est donc un indicateur qui permet de guider les analystes mais ne supplante pas leur travail d'argumentation.

On retrouve la possibilité d'effectuer ces calculs en utilisant l'outil Agenarisk, à la fois pour modéliser le réseau comme sur la Figure 3.28 auquel on associe la table présentée Figure 3.29, ainsi que pour générer le graphe de Tornado, présenté Figure 3.30 (AgenaRisk ne permet pas de représenter la sensibilité des poids p et q). On peut noter qu'un des intérêts de notre approche est que les règles de calcul que l'on utilise sont les mêmes que pour les réseaux bayésiens, et que donc il est possible d'utiliser directement des outils comme Agenarisk.

3.4 Conclusion

Nous avons présenté dans cette section une contribution à l'évaluation quantitative de la confiance dans un argumentaire. Ce thème étant encore récent dans le domaine de

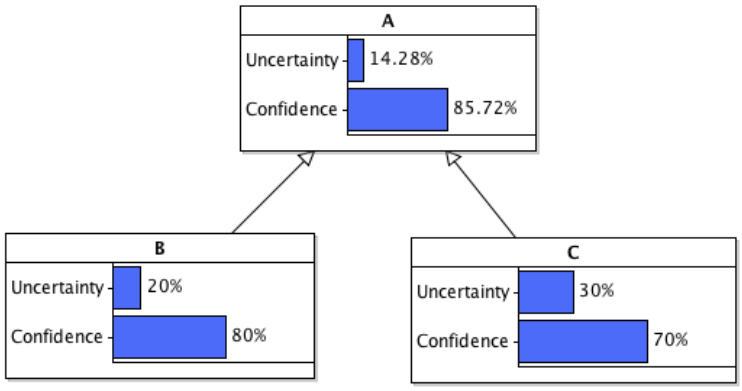


FIGURE 3.28 – Réseau pour un argumentaire alternatif avec l'outil Agenarisk

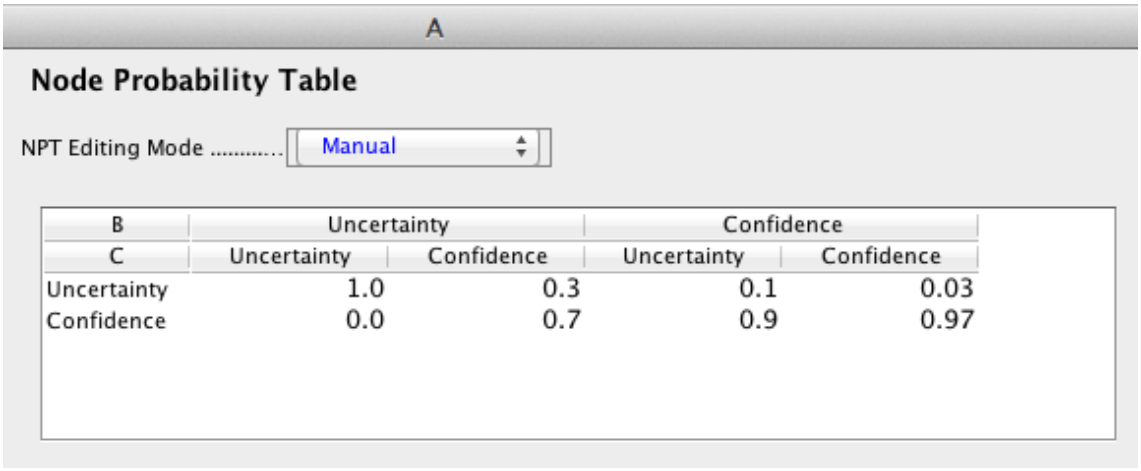


FIGURE 3.29 – Table de confiance avec g(B) et g(C) pour argumentation alternative avec l'outil Agenarisk

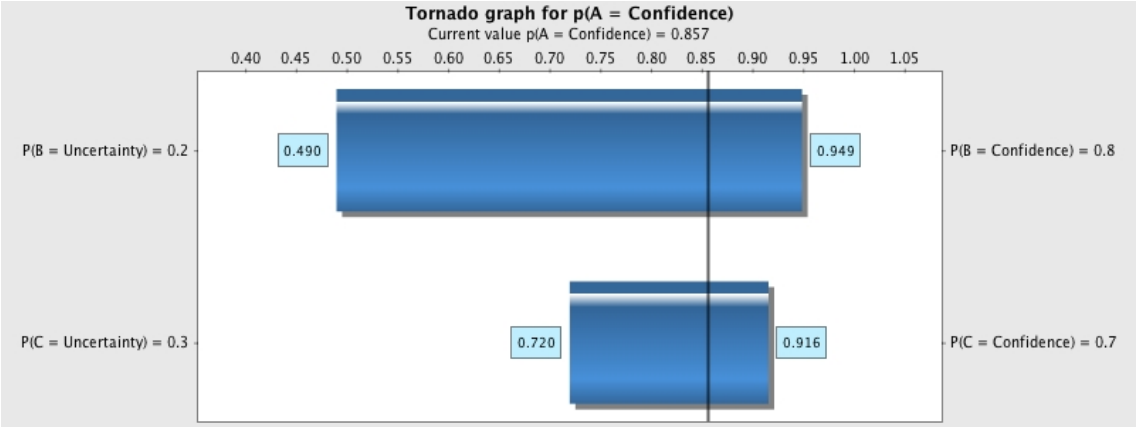


FIGURE 3.30 – Graphe de tornade avec g(B) et g(C) pour argumentation alternative

l'argumentation de sécurité, nous avons identifié deux grandes tendances : les méthodes basées sur les réseaux Bayésiens, et celles basées sur la théorie de Dempster-Shafer. Les premières ne faisant pas apparaître explicitement la notion de confiance et d'incertitude, et les secondes ne correspondent pas à nos besoins sur les règles de propagation, nous avons proposé une méthode partant de la définition de la croyance, mais pouvant s'appuyer sur des calculs similaires aux réseaux Bayésiens.

Le fait de se baser sur la théorie de Dempster-Shafer mais dans le cadre de la fonction à support simple (c.à.d, avec $m(A, \overline{A}) = 0$) a deux avantages :

- Le cadre conceptuel permet dès le départ de prendre en considération les incertitudes, et de les quantifier.
- On reste dans un cadre mathématique qui permet d'utiliser les réseaux Bayésiens ainsi que tous les outils qui les accompagnent

Nous avons proposé d'utiliser deux types d'argumentaire : l'alternative, reposant sur une propagation de type *noisy-or*, et le complémentaire reposant sur une version adaptée de *leaky noisy-and*.

Le modèle de propagation de la confiance que nous avons choisi est une première proposition qui doit encore être améliorée notamment pour l'argumentation complémentaire. Cependant le cadre conceptuel devrait permettre d'explorer d'autres expressions mathématiques.

L'objectif étant l'étude de sensibilité, il est important de noter que ce sont les variations relatives de la confiance qui nous intéressent et pas une valeur absolue de la confiance, qui serait en réalité très difficile à justifier.

Un point important de cette étude et qui n'est pas abordé dans ce manuscrit est la détermination des valeurs de confiance des nœuds et des inférences. Ce travail complexe repose sur de nombreux domaines, dont les sciences sociales et les statistiques pour l'aspect quantification d'avis d'experts, et sort du cadre de cette thèse.

Chapitre 4

Application à un robot d'aide à la déambulation

La méthode présentée dans ce manuscrit a été utilisée dans le cadre du développement d'un robot déambulateur au sein du projet ANR-MIRAS de l'appel TECSAN2009. Dans ce chapitre, nous présentons d'abord le contexte de ce cas d'étude et les principales fonctionnalités du robot d'aide à la déambulation. Nous résumons ensuite les résultats de l'application de la méthode HAZOP-UML qui ont conduit à établir la liste des dangers à prendre en compte pour l'analyse de sécurité du robot. Cette analyse inclut une étude comparative de l'application des mots guides aux différents types de représentation considérés dans le cadre de la méthode HAZOP-UML (cas d'utilisation, diagrammes de séquence, diagrammes d'états-transitions) ainsi qu'avec la méthode classique d'analyse préliminaire des risques (APR). La dernière partie de ce chapitre traite de la construction d'argumentaire de sécurité basé sur GSN et l'évaluation de la confiance dans cette argumentaire basée sur la méthode présentée dans le chapitre 3.

4.1 Présentation générale du projet MIRAS

Parmi les systèmes de la robotique de service, la robotique de rééducation (*rehabilitation robotics*) a pour vocation l'aide à la manipulation (pour remplacer un mouvement impossible par exemple), à la mobilité (fauteuils roulants dits intelligents, exosquelettes), et à la thérapie (réalisation d'exercices de mobilité assisté par un robot).

Le projet ANR MIRAS *(Multimodal Interactive Robot for Assistance in Strolling)* (2009 à 2012) présenté dans (Pasqui et al., 2012), s'inscrit dans l'aide à la mobilité. L'objectif de ce projet est la conception et la mise au point d'un robot pour l'aide à la déambulation et

la surveillance de l'état physiologique de personnes âgées atteintes de troubles de la marche et d'orientation. Les partenaires de ce projet étaient les concepteurs de robots de service ROBOSOFT [1], les laboratoire ISIR [2] et LAAS-CNRS, les hôpitaux du réseau Assistance Publique-Hôpitaux de Paris (Henri-Mondor [3] et Charles-Foix [4]) et le Centre Hospitalier Universitaire de Toulouse [5]. Les travaux de recherche et de développement du robot et de ses fonctions multimodales sont réalisés dans les laboratoires de recherche et les évaluations cliniques sont effectuées au sein des hôpitaux. Les bénéfices attendus d'un tel système sont :

— rendre l'autonomie de la marche à des patients souffrant de troubles de l'équilibre et d'orientation ;
— adapter et contrôler l'activité de marche pour l'entrainement et l'amélioration de la condition physique des personnes âgées ;
— libérer les personnels hospitaliers pour des actes plus techniques ;
— sécuriser le patient.

Un exemple typique dans le contexte hospitalier et qui illustre les objectifs scientifiques et techniques est l'accompagnement aux toilettes. La personne ne pouvant ni se lever seule de son fauteuil ni marcher seule quelques mètres, appelle un soignant qui ne fait que la soutenir. Le robot remplace ici avantageusement le soignant. Pour cela, il doit exécuter un certain nombre d'actions qui dépendent de l'utilisateur (il ne « sait » pas a priori que la personne veut aller aux toilettes).

4.2 Vue générale du travail réalisé

La Figure 4.1 présente les activités réalisées lors du projet MIRAS dans le cadre de cette thèse. Une première analyse HAZOP-UML sur un sous-ensemble des fonctionnalités et limitée aux cas d'utilisation et diagrammes de séquence a été réalisée en amont de la thèse. À la suite des recommandations préliminaires de sécurité issues de cette étude, ainsi que de la redéfinition de l'architecture mécanique pour des exigences ergonomiques, une deuxième version a été spécifiée. Celle-ci a été analysée lors d'une deuxième étude HAZOP-UML (présentée section 4.4, avec une extension des mots-guides HAZOP, et une application aux diagrammes d'états-transitions. Une analyse préliminaire des risques (APR) a été également réalisée pour établir un point de comparaison avec HAZOP-UML, ce qui a contribué à l'analyse de la validité de la méthode HAZOP-UML (présentée section 4.4.2). Puis une détermination des critères de risques et leur utilisation a permis d'établir une

1. http://www.robosoft.com
2. http://www.isir.upmc.fr
3. http://chu-mondor.aphp.fr
4. http://www.aphp.fr/hopital/charles-foix/
5. http://www.chu-toulouse.fr

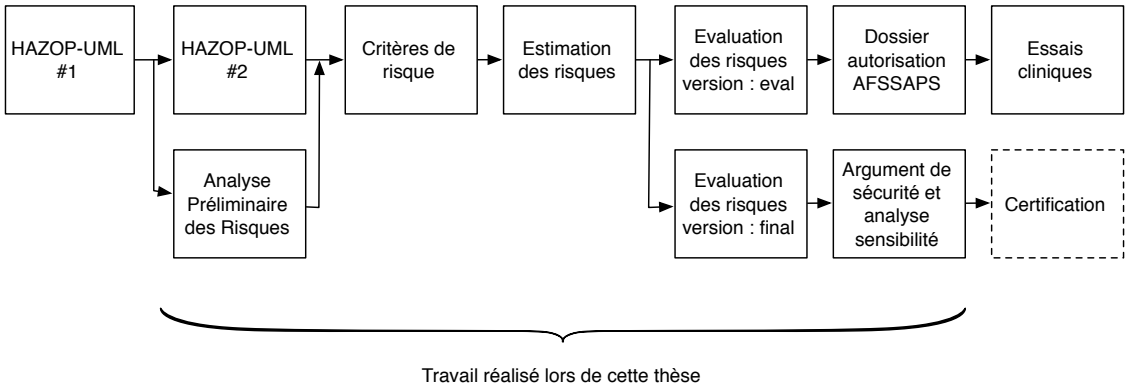


FIGURE 4.1 – Activités de gestion du risque dans le cadre du projet MIRAS

estimation des risques (Section 4.4.3). Cette estimation a permis de réaliser une évaluation des risques pour une version du robot destinée aux évaluations cliniques (section 4.4.2), ce qui a notamment servi de base au dépôt et à l'acceptation d'une requête auprès de l'AFSSAPS [6], une agence d'évaluation, qui délivre les autorisations pour les essais cliniques dans le domaine de la santé. Parallèlement, l'évaluation de la version finale du robot a été réalisée en construisant un argumentaire de sécurité (Section 4.5), ainsi que son modèle de confiance (Section 4.6).

L'ensemble de ce travail n'est pas présenté ici dans son intégralité, mais est disponible pour partie dans les livrables (Caquas et al., 2012; Do Hoang et Guiochet, 2012). Les évaluations cliniques ont été réalisées mais le robot considéré pour l'étude a finalement été amélioré et a donné naissance à une troisième version développée à l'ISIR hors du cadre de MIRAS.

4.3 Les fonctionnalités du robot d'aide à la déambulation

Les tâches de base que le robot doit effectuer sont d'aider le patient à se lever (d'une chaise ou d'un lit), de déambuler et de s'assoir. Il est destiné à une utilisation quotidienne par les personnes âgées atteintes de troubles de la marche et de l'équilibre. Le robot doit pouvoir distinguer un mouvement volontaire du patient pour l'assister davantage et pour faire face à un mouvement perturbateur (comme la perte d'équilibre) afin de corriger la posture du patient. En plus de ces fonctions, plusieurs services ont été ajoutés au robot comme la détection des problèmes physiologiques (fatigue, augmentation du rythme cardiaque),

6. Agence française de sécurité sanitaire du médicament et des produits de santé (Afssaps) devenue Agence nationale de sécurité du médicament `http://ansm.sante.fr/`

mauvaise posture et chute du patient. Une décomposition en cas d'utilisation UML, a été réalisée et donnée ci-dessous ainsi que sur la Figure 4.2 :

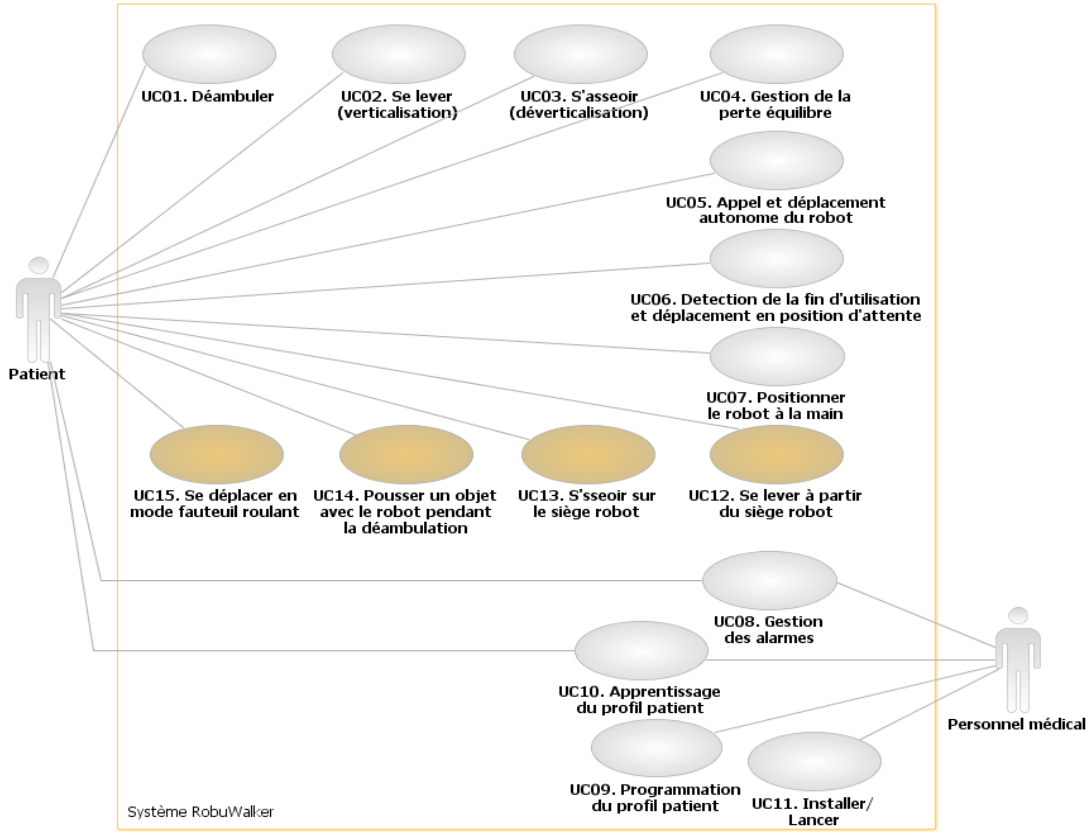


FIGURE 4.2 – Le diagramme de cas d'utilisation du déambulateur intelligent (les cas en couleur correspondent à ceux ajoutés lors de la 2ème version du robot)

`UC01` **Déambuler** Le robot assiste la déambulation du patient.

`UC02` **Se lever (verticalisation)** Le robot assiste le patient quand il se lève par un mouvement de la base et des bras du robot.

`UC03` **S'asseoir (déverticalisation)** Le robot assiste le patient quand il s'assied par un mouvement de la base et des bras du robot.

`UC04` **Gestion de la perte d'équilibre** Le robot détecte les pertes d'équilibre du patient et aide ce dernier à retrouver son équilibre.

`UC05` **Appel et déplacement autonome du robot** Le robot réagit à l'appel du patient, se déplace de manière autonome vers ce dernier, et se positionne de manière à ce que le patient puisse saisir les poignées.

`UC06` **Détection de la fin d'utilisation et déplacement en position d'attente** Le robot retourne à sa position d'attente lorsqu'il n'est plus utilisé.

`UC07` **Positionner le robot à la main** L'utilisateur déplace le robot selon ses besoins, notamment avant la verticalisation pour un meilleur confort du patient

`UC08` **Monitoring du patient et gestion des alarmes** Le robot surveille en permanence l'état physiologique et la posture du patient et déclenche l'alarme pour avertir le personnel médical si un problème est détecté (fatigue du patient, mauvaise posture, ou chute).

`UC09` **Programmation du profil patient** Le personnel médical choisit le profil adéquat du patient afin que le robot puisse l'assister.

`UC10` **Apprentissage du profil patient** Le patient fait quelques essais avec le robot qui mémorise ses aptitudes physiques.

`UC11` **Installer/Lancer** Le personnel technique installe et démarre le robot.

`UC12` **Se lever à partir du siège robot** Le patient se lève du siège du robot avec aide

`UC13` **S'asseoir sur siège robot** Le patient s'assied sur le siège du robot avec aide

`UC14` **Pousser un objet avec le robot pendant la déambulation** Le patient pousse un objet avec le robot pendant la déambulation.

`UC15` **Se déplacer en mode fauteuil roulant** Le patient s'assied sur le siège et le robot se déplace suivant la commande du patient.

En se basant sur les besoins fonctionnels UC01 à UC11, un premier prototype du robot a été réalisé (Figure 4.3). Cependant, à la suite des analyses de risques, il a été recommandé que la deuxième version inclue un siège pour que le patient fatigué puisse se reposer dessus, ou se reposer en cas de défaillance du robot. Pour des questions pratiques, ce siège pourrait être robotisé pour aider le patient à se lever et à s'asseoir. Cette nouvelle version prévoit également que le patient puisse pousser un objet comme une porte avec le robot (usage existant dans les déambulateurs classiques). La deuxième version inclut donc les cas d'utilisation UC12,UC13 et UC14. Le cas d'utilisation UC15 est mentionné ici mais a été considéré comme en dehors du champ d'étude du projet MIRAS, il correspond à une utilisation du déambulateur comme un fauteuil roulant lorsque le patient est assis sur le siège.

Au delà des exigences fonctionnelles, un important travail sur l'ergonomie du robot a été réalisé. En effet, les patients qui se déplacent à l'aide d'un déambulateur classique déambulent en s'appuyant sur les poignées. L'objectif du robot pendant la déambulation



FIGURE 4.3 – Un déambulateur classique (à gauche) et les versions du Robot MIRAS

est de rendre ces efforts répétitifs moins importants, et donc d'avoir un couple moteur qui réduit la force nécessaire pour pousser un robot de ce poids. Il faut également une bonne mobilité afin de pouvoir tourner dans les hôpitaux et d'éviter l'effet « caddy » (un mouvement latéral n'est pas possible). La figure 4.4 présente les deux prototypes.

4.4 Analyse et évaluation des risques

4.4.1 Application d'HAZOP-UML

La modélisation UML a été réalisée au LAAS puis validée par les partenaires du projet, notamment les chercheurs de l'ISIR en charge du développement des fonctions robotiques, et par les docteurs des hôpitaux. Chaque cas d'utilisation a été décrit au moyen d'une table (pré et post conditions, invariants) (voir dans la partie supérieure de la Figure 4.7) et d'un diagramme de séquence pour le scénario nominal, et d'un ou plusieurs diagrammes de séquence pour les scénarios alternatifs. La Figure 4.5 illustre par exemple le fait qu'un patient puisse lâcher les poignées pendant la verticalisation. Notons que ces diagrammes, ont servi de base pour discuter au sein du projet pour mieux définir les spécifications, et les diagrammes de séquence ont été un outil particulièrement puissant pour discuter avec des non-spécialistes de la conception système comme les docteurs par exemple. Un diagramme d'états-transitions a été réalisé, dont une version simplifiée est donnée Figure 4.6. L'ensemble des diagrammes UML est consultable en ligne [7].

L'identification a été réalisée en effectuant une APR qui a consisté en plusieurs réunions de *brainstorming* avec les membres de l'ISIR, puis avec une analyse HAZOP-UML. Un

7. http ://homepages.laas.fr/qdohoang/MIRAS/



FIGURE 4.4 – Le premier prototype (Robosoft) (à gauche) et le deuxième (ISIR)(à droite)

extrait d'une table HAZOP telle que nous l'avons déployée est donné Figure 4.7. Cette étude a donné lieu à une liste de dangers présentée Figure 4.9, que l'on a extrait des tables HAZOP. Pour chaque danger nous avons également utilisé des références vers les lignes des tables HAZOP induisant le danger. Ce travail permet ainsi d'avoir une traçabilité entre les causes et les conséquences, et peut s'avérer très utile si l'on souhaite utiliser d'autres techniques d'analyse du risque comme les arbres de fautes ou une AMDEC par exemple.

La liste est donnée ci-dessous :

`HN01` Posture incorrecte du patient pendant l'utilisation du robot (penché en avant ou en arrière)

`HN02` Chute du patient pendant l'utilisation du robot (comme pour un déambulateur classique), soit au sol, soit sur le robot lui-même

`HN03` Arrêt total du robot pendant l'utilisation (absence d'énergie), rendant impossible toute action du robot

`HN04` Chute du patient sans alarme ou avec alarme tardive

`HN05` Problème physiologique du patient sans alarme ou avec alarme tardive

`HN06` Chute du patient provoquée par le robot (mouvement non désiré du robot)

`HN07` Incident détecté mais défaut de passage en mode sûr ; le robot continue à se déplacer alors qu'il a un déséquilibre, une chute ou une fatigue du patient

`HN08` Le robot coince un membre du patient entre 2 parties du robot ou entre le robot et un objet fixe

`HN09` Collision entre le robot (ou partie du robot) et le patient

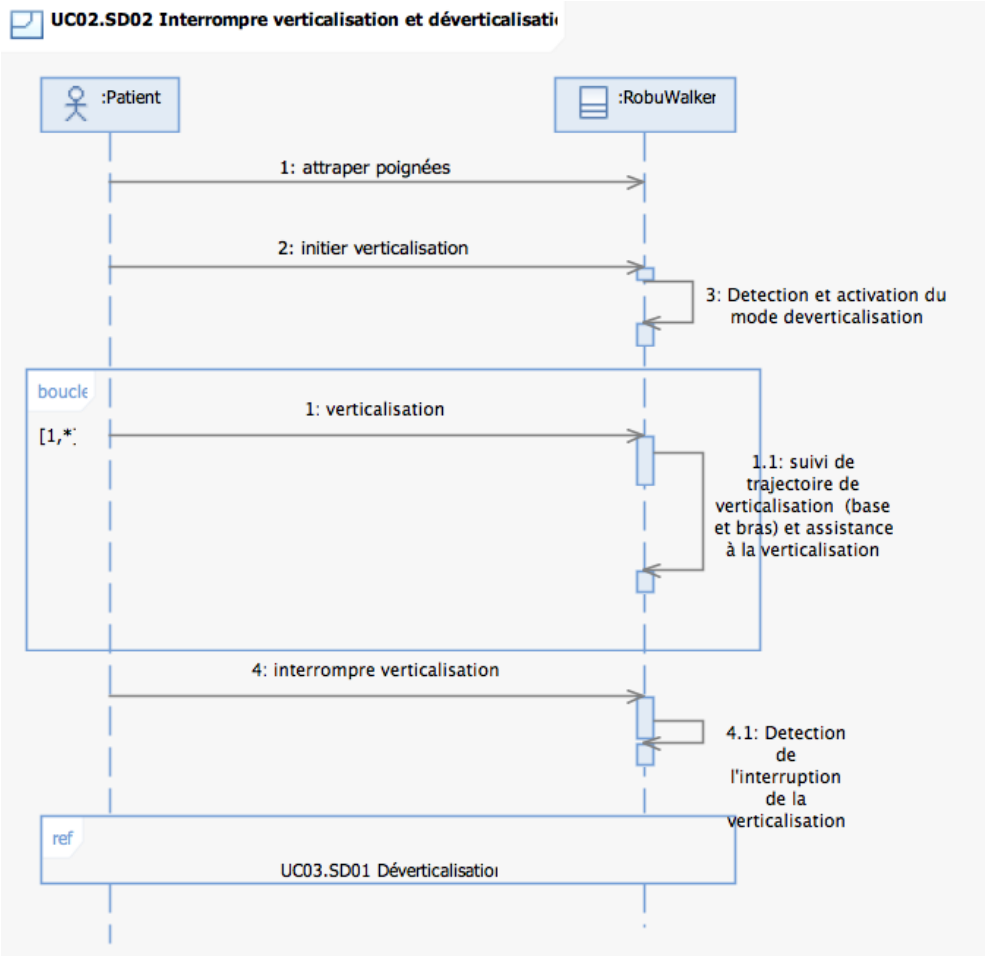


FIGURE 4.5 – Diagramme de séquence illustrant une interruption au milieu de la verticalisation

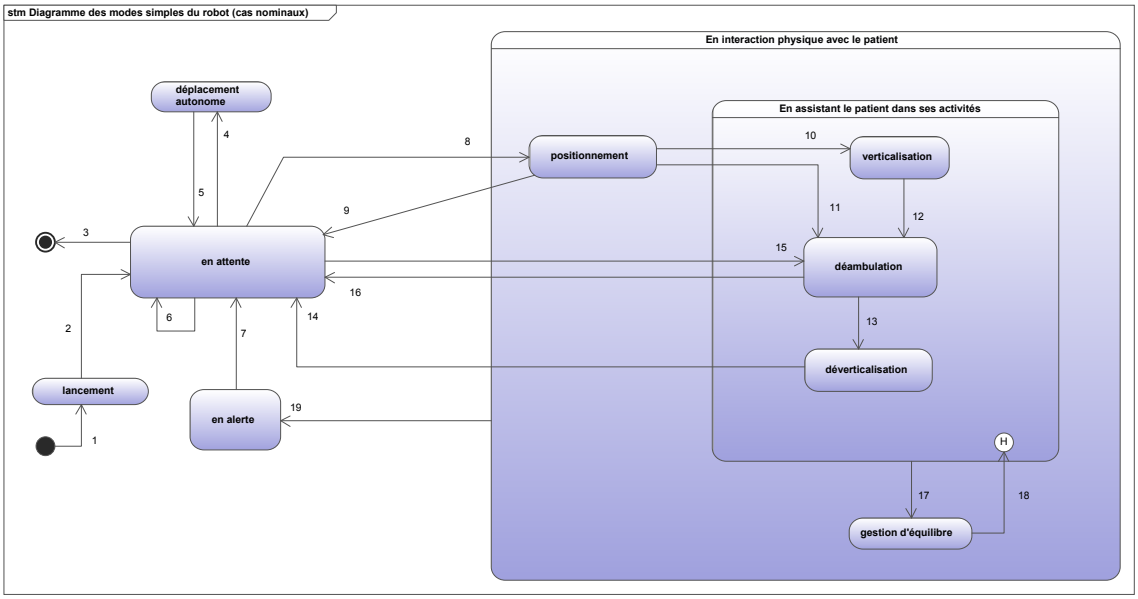


FIGURE 4.6 – Diagramme d'état transition du robot déambulateur

`HN10` Collision entre le robot et une personne autre que le patient

`HN11` Le robot gêne le personnel médical pendant une intervention.

`HN12` Déséquilibre du patient provoqué par le robot (sans chute)

`HN13` Fatigue du patient due à une mauvaise ergonomie ou commande du robot (sans chute)

`HN15` Chute du patient depuis le siège

`HN16` Alarme trop fréquente (Faux positif)

Nous avons choisi de différencier HN02, HN04 et HN06 car chacun correspond à une situation dangereuse spécifique. HN02 correspond à la chute dans le même contexte que lors de l'utilisation d'un déambulateur classique (mauvaise utilisation, problème physiologique du patient, etc.). HN06 correspond à une chute provoquée par un dysfonctionnement du robot, alors que HN04 correspond à un dysfonctionnement du système d'alerte quelle que soit la raison de la chute de l'utilisateur. Le danger HN14 n'est pas présenté car il correspond à une utilisation du déambulateur hors du cadre de MIRAS (utilisation du robot en mode fauteuil roulant).

Il est important de mentionner que nous n'avons traité ici que les risques opérationnels, et pas les risques inhérents à l'utilisation d'une machine (ex : électrocution, explosion, émission de substances dangereuses, etc.). Pour ces aspects, les techniques et normes clas-

Project: MIRAS
HAZOP table number: UC12
Entity: UC12

Use case description

Use case name: Se lever à partir du siège robot

Precondition: - Le patient est assis sur le siège
- Les batteries sont suffisamment chargées pour réaliser cette tâche + déverticalisation

Postcondition: - Le patient est debout face au robot
- Le robot est prêt à déambuler

Invariant: - Les indicateurs physiologiques indiquent un état acceptable
- Base du robot immobile

Date: 29/04/10
Prepared by: Quynh Anh DO HOANG
Revised by:
Approved by:

Line Number	Element	Guideword	Deviation	Use Case Effect	Real World Effect	Severity	Possible Causes	Integrity Level Requirements	New Safety Requirements	Remarks	Hazard Number
1	Le patient est assis sur le siège (precondition)	No/none	Le patient n'est pas assis sur le siège du robot mais le robot pense que si	Le robot démarre la verticalisation	Le robot ne fait pas ce que le patient demande	Néant	Echec logiciel/matériel	Néant	Capteur de pression sur le siège		
2		Other than	ref L1								
3			Le patient est assis sur le siège mais le robot ne le détecte pas	Le robot est dans un mauvais mode	La base du robot peut bouger, provoquant la chute du patient	Sérieuse	Echec logiciel/matériel	SIL2 : SW/HW détection de position du patient	Le robot doit vérifier la position du patient avant d'effectuer la verticalisation		6
4		As well as	**N/A**								
5		Part of	Le patient est mal assis	Le robot passe en mode verticalisation du siège avec le patient mal assis	Déséquilibre/Chute du patient	Sérieuse	Echec logiciel/matériel Le robot ne détecte pas car le poids du patient est trop faible	SIL2 : SW/HW détection de position du patient	Ajouter un loquet pour bloquer le siège à l'horizontal Se renseigner sur le poids des patients		6

FIGURE 4.7 – Extrait d'une table HAZOP-UML appliquée à un cas d'utilisation

Line Number	Severity	New Safety Requirements	Hazard Number	Origine
1	Modérée	Déterminer les délais de réaction entre la détection et l'activation du mode déverticalisation sur siège du robot. Garantir la réaction dans les délais corrects.	12	UC13.SD01 Ligne 29
2	Modérée	Etablir les délais de réactions pour assurer la synchronisation correcte entre la descente des poignées et la déverticalisation du patient	13	UC03.SD02 Ligne 57
3	Modérée	La force de l'actionneur ne doit pas "éjecter" le patient.	12,13	UC12.SD01 Ligne 19,30
4	Modérée	Maintenir le blocage des roues pendant un temps MIN sans dépasser un temps MAX. Vérification de la position de la position du patient avant le déblocage.	12,13	UC12.SD01 Ligne 62,89
5	Modérée	Etude de la posibilité d'un profil de la déverticalisation sur siège.	2	UC13.SD01 Ligne 1,2,3

FIGURE 4.8 – Extrait de la liste des recommandations

siques pour les machines et les dispositifs médicaux électriques sont applicables, et sortent du cadre de cette thèse.

Un autre document produit lors de cette analyse est une liste de recommandations, extraite des tables HAZOP, avec des références vers les lignes correspondantes dans les tables HAZOP. Un extrait de cette liste est fourni Figure 4.8. L'ensemble des tables HAZOP est consultable en ligne [8].

4.4.2 Validation de l'approche HAZOP-UML

Afin de valider l'approche, et notamment l'utilisation des mots-guide plusieurs études comparatives ont été menées. Dans le cadre de cette thèse, une étude a été réalisée dans le cadre du projet MIRAS présenté précédemment mais nous utilisons également les résultats d'une HAZOP-UML effectuée dans le cadre du projet Européen PHRIENDS (2006). Ce projet avait pour objectif d'étudier aux différents niveaux d'une architecture robotique comment maîtriser la sécurité dans le cadre d'applications en contact physique avec l'humain. Notre cas d'étude a été un robot mobile équipé d'un bras, coopérant avec un ouvrier dans une usine pour effectuer des actions de *pick and place* (prendre un objet et le placer) en interaction avec l'opérateur (prendre un objet de la main de l'opérateur, ou lui donner, avec la possibilité pour l'opérateur de toucher le bras robot pour le stopper ou reprendre sa tâche).

Tout d'abord du point de vue qualitatif, cette méthode s'est complètement intégrée dans les processus de développement, grâce au partage des modèles de conception UML. Dans

8. http ://homepages.laas.fr/qdohoang/MIRAS/

les deux projets, les diagrammes de cas d'utilisation et de séquence UML ont été réalisés avec les développeurs des cas d'études. Dans MIRAS, un diagramme d'états-transition a été également réalisé. Le fait de baser cette approche sur UML a également permis de mettre à jour très rapidement les tables HAZOP-UML lorsque les scénarios d'utilisation ou la spécification des systèmes étudiés étaient modifiés (ce qui est arrivé plusieurs fois). En effet, chaque déviation étant reliée à un élément du modèle, toute modification pouvait être tracée et répercutée vers les tables HAZOP-UML. Le fait d'utiliser des méthodes standardisées telles que UML et HAZOP, et d'avoir réduit l'utilisation d'UML permet également très rapidement de prendre en main cette méthode.

Pour démontrer la validité de l'approche, nous avons comparé une analyse préliminaire des risques (notée APR) et HAZOP-UML. En effet, HAZOP-UML, tel qu'il est présenté ici, est une méthode permettant d'effectuer une analyse au tout début du processus de développement. C'est également le cas d'une APR. Cette comparaison du point de vue du nombre de dangers identifiés est donnée Figure 4.9. L'APR a été réalisée en premier, avec les partenaires du projet MIRAS, puis HAZOP-UML a été réalisée. Le résultat peut donc paraître biaisé (puisque l'étude APR a forcément influencé HAZOP-UML), mais les dangers identifiés lors d'HAZOP, également identifiés pendant l'APR, sont tous directement induits par l'utilisation d'un ou plusieurs mot-guides. Sur cette table, sont présentés les principaux dangers du projet MIRAS, ainsi que le nombre d'apparition de ces dangers dans l'application de ces deux méthodes. Nous n'avons noté aucun danger que seule l'APR aurait identifié. En revanche, HN2 et HN10, n'ont été identifiés que par notre méthode d'HAZOP-UML. Ces dangers n'ont rien de particulier, mais c'est en analysant des scénarios d'utilisation qu'ils ont émergé, alors que lors de l'APR, ils n'ont pas été identifiés. Un autre résultat issu de cette table, est la complémentarité des analyses basées sur les Cas d'utilisation (CU) et des diagrammes de séquence (Seq). Pour ce qui est de l'analyse du diagramme d'états-transition, aucun nouveau danger n'a été identifié, en revanche de nombreuses déviations qui n'avaient pas été envisagées ont été découvertes lors de cette phase, et ont donné de nombreuses nouvelles recommandations.

Plus spécifiquement sur l'analyse HAZOP du diagramme d'états transitions, nous avons relevé sur le tableau 4.1 pour chaque mot-guide le nombre d'interprétations (et d'interprétations avec recommendation) qui avaient été déduites. Cette étude a permis d'illustrer que la sélection des mots-guide réalisée était cohérente, et que les interprétations génériques présentées dans le chapitre 2 permettaient toutes d'identifier des déviations que nous avons retenues pour cette étude.

Du point de vue de l'applicabilité de la méthode, le tableau 4.2 présente le nombre d'interprétations (quand un mot-guide amène à une déviation possible) pour les deux projets MIRAS et PHRIENDS. Le nombre d'éléments étant du même ordre de grandeur (autour

HAZOP-UML Etats-Trans		**Déviation**	**Interprétation**	**Interprétation avec recommandation**
Élément	**Mot-guide**			
(destination state)	Other than	19	16	13
(transition)	Never	19	8	4
	No/None	19	15	9
(event)	No/None	19	17	12
	Other than (not triggered)	19	14	10
	Other than (triggered)	19	11	10
(condition)	No/None	17	17	11
	Other than	17	17	11
	Other than	17	17	11
	As well as	17	2	2
	Part of	17	3	3
	Early	17	0	0
	Late	17	13	11
(action)	No/None	19	19	16
	Other than	19	6	6
	As well as	19	2	2
	Part of	19	11	11
	Early	19	8	3
	Late	19	17	14
	More	19	1	1
	Less	19	1	1
	Total	**385**	**215**	**161**

TABLE 4.1 – Utilisation des mots-guide pour HAZOP-UML Statecharts

Num	Description	APR	HAZOP-UML		
			CU	Seq	Etats-Trans
HN1	Posture incorrecte du patient pendant l'utilisation du robot	2	4	3	4
HN2	Chute du patient pendant l'utilisation du robot (comme pour un déambulateur classique)		29	27	30
HN3	Arrêt total du robot pendant l'utilisation (absence d'énergie)	1	2		
HN4	Chute du patient sans alarme ou avec alarme tardive		11	13	32
HN5	Problème physiologique du patient sans alarme ou avec alarme tardive		15	10	
HN6	Chute du patient provoquée par le robot (mouvement non désiré du robot)	10	51	37	10
HN7	Incident détecté mais défaut de passage en mode sûr ; le robot continue à se déplacer		8		
HN8	Le robot coince un membre du patient (entre 2 parties du robot ou entre le robot et un	3	5	4	
HN9	Collision entre le robot (ou partie du robot) et le patient	2	14	14	
HN10	Collision entre le robot et une personne autre que le patient		5	14	2
HN11	Gêne du personnel médical pendant une intervention		1		
HN12	Déséquilibre du patient provoqué par le robot (sans chute)	11	1	70	1
HN13	Fatigue du patient dû à une mauvaise ergonomie ou commande du robot (sans chute)	12	1	53	21
HN15	Chute du patient depuis le siège	2	10	12	
HN16	Alarme trop fréquente (Faux positif)			3	

FIGURE 4.9 – Dangers identifiés par l'utilisation de l'analyse préliminaire des dangers (PHA), et HAZOP-UML

d'une centaine au total), le nombre de lignes de tables HAZOP (déviations interprétées) montre qu'il n'y a pas d'explosion combinatoire. Même si le nombre de déviations analysées semble important (1694 et 993), c'est tout à fait acceptable en comparaison avec d'autres analyses du risque et il est commun dans ce type d'analyse systématique d'obtenir ces ordres de grandeur. Sur le tableau 4.3 on retrouve également des ordres de grandeur qui sont maîtrisables en termes de ressources et d'outils disponibles.

Il est important de noter que la méthode qui est proposée et évaluée dans ce manuscrit s'inscrit à la fois dans les méthodes collaboratives et génératrice d'idées (*brainstorm*) mais également dans une approche systématique puisque basée sur une description des scénarios. Elle est également limitée à l'identification des dangers opérationnels liés à la réalisation de la tâche (vie opérationnelle du robot), mais ce sont ces dangers qui sont nouveaux par rapport aux dangers classiques propres aux «machines» et déjà couverts par de nombreuses normes (électriques, mécaniques, etc.).

4.4.3 Critères de risque

Cette section présente comment ont été déterminés les critères de risque (notamment les échelles de gravité et de probabilité), et l'estimation des risques du projet MIRAS.

	PHRIENDS	MIRAS
Cas d'utilisation	9	11
Conditions	39	45
Déviations analysées	297	317
Déviations interprétées	179 (60.3%)	134 (42.3%)
Déviations interprétées avec recommandation	120 (40.4%)	72 (22.7%)
Diagrammes de séquence	9	12
Messages	91	52
Déviations analysées	1397	676
Déviations interprétées	589 (42.2%)	163 (24.1%)
Déviations interprétées avec recommandation	274 (19.6%)	85 (12.6%)
Totaux		
Eléments UML	130	97
Déviations analysées	1694	993
Déviations interprétées	768 (45.33%)	297 (29.9%)
Déviations interprétées avec recommandation	394 (23.25%)	157 (15.8%)

TABLE 4.2 – Application de la méthode HAZOP-UML aux cas d'utilisation et aux diagrammes de séquence– Statistiques

	MIRAS
Diagramme d'état-transitions	1
Etats	9
Transitions	19
Déviations analysées	215
Déviations interprétées avec recommandation	161

TABLE 4.3 – Application de la méthode HAZOP-UML états-transitions – Statistiques

Niveaux de gravité	
Niveau	Description
Catastrophique	Entraîne le décès du patient
Critique	Entraîne une déficience permanente ou une blessure mettant en danger la vie du patient
Important	Entraîne une blessure (a) avec l'intervention de professionnels de la santé ou (b) entraînant une perte de confiance envers le système de la part du patient
Mineur	Entraîne une blessure temporaire (a) sans l'intervention de professionnels de la santé ou (b) entraînant une perte de confiance envers le système de la part du personnel médical
Négligeable	Nuisance ou gêne temporaire

TABLE 4.4 – Les degrés de gravité des dommages dans MIRAS

Cinq niveaux de gravité ont été choisis sur la base de tables équivalentes issues des normes, mais nous avons ajouté deux éléments spécifiques : une blessure entraînant une perte de confiance envers le système de la part du patient, dont le niveau de gravité est classé Important, alors que si la perte de confiance se situe du côté du personnel médical le niveau de gravité est jugé Mineur (table 4.4). Afin d'établir cette table, nous avons interagi avec les représentants des futurs utilisateurs de ces systèmes : les docteurs des 3 hôpitaux. En effet, les utilisateurs finaux (les patients), ont des pathologies trop lourdes (Alzheimer par exemple) pour donner des retours sur ces valeurs. Nous nous sommes basés sur la liste des dangers Figure 4.9, en effectuant une catégorisation des dommages potentiels, puis en établissant des groupes en niveaux considérés équivalent.

Autant pour les niveaux de gravité il a été assez simple de converger vers une liste commune, autant pour les niveaux de fréquence, l'exercice a été plus complexe. En effet, l'appréciation d'une fréquence n'a pas le caractère objectif de la conséquence d'un dommage (il existe même pour le domaine médical des catégorisations pour les blessures). Nous avons également pour cette étape utilisé la liste des dangers et effectué une identification conjointe niveaux de probabilité et niveaux d'acceptabilité. Pour cela nous avons demandé aux 3 partenaires du domaine médical, pour chaque danger, à quelle fréquence le danger serait-il acceptable, tolérable ou inacceptable, avec les définitions suivantes :

`Non acceptable`

le bénéfice lié à l'utilisation ne peut justifier le risque encouru

		Fréquence d'occurrence						
N° Danger	Gravité	Improbable	Rare	Peu Probable	Peu frequent	Moyennement Fréquent	Fréquent	Très fréquent
HN3	Critique	Accept.	Tolérable	Tolérable	Non Accept.	Non Accept.	Non Accept.	Non Accept.

FIGURE 4.10 – Exemple des niveaux d'acceptabilité du danger HN3

Fréquence d'occurrence	
Niveau	Fréquence
Très fréquent	1 fois/ semaine
Fréquent	1 fois/ mois
Moyennement Fréquent	1 fois / 6 mois
Peu fréquent	1 fois/ an
Peu probable	1 fois/ 10 ans
Rare	1 fois/ 100 ans
Improbable	Inférieur à 1 fois/ 100 ans

TABLE 4.5 – Les fréquences d'occurrence dans MIRAS

`tolérable`
: les coûts (financier, humains, organisationnel, ou techniques) ne permettent pas de réduire le risque, mais au regard du bénéfice apporté, il est toléré

`acceptable`
: le niveau de risque ne requiert aucune réduction supplémentaire

Ce travail a permis d'établir pour chaque partenaire une liste de 15 évaluations, ainsi que des valeurs pour le découpage en fréquence. Pour chaque hôpital, nous avons recueilli auprès des docteurs leur estimation :

— du niveau de gravité du dommage potentiel de chaque danger
— de l'acceptabilité du risque, en considérant cette gravité, en fonction de différentes fréquences d'occurrence

L'intégralité de ce travail n'est pas présenté car il a consisté à croiser les données (sous forme de tables excel), puis à harmoniser les réponses, parfois différentes, pour obtenir une version finale des fréquences d'occurrence et des niveaux d'acceptation. La table finale des niveaux de fréquence d'occurrence est donnée Table 4.5.

Après plusieurs itérations, nos partenaires se sont mis d'accord sur les niveaux d'acceptabilité en fonction des niveaux de gravité et de probabilité d'occurrence, récapitulés dans

la Figure 4.11. Chaque danger s'est vu attribuer un niveau de gravité de dommage associé, mais son niveau d'acceptabilité est fonction de sa probabilité d'occurrence.

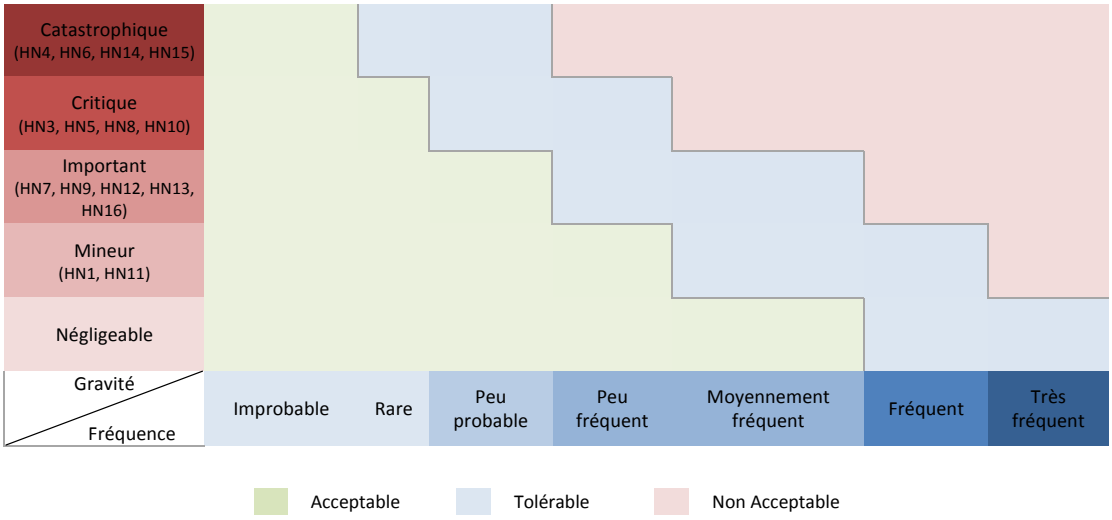


FIGURE 4.11 – Matrice harmonisée de l'acceptabilité des risques, avec les niveaux de gravité des dangers tels que perçus pour une utilisation finale du robot

Il est important de noter que cette matrice a été proposée pour l'utilisation d'un seul robot, en se basant sur une utilisation de 2h par semaine, 7j/7. La question du point de vue d'un service hospitalier utilisant une flotte de robot est différente, et il faudrait se placer dans un contexte différent pour revoir les fréquences d'occurrence notamment.

4.4.4 Estimation et évaluation du risque pour la version évaluation clinique

La version « eval » du robot (prototype pour évaluation clinique) consiste en un sous-ensemble des fonctionnalités visées dans le projet. En utilisant la représentation en cas d'utilisation exposée précédemment, cette version correspond à l'implémentation des cas suivants :

— le patient déambule avec le robot (UC01),
— le patient se verticalise à l'aide du robot (UC02),
— le personnel médical ou le patient positionne le robot à la main (UC07),
— le robot est configuré pour être utilisé par un patient (UC09),
— et le robot est installé par le personnel médical pour être utilisé (UC11).

Pour ce robot utilisé dans le cadre de l'hôpital, nous avons considéré le contexte suivant :

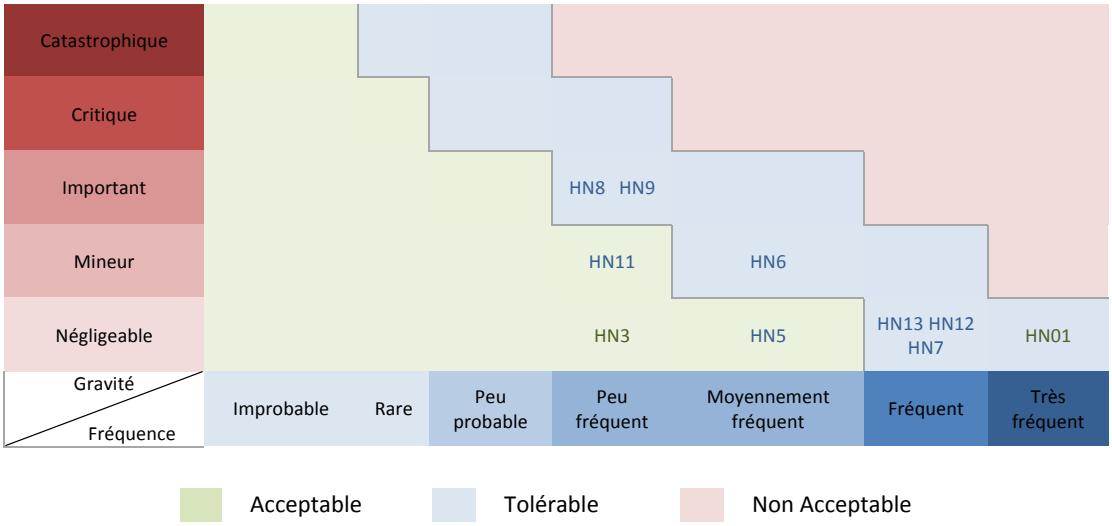


FIGURE 4.12 – Matrice harmonisée de l'acceptabilité des risques pour un robot

1. Surveillance permanente par un personnel médical ayant accès à un arrêt d'urgence
2. Possibilité d'intervention immédiate en cas de déséquilibre du patient

Pour les premières évaluations cliniques un professionnel de santé accompagne le patient tout le temps pendant l'utilisation du robot. Il est donc capable d'intervenir en cas de problème, en particulier pour soutenir les patients en cas de situation dangereuse (déséquilibre / chute / etc.). Le système doit comporter un arrêt d'urgence permettant de couper l'alimentation des moteurs de la base et des bras. Une personne de l'équipe technique du promoteur est en charge du bouton d'urgence (déporté). De cette façon, même les patients présentant les pathologies les plus graves peuvent participer aux évaluations.

Dans les conditions décrites ci-dessus, les risques évalués sont présentés dans la Figure 4.12. Par rapport à la matrice présentée Figure 4.11, des réductions de fréquences et de gravité ont été estimées en concertation avec les concepteurs du robot (ISIR). Plusieurs dangers ont également été retirés de la liste car le fait d'avoir une surveillance constante pendant les essais les rendaient non réalistes (par exemple « HN05 Problème physiologique du patient sans alarme ou avec alarme tardive » n'a plus de sens).

La Figure 4.13 présente pour les dangers considérés un résumé des justifications des risques présentés sur la Figure 4.12.

Num	Description	Justification niveau de risque
HN01	Posture incorrecte du patient pendant l'utilisation du robot	Détection par le personnel médical, réduction de la sévérité à négligeable, probabilité estimée à rare (le personnel intervenant sans délai)
HN02	Chute du patient pendant l'utilisation du robot (comme pour un déambulateur classique)	N/A Non évalué car danger identique avec déambulateur classique
HN03	Arrêt total du robot pendant l'utilisation (absence d'énergie)	Probabilité évaluée à peu fréquente, mais surtout l'impact devient négligeable
HN04	Chute du patient sans alarme ou avec alarme tardive	N/A Non applicable dans version « eval »
HN05	Problème physiologique du patient sans alarme ou avec alarme tardive	N/A Non applicable dans version « eval »
HN06	Chute du patient provoquée par le robot (mouvement non désiré du robot)	Cet événement passe à Mineur car le personnel est censé pouvoir intervenir en cas de déséquilibre. Sur une fréquence Moyenne, le risque devient tolérable
HN07	Incident détecté mais défaut de passage en mode sûr ; le robot continue à se déplacer	Les mécanismes de détection seront compensés par la surveillance du personnel médical. La gravité devient donc négligeable.
HN08	Le robot coince un membre du patient (entre 2 parties du robot ou entre le robot et un objet fixe)	Lors des tests, l'arrêt d'urgence permettra de prévenir un coincement. Le niveau de gravité a été descendu à important. Le phénomène restant peu fréquent, le risque devient tolérable.
HN09	Collision entre le robot (ou partie du robot) et le patient	Idem HN08
HN10	Collision entre le robot et une personne autre que le patient	N/A Non applicable dans version « eval »
HN11	Gêne du personnel médical pendant une intervention	N/A Non applicable dans version « eval »
HN12	Déséquilibre du patient provoqué par le robot (sans chute)	Le déséquilibre provoquant une perte de confiance est estimé négligeable pour des évaluations clinique
HN13	Fatigue du patient dû à une mauvaise ergonomie ou commande du robot (sans chute)	Le personnel médical doit intervenir au plus tôt, ce qui amène à une sévérité négligeable.
HN15	Chute du patient depuis le siège	N/A Non applicable dans version « eval »
HN16	Alarme trop fréquente (Faux positifs)	N/A Non applicable dans version « eval »

FIGURE 4.13 – Justifications des niveaux de risques pour la version «eval» du robot

4.4.5 Estimation et évaluation du risque pour la version finale

L'évaluation du risque pour la version « finale » du robot est plus complexe. En effet, sans les moyens de protection et de réaction qu'offre la présence de personnel médical et/ou technique, il faut évaluer précisément les fréquences d'occurrence pour décider de l'acceptabilité d'un risque. Cependant, cela n'est pas possible du fait d'éléments non encore implémentés, ainsi que de l'impossibilité d'obtenir certains taux de défaillance. Par exemple, il n'existe pas à l'heure actuelle de données sur les taux de chute, ou de déséquilibres, et il n'est pas envisageable d'estimer des taux de défaillance du logiciel développé dans le cadre du projet MIRAS.

Nous avons donc proposé de construire un argumentaire qui spécifie l'ensemble des preuves, et des solutions permettant de réduire les risques identifiés dans MIRAS. Cet argumentaire a été réalisé si une version complète du robot (version «finale») venait à être développée.

4.5 Argumentaire de sécurité du robot

Le système développé dans MIRAS doit permettre de se substituer à un déambulateur classique, mais est innovant par certaines fonctionnalités dont nous n'avons pas la maîtrise totale des technologies utilisées. C'est pourquoi nous proposons que la revendication de sécurité que le système doit atteindre soit la suivante : « G1 : Le robot est au moins aussi sûr qu'un déambulateur classique ». Afin d'argumenter, nous avons décomposé G1 en deux sous-revendications comme présenté Figure 4.14. La première (G1.1) consiste à traiter les risques qu'on peut également trouver dans un déambulateur classique, et la deuxième (G1.2) traite les risques qui sont propres au déambulateur robotisé développé dans le cadre de MIRAS. Nous argumentons ensuite que chacun des risques est correctement traité en fournissant, via le développement de l'argumentaire de sécurité, les éléments de preuve pertinents. La construction de cet argumentaire est basée sur le « GSN Hazard Avoidance Pattern » présenté sur la Figure 1.17. Les décompositions pour les nœuds G1.1 et G1.2 sont présentés Figure 4.15 et 4.16. L'ensemble des modèles GSN développés se trouvent en Annexe A.

Parmi différents nœuds développés, nous retrouvons :

— un pattern que nous réutilisons pour les objectifs de type « Le système de détection de faute est acceptable » qui se décompose par la suite en 3 revendications : absence de faute de conception, le taux de défaillance du système est acceptable et les imprécisions de la détection sont acceptables.

— plusieurs *Goal* supportés par un seul élément de preuve

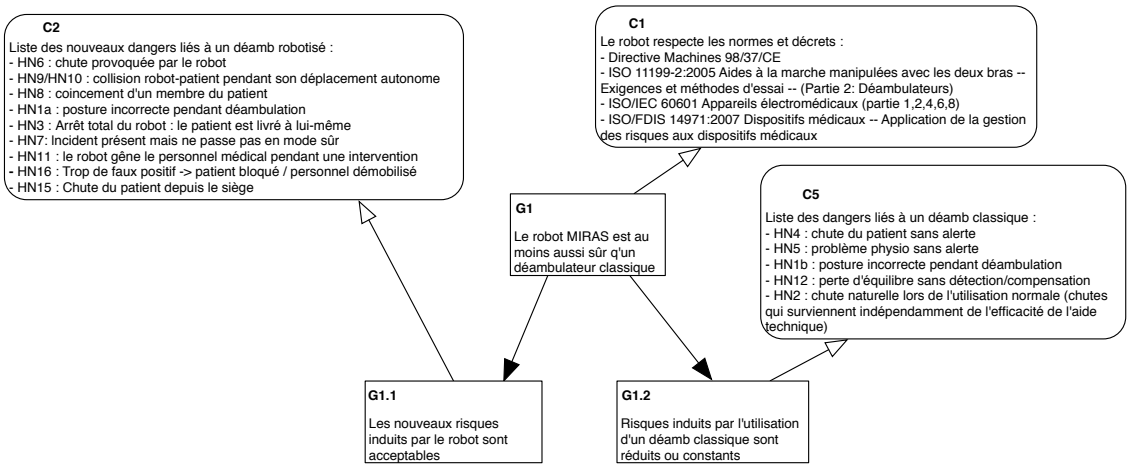


FIGURE 4.14 – Vue globale de l'argumentaire du robot déambulateur

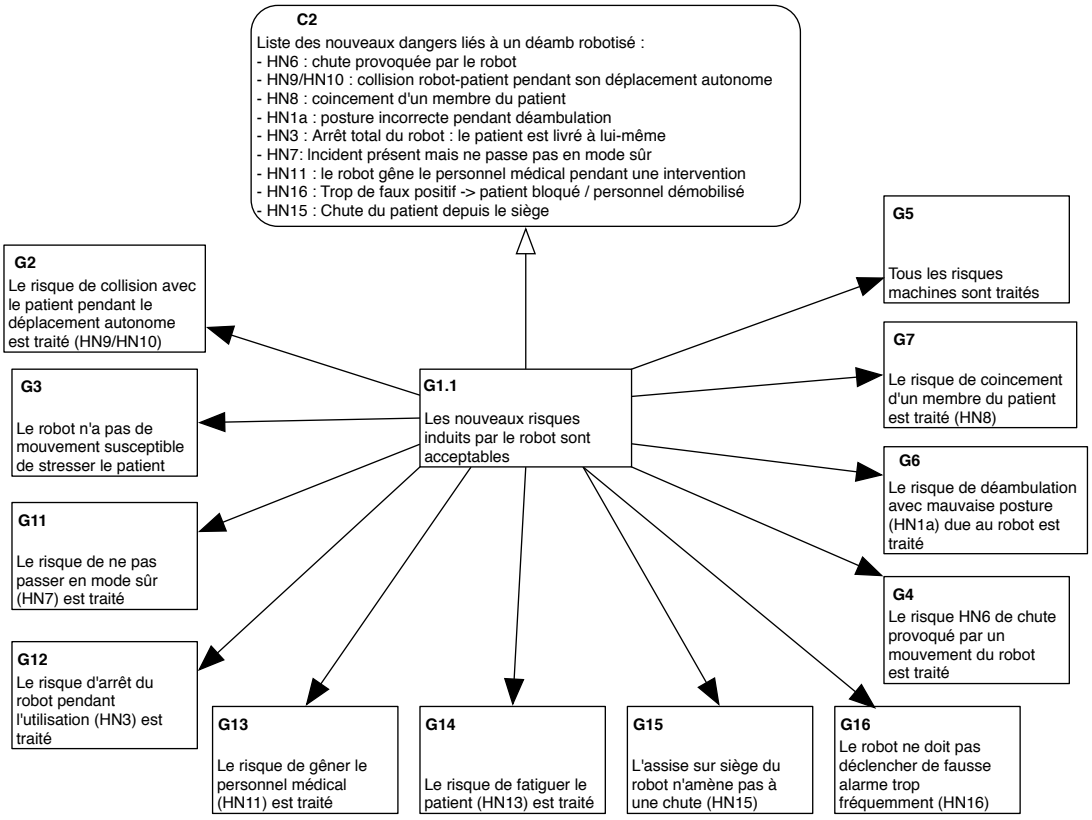


FIGURE 4.15 – Décomposition GSN du nœud G1.1

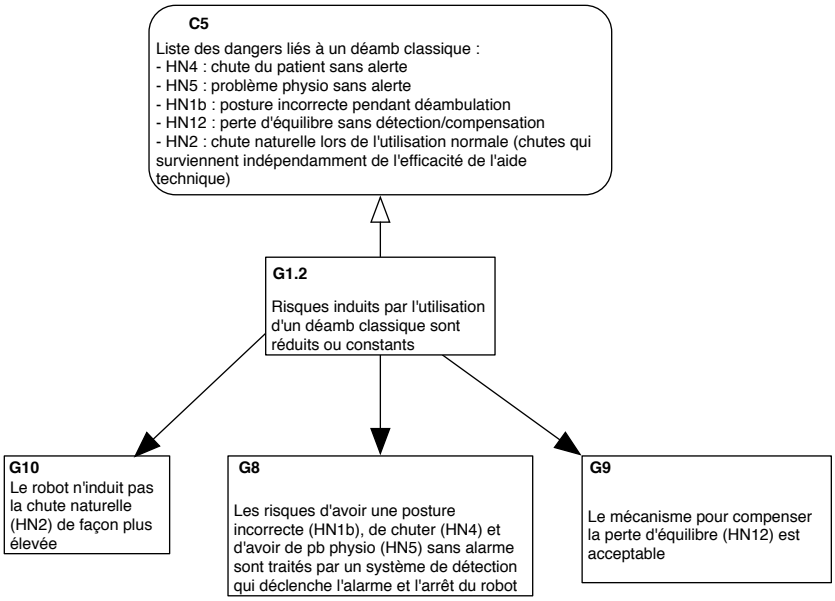


FIGURE 4.16 – Décomposition GSN du nœud G1.2

— la presque totalité des noeuds sont de type complémentaire.

En procédant à la décomposition des objectifs, nous arrivons à un niveau où une revendication peut-être prouvée par des faits et principalement :

— la conformité avec une norme
— des choix d'ergonomie du robot
— la mise en place de dispositifs de protection
— les résultats des tests fonctionnels satisfaisant les exigences requises
— les résultats des tests montrant l'efficacité de la fonction
— les résultats des tests avec injection de fautes
— le résultat des études analytiques comme les Arbres de fautes

Pour de nombreux éléments, on trouve également l'élément de preuve « Méthode de développement rigoureuse (exigence SIL 61508) ». Dans ce contexte, ces éléments correspondent à une démonstration de la bonne utilisation de méthodes de développement du logiciel selon le listing fourni dans la norme (IEC61508, 2010). En effet, le système informatique du robot se trouve dans le cadre d'application de la norme sur les logiciels médicaux (IEC62304, 2006). Cette norme définit une classification des composants logiciels, en fonction de la gravité du dommage que sa défaillance peut induire. Cette classification en trois niveaux (A, B et C) correspond à celle en 5 niveaux que nous avons utilisée dans MIRAS (présentée section 3.1) en utilisant les correspondances présentées dans le Tableau 4.6. La

Terme	Description possible	Classe CEI 62304
Catastrophique	Entraîne le décès du patient	C
Critique	Entraîne une déficience permanente ou une blessure mettant en danger la vie du patient	B
Important	Entraîne une blessure (a) avec l'intervention de professionnels de la santé ou (b) entraînant une perte de confiance envers le système de la part du patient	B
Mineur	Entraîne une blessure temporaire (a) sans l'intervention de professionnels de la santé ou (b) entraînant une perte de confiance envers le système de la part du personnel médical	A
Négligeable	Nuisance ou gêne temporaire	A

TABLE 4.6 – Correspondance des niveaux de Gravité dans la norme IEC 62304

Gravité	Norme 62304	Norme 61508
Catastrophique	C	SIL3
Critique	B	SIL2
Important	B	SIL2
Mineur	A	SIL1
Négligeable	A	SIL0

TABLE 4.7 – Correspondances de niveaux entre les normes IEC 62304 et IEC 61508

norme IEC 62304 ne propose pas directement des mesures ou techniques pour atteindre la classe fixée ; elle renvoie à la norme 61508-7 (Section présentation des techniques et mesures). A titre indicatif, nous établissons une correspondance entre les classifications de ces deux normes dans le Tableau 4.7. Les niveaux de SIL définis dans la norme 61508 sont numérotés de SIL1 à SIL4. Cependant, dans le cadre de ce robot, nous estimons que le niveau de SIL raisonnablement le plus élevé est SIL3, étant donné que la conséquence la plus grave est la chute du patient et non pas la mort directe.

Parmi les éléments de preuve de l'argumentation avec GSN (présentés dans l'annexe A), les fonctions logicielles critiques sont classées selon la norme IEC62304, il convient par la suite de chercher dans la norme 61508-7 les techniques requises pour le développement logiciel. Un exemple est donné Tableau 4.17. D'après ce tableau, pour une fonction logicielle de classe C (norme IEC 62304) qui est aussi de classe SIL3 (dans notre table d'équivalence), la simulation de processus est hautement recommandée [9].

9. la description détaillée de cette technique est expliquée dans IEC 61508-7 Annexe C.5.18)

Techniques/Mesures*		réf.	SIL 1	SIL 2	SIL 3	SIL 4
1	Essai probabiliste	C.5.1	---	R	R	HR
2	Simulation de processus	C .5.18	R	R	HR	HR
3	Modélisation	Tableau B.5	R	R	HR	HR
4	Essais fonctionnels et boîte noire	B.5.1 B.5.2 Tableau B.3	HR	HR	HR	HR
5	Traçabilité ascendante entre la spécification des exigences pour la sécurité du logiciel et le plan de validation de la sécurité du logiciel	C.2.11	R	R	HR	HR
6	Traçabilité descendante entre le plan de validation de la sécurité du logiciel et la spécification des exigences pour la sécurité du logiciel	C.2.11	R	R	HR	HR
NOTE 1 Voir Tableau C.7 NOTE 2 Les références (informatives et non pas normatives) « B.X.X.X », « C.x.x.x » dans la colonne 3 (réf.) désignent des descriptions détaillées de techniques/mesures données dans les Annexes B et C de la CEI 61508-7.						
* Les techniques/mesures appropriées doivent être sélectionnées en fonction du niveau d'intégrité de sécurité						

FIGURE 4.17 – Extrait de la norme IEC61508

4.6 La confiance dans l'argumentaire de sécurité

L'argumentaire de sécurité du robot déambulateur étant réalisé, il convient de poser la question sur la confiance que nous pouvons accorder à cet argumentaire. Notre étude de cas de la confiance dans l'argumentaire de sécurité a été réalisée sur la base d'estimations car le projet n'est pas allé jusqu'à la version finale, et donc les éléments de preuve n'ont pas été développés (par exemple tous les arbres de fautes, ou tous les tests spécifiés dans l'argumentaire n'ont pas été réalisés). Cependant, afin de valider notre approche, nous avons conduit l'étude en fixant des valeurs cohérentes sur les confiances afin de savoir si la méthode est pertinente et si la complexité du réseau était maîtrisée pour l'utilisation d'outils. Nous rappelons également que le processus pour l'estimation des valeurs de confiance est une problématique importante mais qui n'est pas abordée dans ce manuscrit.

4.6.1 Construction du réseau de confiance

Nous avons suivi la méthode de conversion présentée section 3.3 pour convertir systématiquement la notation GSN vers le réseau de confiance. Par exemple pour l'argumentaire portant sur G3 « Le robot n'a pas de mouvement susceptible de stresser le patient » : le nœud G3.1 n'a pas besoin d'être représenté car l'inférence G3.1 vers G3 est supposée parfaite (hypothèse choisie pour simplification), l'argumentaire est de type alternative car les

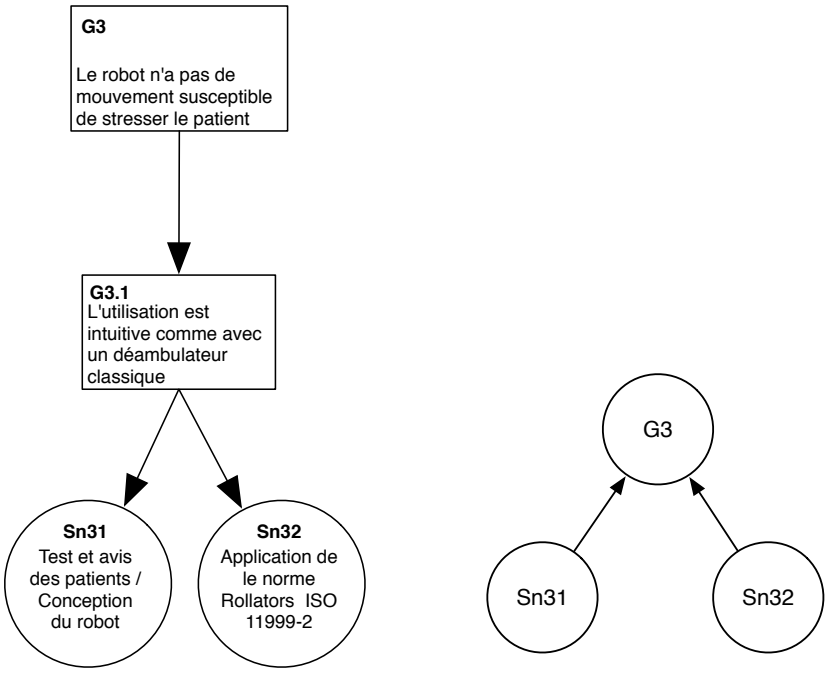


FIGURE 4.18 – L'argumentaire en notation GSN de G3 converti en réseau de confiance équivalent

tests peuvent suffire eux-mêmes pour conclure (l'application de la norme n'étant pas obligatoire). Nous obtenons ainsi sa conversion en réseau de confiance présentée Figure 4.18.

Nous notons également que les éléments de type justification dans la structure GSN n'influencent pas la confiance et sont simplement supprimés du réseau de confiance. En procédant de la même manière nœud par nœud, nous arrivons à déployer entièrement le réseau de confiance équivalent de l'argumentaire du sécurité du robot déambulateur comme présenté sur la Figure 4.19. Ce diagramme n'est pas lisible en l'état mais il permet d'illustrer la complexité du réseau.

4.6.2 Choix de l'outil AgenaRisk

À partir du réseau de confiance réalisé dans la section 4.6, nous pouvons attribuer des valeurs aux nœuds dits « feuilles » afin d'observer la propagation de la confiance placée dans les éléments de preuve vers l'objectif principal (ici « Le robot MIRAS est au moins aussi sûr qu'un déambulateur classique »). Dans le cadre de notre étude, nous avons utilisé l'outil AgenaRisk pour calculer cette propagation. Cet outil est développé pour modéliser, analyser

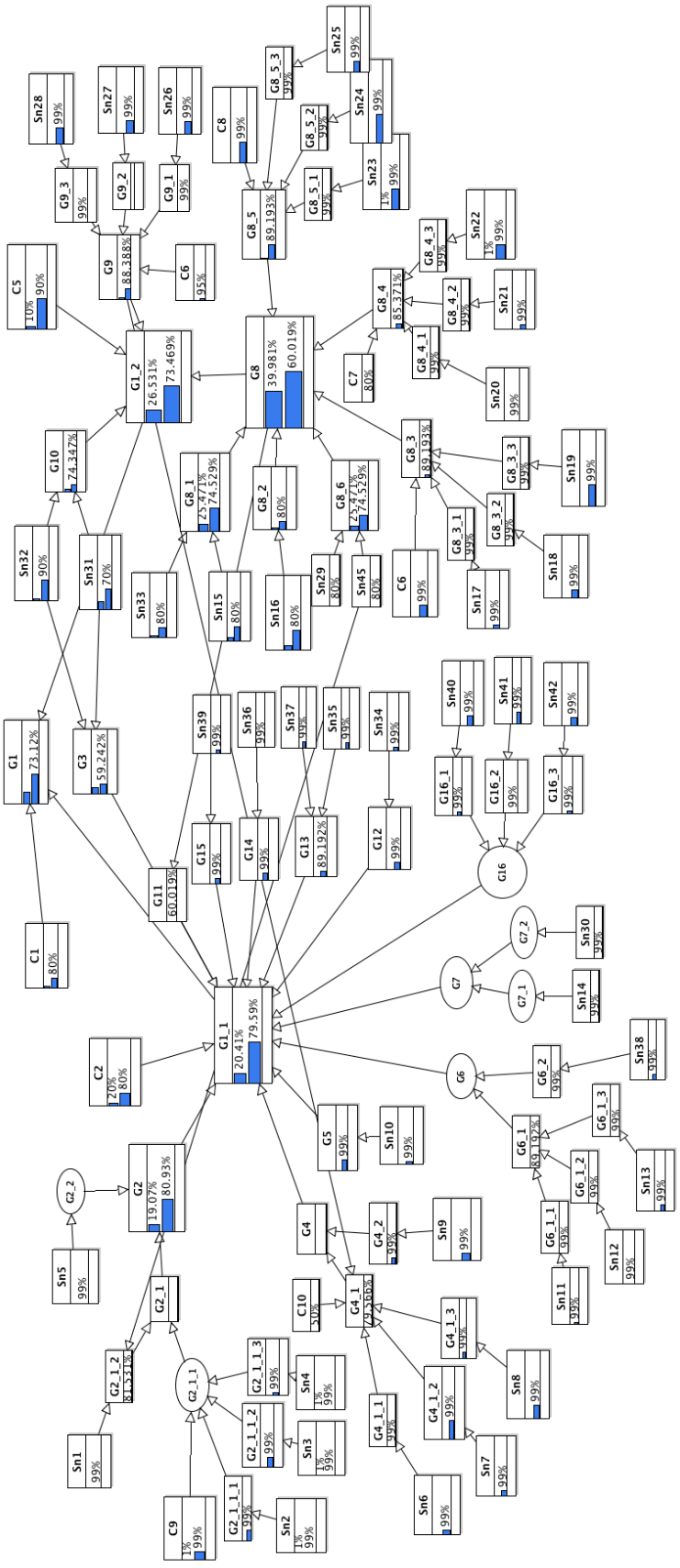


FIGURE 4.19 – L'argumentaire de confiance en notation GSN du robot déambulateur

des modèles de risque avec un réseau d'inférence de type Bayésien. L'outil propose par exemple de remplir les tables de confiance avec des formules *Noisy-OR*, *Noisy-AND* et aussi de générer le graphe de tornade pour l'étude de sensibilité (présenté dans la section 4.6.3).

Dans le cas d'un argumentaire de type alternatif, nous utilisons la formule de *Leaky Noisy-OR* avec la syntaxe :

$$noisyor(parent1, valeur1, parent2, valeur2, ..., parentN, valeurN, leak)$$

où $parentX$ est l'identificateur du nœud parent, $valeurX$ la confiance dans l'inférence du nœud parent vers le nœud fils et $leak$ l'incertitude dans l'argumentaire.

Dans le cas d'un argumentaire de type complémentaire, nous avons utilisé à titre d'exemple la formule *Leaky Noisy-AND* que propose l'outil qui donne sensiblement les mêmes résultats de nos calculs théoriques présentés dans la section 3.3.6 avec la syntaxe :

$$noisyand(parent1, valeur1, parent2, valeur2, ..., parentN, valeurN, leak)$$

où $parentX$ est l'identificateur du nœud parent, $valeurX$ le poids de l'inférence du nœud parent vers le nœud fils et $leak$ l'incertitude dans l'argumentaire.

Une fois les tables renseignées à l'aide de ces formules, le calcul est automatique et simple. La durée de calcul de notre réseau est de l'ordre de quelques minutes, une durée admissible pour cette étude. Cependant, le graphe de tornade que génère l'outil AgenaRisk ne fait varier que de la confiance dans les nœuds « feuille » et pas la confiance dans les inférences menant d'un nœud parent à un nœud fils comme nous l'avons vu précédemment. Cependant l'utilisation de ce logiciel nous a permis de valider la transposition de notre approche à un outillage. Les résultats obtenus à l'aide de l'outil sont présentés dans la section suivante.

4.6.3 Étude de la sensibilité de la confiance

En se basant sur le réseau de confiance résultant de la section 4.6.2, nous avons utilisé aussi l'outil AgenaRisk pour le calcul et la génération du graphe de tornade. Ainsi, la Figure 4.20 présente le réseau d'inférence complet du robot déambulateur à partir duquel une étude de sensibilité a été réalisée sur le nœud G1 (« Le robot MIRAS est au moins aussi sûr qu'un déambulateur classique »). Pour chaque nœud est affiché en premier l'incertitude puis la confiance. Par exemple, la confiance de G1 est de 73,12%. Tous les nœuds n'ont pas été agrandi par souci de visibilité. Dans ce réseau de confiance, nous retrouvons l'interconnexion des éléments servant à supporter différents nœuds à la fois :

— Le nœud G9 supporte à la fois les nœuds G1-2 et G4-1

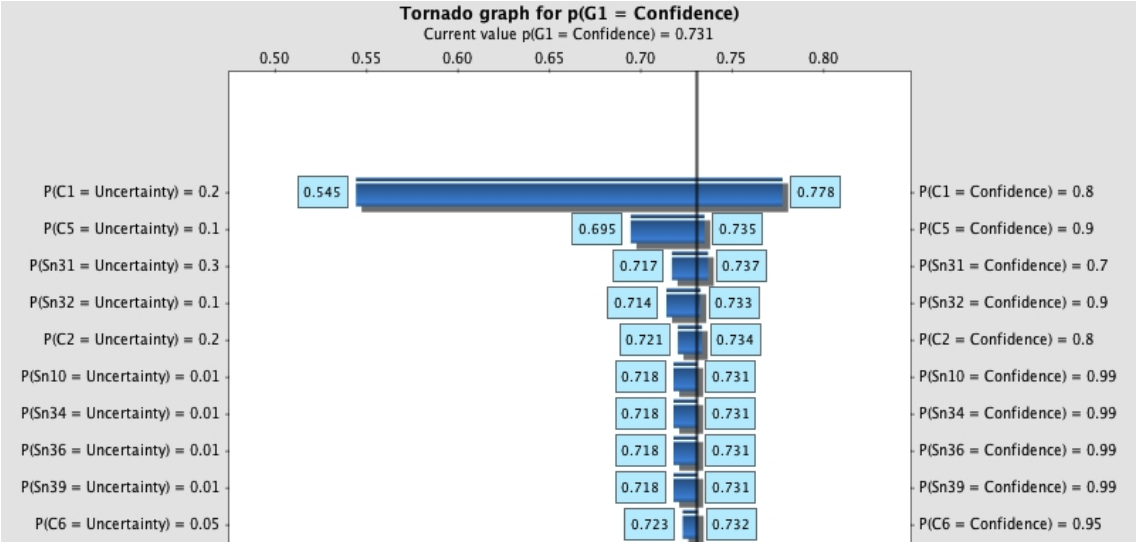


FIGURE 4.20 – Diagramme de tornade généré par AgenaRisk, présentant la sensibilité de la confiance dans l'argumentaire (seules les 10 nœuds feuilles les plus influents sont représentés)

— Les éléments de preuve Sn31 (« Test et avis des patients/Conception du robot ») et Sn32 (« Application de la norme Rollators ISO 11999-2 ») supportent simultanément le nœud G10 (« Le robot n'induit pas la chute naturelle de façon plus élevé ») et le nœud G3 (« Le robot n'a pas de mouvement susceptible de stresser le patient »)

— Le nœud G8 (« Les risques d'avoir une posture incorrecte (HN1b), de chuter (HN4) et d'avoir des problèmes physiologiques (HN5) sans alarme sont traités par un système de détection qui déclenche l'alarme et l'arrêt du robot) ») supporte à la fois G11 (« Le risque de ne pas passer en mode sûr ») et G1-2 (« Les risques induits par l'utilisation d'un déambulateur classique sont réduits ou constants »)

— L'élément de preuve Sn45 (« Tests démontrant que l'arrêt temporaire n'induit pas de dangers ») supporte à la fois G8-6 (« La détection d'un problème entraîne toujours le déclenchement de l'alarme et l'arrêt temporaire du robot ») et G2-1-2 (« La commande d'arrêt est obligatoire suite à une détection d'obstacle »)

Dans l'objectif de valider l'aspect calculatoire de la méthode, nous avons choisi de manière arbitraire les valeurs de la confiance que nous plaçons dans les éléments de preuve :

— les éléments de type « la présence de certain système de contrôle » ou « le suivi d'un processus de développement » ou un contexte donné sont des éléments irréfutables, la confiance que nous choisissons d'y placer est de 99%

— les éléments de type « résultats de test » comportent une incertitude que nous estimons à 20% (soit une confiance à 80%).
— un choix d'attribuer des poids équivalents à chacune des inférences dans le cas d'un argumentaire de type complémentaire, en respectant la contrainte que leur somme soit égale à la confiance globale de l'argumentaire
— un choix d'attribuer la même confiance à chacune des inférences dans le cas d'un argumentaire de type alternatif

Le calcul de la confiance que nous pouvons placer dans l'objectif « Le déambulateur MIRAS est au moins aussi sûr qu'un déambulateur classique » donne 73,12%, mais ce chiffre n'est en réalité qu'une valeur de référence, à partir de laquelle on peut réaliser l'étude de sensibilité. Le graphe de tornade généré à partir de ce réseau de confiance est présenté Figure 4.20. Comme indiqué précédemment, les valeurs numériques ne correspondent pas exactement à notre modèle du à l'utilisation d'AgenaRisk, mais notre objectif étant l'étude de sensibilité, le travail d'adaptation du logiciel n'a pas été notre priorité. Ce graphe met néanmoins en valeur, parmi les contextes et les éléments de preuve, les éléments qui influencent le plus la confiance dans G1. Nous observons par exemple une large variation (donc grande influence) associée au contexte C1, qui est due au fait que C1 est directement connecté au nœud G1 alors que les autres contextes et éléments de preuves se trouvent à différents niveaux du réseau. Ce travail a permis d'illustrer la faisabilité de la transposition de notre approche à l'utilisation d'un outil mais un travail important de validation reste à réaliser dans la continuité de ce travail.

4.7 Conclusion

La méthode que nous avons proposée permet de couvrir différentes étapes importantes du processus de gestion du risque d'un système et de l'évaluation de la confiance que l'on peut placer dans l'argumentaire de sécurité associé. La méthode inclut les activités suivantes : l'identification de danger, l'évaluation de danger, l'estimation de la confiance et l'étude de la sensibilité de la confiance d'un système. À travers l'application de cette méthode à un système robotique en interaction avec l'humain, notre objectif a été de tester la faisabilité sur un cas réel de ce cycle complet de la gestion du risque.

Au vu des résultats obtenus, la méthode nous semble pertinente et efficace. La méthode HAZOP-UML avec les trois types de diagrammes considérés (cas d'utilisation, diagramme de séquence et diagramme d'état-transition) s'avère bien adaptée aux systèmes robotiques en interaction avec l'humain comme celui développé dans le cadre du projet MIRAS. L'analyse du risque a permis d'en identifier plusieurs qui viennent s'ajouter à la liste des risques machines connus et qui servent comme données pour la génération de l'argumentaire. Nous

avons aussi observé que les trois types de diagrammes fournissent des informations complémentaires. Leur utilisation de façon conjointe est pertinente pour aboutir à une couverture plus large des différentes types de dangers à prendre en compte. En déployant cette méthode pour ce cas concret, nous constatons que pour les 16 dangers identifiés, la taille de l'argumentaire reste exploitable et ne conduit pas à une explosion combinatoire d'éléments dans la structure. Ceci est très important car la complexité de l'argumentaire reflète également celle du calcul de la confiance réalisée dans l'étape suivante.

Plus précisément, dans cette application, l'expérimentation n'est validée que partiellement. Alors que la liste des dangers et la construction de l'argumentaire de sécurité ont été validées avec nos partenaires, la plupart des confiances que nous plaçons dans divers éléments du réseau de confiance sont purement hypothétiques à ce stade de nos travaux. Le développement du robot dans le cadre du projet MIRAS n'étant pas finalisé, certains résultats de tests ne sont pas encore disponibles. Sur certaines parties de l'argumentaire, nous pouvons fournir des estimations proches mais sur d'autres parties, nous avons choisis arbitrairement des valeurs qui nous semblent cohérentes. Il est important de rappeler que le résultat des calculs réalisés n'a pas pour vocation de refléter la valeur réelle de la confiance que l'on peut placer dans le système. L'objectif est plutôt d'apporter un support et des outils adaptés pour interpréter et identifier les éléments les plus sensibles d'un argumentaire de sécurité, afin de développer ensuite des solutions adéquates permettant de renforcer la confiance dans le système.

Synthèse, contributions majeures et perspectives

Synthèse

Avec l'augmentation du nombre et de la diversité des systèmes de robotique de service, le besoin de méthodologies d'analyse de la sécurité liée à leur utilisation en contact avec les humains se fait plus pressant. Les normes existantes n'étant pas applicables à l'utilisation de ce type de systèmes, il est donc difficile de valider ou de certifier leur développement. Les travaux de recherche présentés dans ce manuscrit consistent à enrichir le processus de gestion de risques par une approche basée modèle, ainsi qu'à étendre l'utilisation des argumentaires de sécurité avec un modèle de confiance dans cet argumentaire.

Nous avons commencé par présenter la problématique autour de la gestion du risque d'un système robotique en interaction avec l'humain, puis nous avons détaillé notre approche en deux parties : la première présente la méthode d'analyse du risque HAZOP-UML, qui consiste à identifier les dangers en se servant des modèles UML du système ; la deuxième partie consiste à construire un argumentaire pour justifier la sécurité du système à partir de divers éléments de preuve comme les résultats de tests, les taux de défaillance, le suivi des processus de développement rigoureux... Nous avons proposé de convertir cet argumentaire en un réseau de confiance (avec deux types d'arguments), et d'attribuer une valeur de confiance dans les éléments feuilles. À partir de ces valeurs, il est possible de faire propager la confiance en suivant des modèles de calcul adaptés à chaque type d'argument (complémentaire ou alternatif). Nous obtenons ainsi une confiance de l'argument qui est sensible à la confiance placée dans les éléments de preuve que nous pouvons visualiser à l'aide d'un graphe de tornade. Ces étapes s'insèrent dans un cycle complet de gestion du risque allant de l'identification du risque puis l'évaluation de risque pour aller jusqu'à l'argument de sécurité et l'analyse de la sensibilité. Ce cycle complet a été appliqué au cas concret du développement d'un robot déambulateur.

Contributions majeures

La première contribution consiste à renforcer la technique d'analyse du risque HAZOP-UML, notamment en ajoutant un troisième type de diagramme dans l'étude de l'analyse du risque. La technique HAZOP-UML initiée avant les travaux de cette thèse ne prenaient en compte que deux types de diagrammes : le diagramme de cas d'utilisation et le diagramme de séquence. Nous avons étendu cette technique à un troisième type de diagramme : le diagramme d'état-transitions. Les travaux effectués ont aidé à identifier les éléments du diagramme d'état-transitions susceptibles d'être combinés avec des mots-guides pour former une déviation possible. Ensuite, l'analyse du risque conduite sur ce nouveau type de diagramme nous a permis d'identifier les nouvelles sources de dangers. Nous avons également montré comment cette technique s'insérait complètement dans le processus de gestion du risque, et donné des résultats sur l'applicabilité, la validité, et l'intérêt de l'approche HAZOP-UML. Un point majeur est qu'il s'agit d'une méthode systématique basée sur des modèles du système, et s'insérant donc complètement dans un processus de développement, ce qui n'est pas le cas de la plupart des méthodes d'analyse du risque actuelles.

La deuxième contribution porte sur la transformation de l'argumentaire de sécurité en réseau de confiance et sur l'étude de sensibilité de cette confiance. Celle-ci propose de combiner plusieurs techniques connues pour proposer une étude de confiance dans le système : nous avons commencé par construire un argumentaire de sécurité, puis nous avons proposé une méthode de conversion vers un réseau de confiance et un modèle de calcul de la propagation de la confiance placée dans les éléments de preuve en fonction de différents types d'argument (alternatif ou complémentaire). Cette confiance est ensuite soumise à une étude de sensibilité qui nous permet de connaître les éléments les plus perturbants du système afin de prendre des mesures nécessaires. Cette approche à l'avantage majeur de se baser sur la théorie de la croyance, et donc de faire apparaître explicitement la notion d'incertitude, tout en restant dans un cadre calculatoire où les outils pour les réseaux Bayésiens sont utilisables.

Perspectives

Les travaux menés dans le cadre de cette thèse ont permis d'établir une méthode complète allant de l'identification de dangers jusqu'à l'argumentation sur la sécurité du système et la confiance que l'on peut placer dans cette argumentation. Cependant, certains points nécessitent un développement supplémentaire, et d'autres n'ont pas été traités dans cette thèse alors qu'ils sont nécessaires à l'application de notre approche.

Dans l'approche présentée, l'application de la méthode HAZOP-UML et la construction de l'argumentaire de sécurité sont indépendantes. Une amélioration serait de créer un lien

de conversion systématique qui relie le ces deux techniques. Ainsi à partir des dangers identifiés, ainsi que des déviations possibles, un argumentaire de sécurité pourrait être généré, ou du moins en partie. Cette possibilité faciliterait la construction de l'argumentaire et permettrait également d'assurer une traçabilité entre les traitements des risques tels que présentés dans l'argumentaire, et les déviations induisant ces risques.

Pour chaque type d'argument (complémentaire ou alternatif), nous avons proposé des modèles de calcul cohérents avec le comportement souhaité de l'évolution de la confiance dans l'argument. Cependant, dans le cas d'un argument de type complémentaire, le modèle proposé pourrait être amélioré. Une piste possible est de s'orienter vers des calculs basés sur la théorie de la croyance.

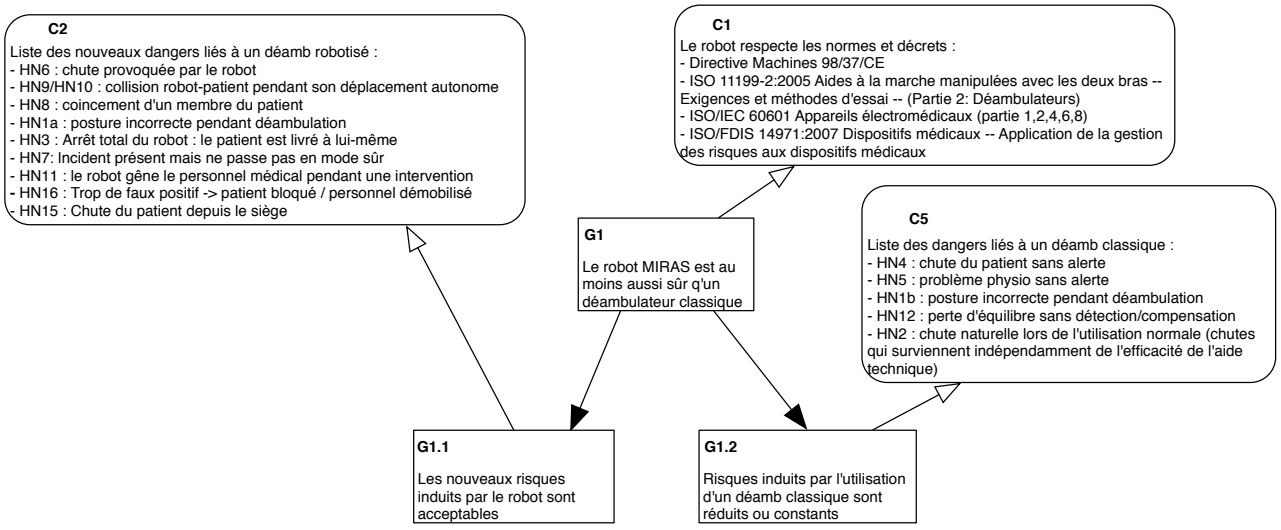
L'étude de la confiance d'un argumentaire de sécurité a été déployée intégralement sur un cas concret, mais toutes les valeurs de confiance ont été fixées afin de valider l'approche théorique de la méthode. Il y a donc un besoin de méthode pour obtenir ces valeurs de confiance, que ce soit au niveau des inférences ou des éléments de preuves ou des contextes eux-mêmes.

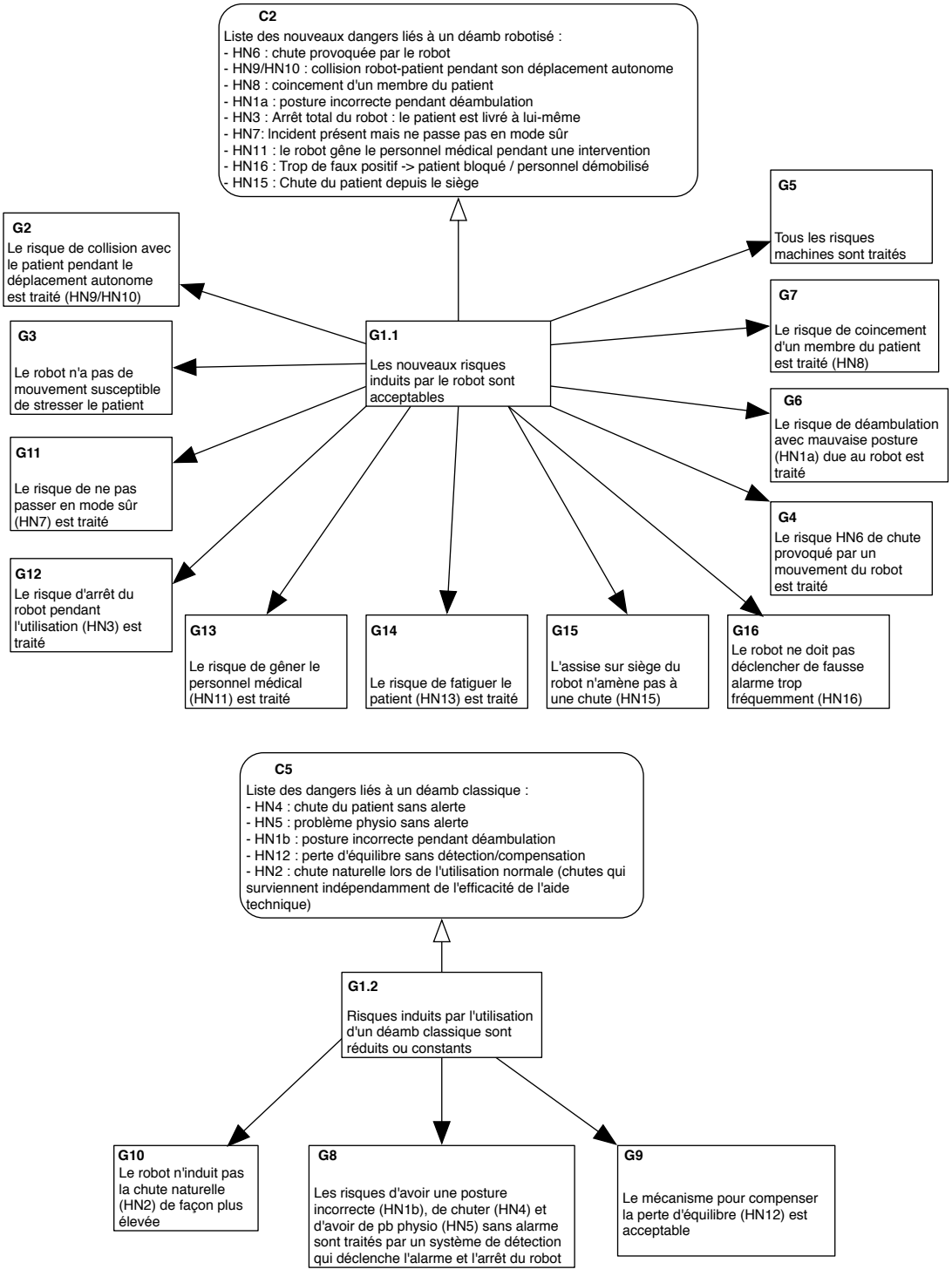
L'outil AgenaRisk nous a permis d'effectuer une étude de sensibilité, c'est-à-dire d'identifier les éléments ayant le plus d'impact sur la confiance globale. Cependant cette étude doit être enrichie afin d'identifier également d'autres caractéristiques d'un argumentaire, comme par exemple comparer différentes versions d'un argumentaire, identifier les branches «faibles» au niveau de la confiance.

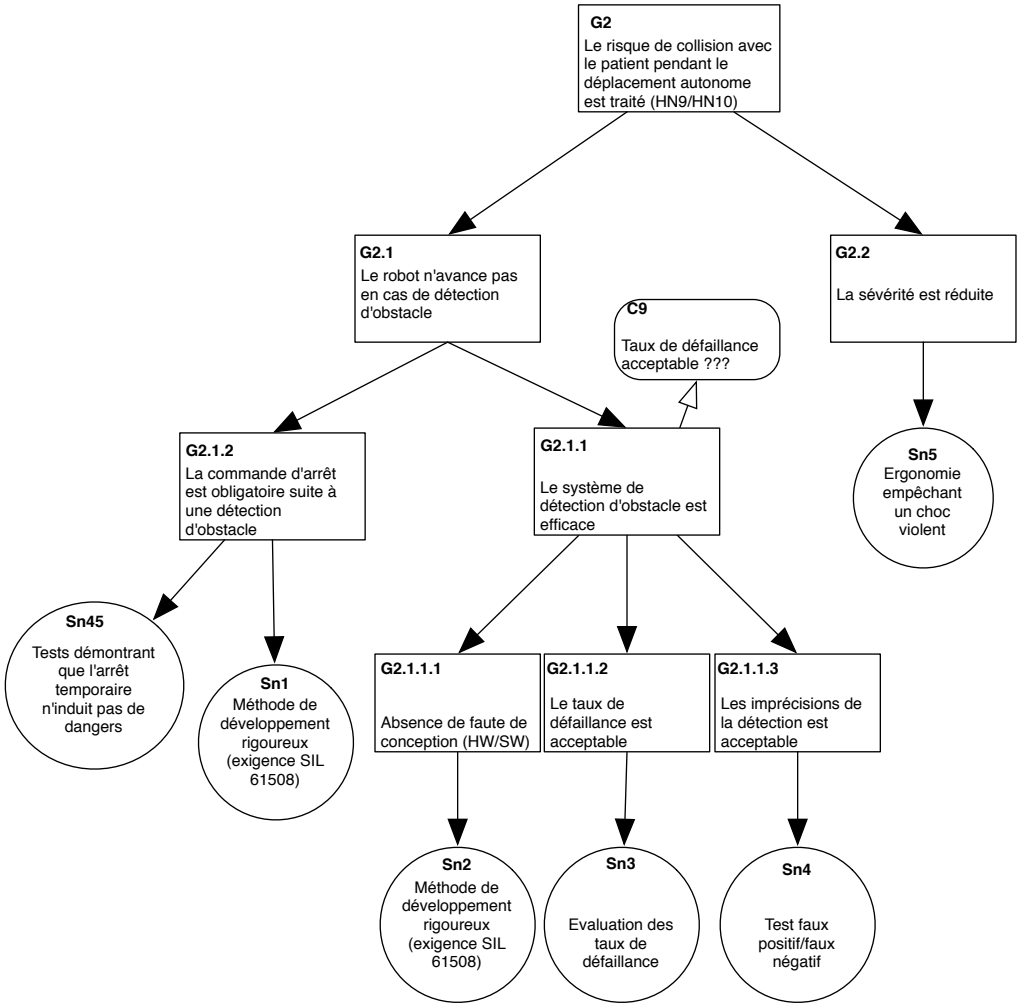
Enfin, il sera important de voir comment notre approche contribue à l'amélioration des argumentaires de sécurité dans le cas de robotique de service comportant des mécanismes décisionnels. En effet, pour ces systèmes la complexité de la perception, et le caractère non-déterministe des heuristiques des planificateurs de tâches ou de mouvement, rendent difficile la justification de la sécurité dépendante de nombreuses incertitudes. L'utilisation de notre approche pourrait par exemple permettre de garantir un niveau de confiance en maîtrisant ces incertitudes.

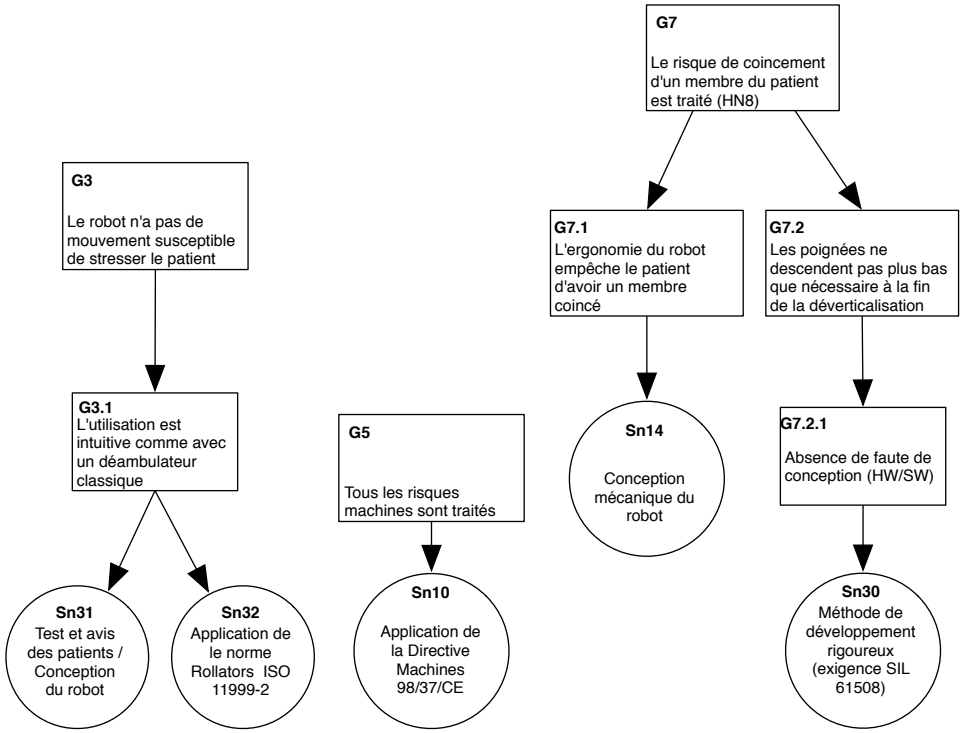
Annexe A

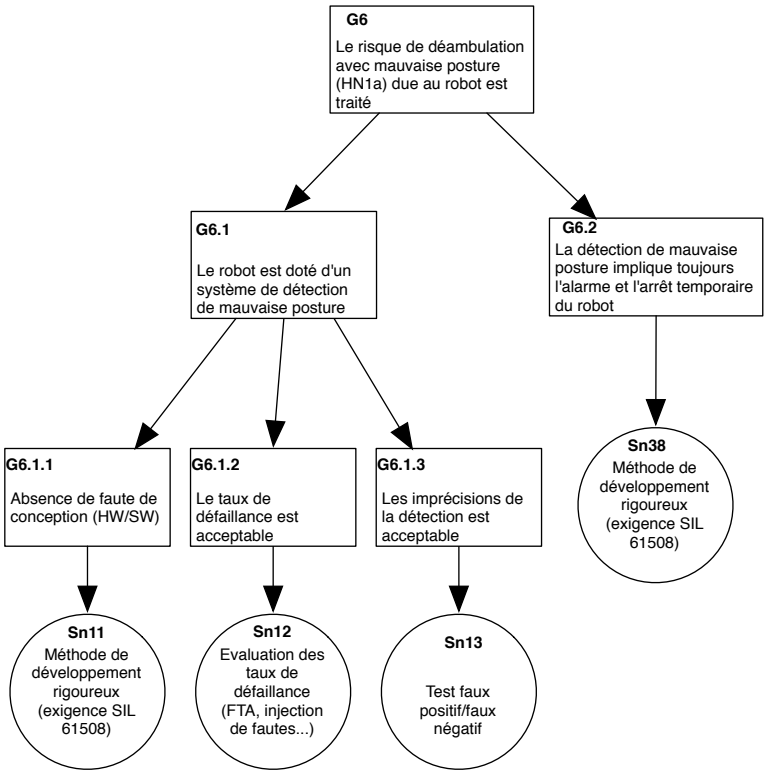
Modèle GSN de MIRAS

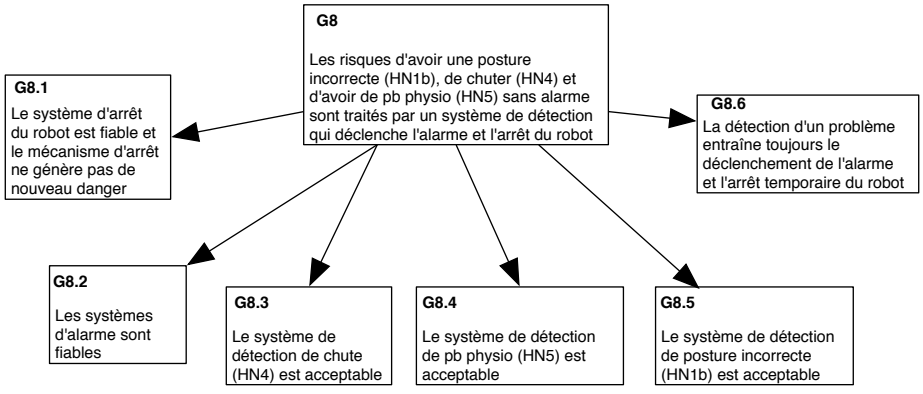


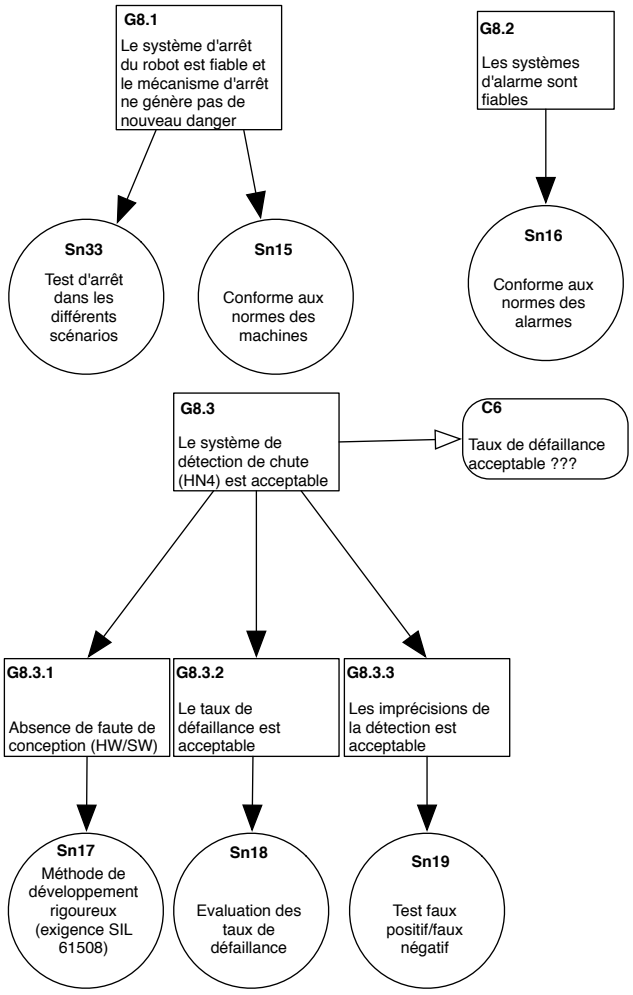

C2
Liste des nouveaux dangers liés à un déamb robotisé :
- HN6 : chute provoquée par le robot
- HN9/HN10 : collision robot-patient pendant son déplacement autonome
- HN8 : coincement d'un membre du patient
- HN1a : posture incorrecte pendant déambulation
- HN3 : Arrêt total du robot : le patient est livré à lui-même
- HN7: Incident présent mais ne passe pas en mode sûr
- HN11 : le robot gêne le personnel médical pendant une intervention
- HN16 : Trop de faux positif -> patient bloqué / personnel démobilisé
- HN15 : Chute du patient depuis le siège
G2
Le risque de collision avec le patient pendant le déplacement autonome est traité (HN9/HN10)
G3
Le robot n'a pas de mouvement susceptible de stresser le patient
G11
Le risque de ne pas passer en mode sûr (HN7) est traité
G12
Le risque d'arrêt du robot pendant l'utilisation (HN3) est traité
G1.1
Les nouveaux risques induits par le robot sont acceptables
G5
Tous les risques machines sont traités
G7
Le risque de coincement d'un membre du patient est traité (HN8)
G6
Le risque de déambulation avec mauvaise posture (HN1a) due au robot est traité
G4
Le risque HN6 de chute provoqué par un mouvement du robot est traité
G13
Le risque de gêner le personnel médical (HN11) est traité
G14
Le risque de fatiguer le patient (HN13) est traité
G15
L'assise sur siège du robot n'amène pas à une chute (HN15)
G16
Le robot ne doit pas déclencher de fausse alarme trop fréquemment (HN16)
C5
Liste des dangers liés à un déamb classique :
- HN4 : chute du patient sans alerte
- HN5 : problème physio sans alerte
- HN1b : posture incorrecte pendant déambulation
- HN12 : perte d'équilibre sans détection/compensation
- HN2 : chute naturelle lors de l'utilisation normale (chutes qui surviennent indépendamment de l'efficacité de l'aide technique)
G1.2
Risques induits par l'utilisation d'un déamb classique sont réduits ou constants
G10
Le robot n'induit pas la chute naturelle (HN2) de façon plus élevée
G8
Les risques d'avoir une posture incorrecte (HN1b), de chuter (HN4) et d'avoir de pb physio (HN5) sans alarme sont traités par un système de détection qui déclenche l'alarme et l'arrêt du robot
G9
Le mécanisme pour compenser la perte d'équilibre (HN12) est acceptable

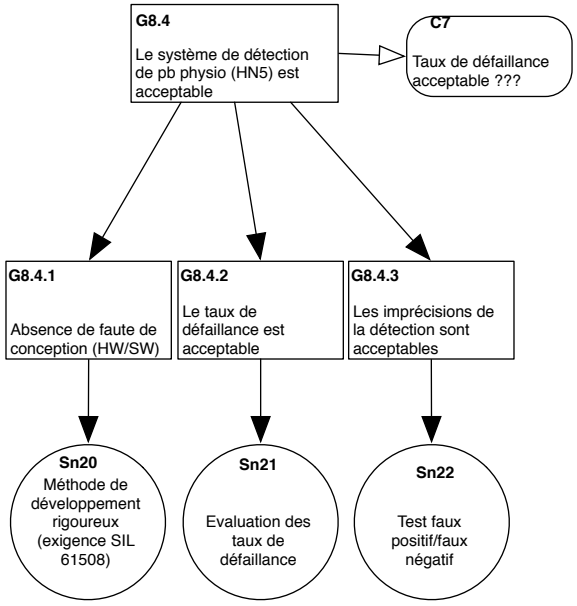

G2
Le risque de collision avec le patient pendant le déplacement autonome est traité (HN9/HN10)
G2.1
Le robot n'avance pas en cas de détection d'obstacle
G2.2
La sévérité est réduite
C9
Taux de défaillance acceptable ???
G2.1.2
La commande d'arrêt est obligatoire suite à une détection d'obstacle
G2.1.1
Le système de détection d'obstacle est efficace
Sn5
Ergonomie empêchant un choc violent
Sn45
Tests démontrant que l'arrêt temporaire n'induit pas de dangers
Sn1
Méthode de développement rigoureux (exigence SIL 61508)
G2.1.1.1
Absence de faute de conception (HW/SW)
G2.1.1.2
Le taux de défaillance est acceptable
G2.1.1.3
Les imprécisions de la détection est acceptable
Sn2
Méthode de développement rigoureux (exigence SIL 61508)
Sn3
Evaluation des taux de défaillance
Sn4
Test faux positif/faux négatif

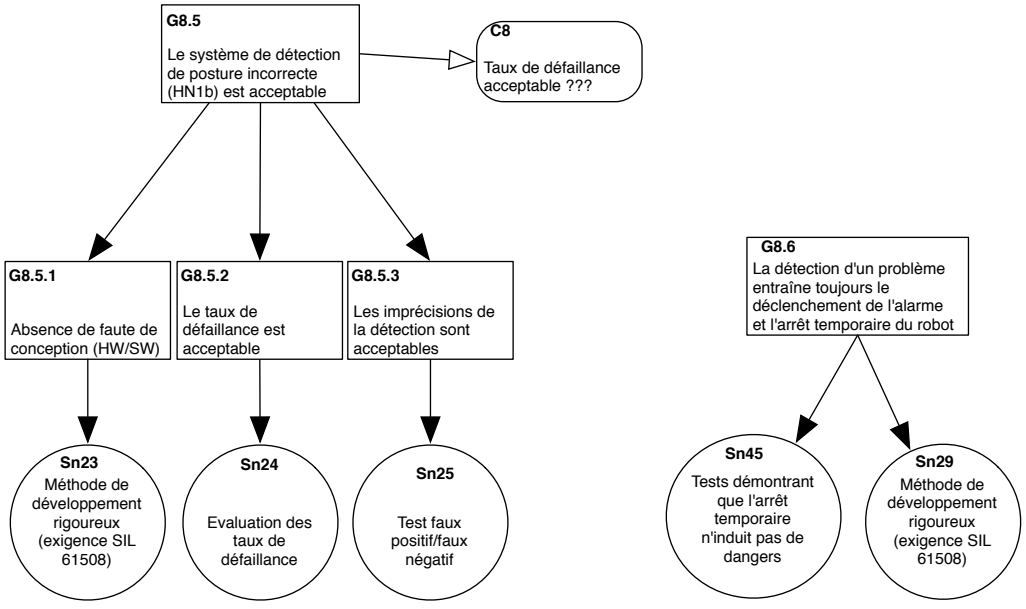

G7
Le risque de coincement d'un membre du patient est traité (HN8)
G3
Le robot n'a pas de mouvement susceptible de stresser le patient
G7.1
L'ergonomie du robot empêche le patient d'avoir un membre coincé
G7.2
Les poignées ne descendent pas plus bas que nécessaire à la fin de la déverticalisation
G3.1
L'utilisation est intuitive comme avec un déambulateur classique
G5
Tous les risques machines sont traités
Sn14
Conception mécanique du robot
G7.2.1
Absence de faute de conception (HW/SW)
Sn31
Test et avis des patients / Conception du robot
Sn32
Application de le norme Rollators ISO 11999-2
Sn10
Application de la Directive Machines 98/37/CE
Sn30
Méthode de développement rigoureux (exigence SIL 61508)

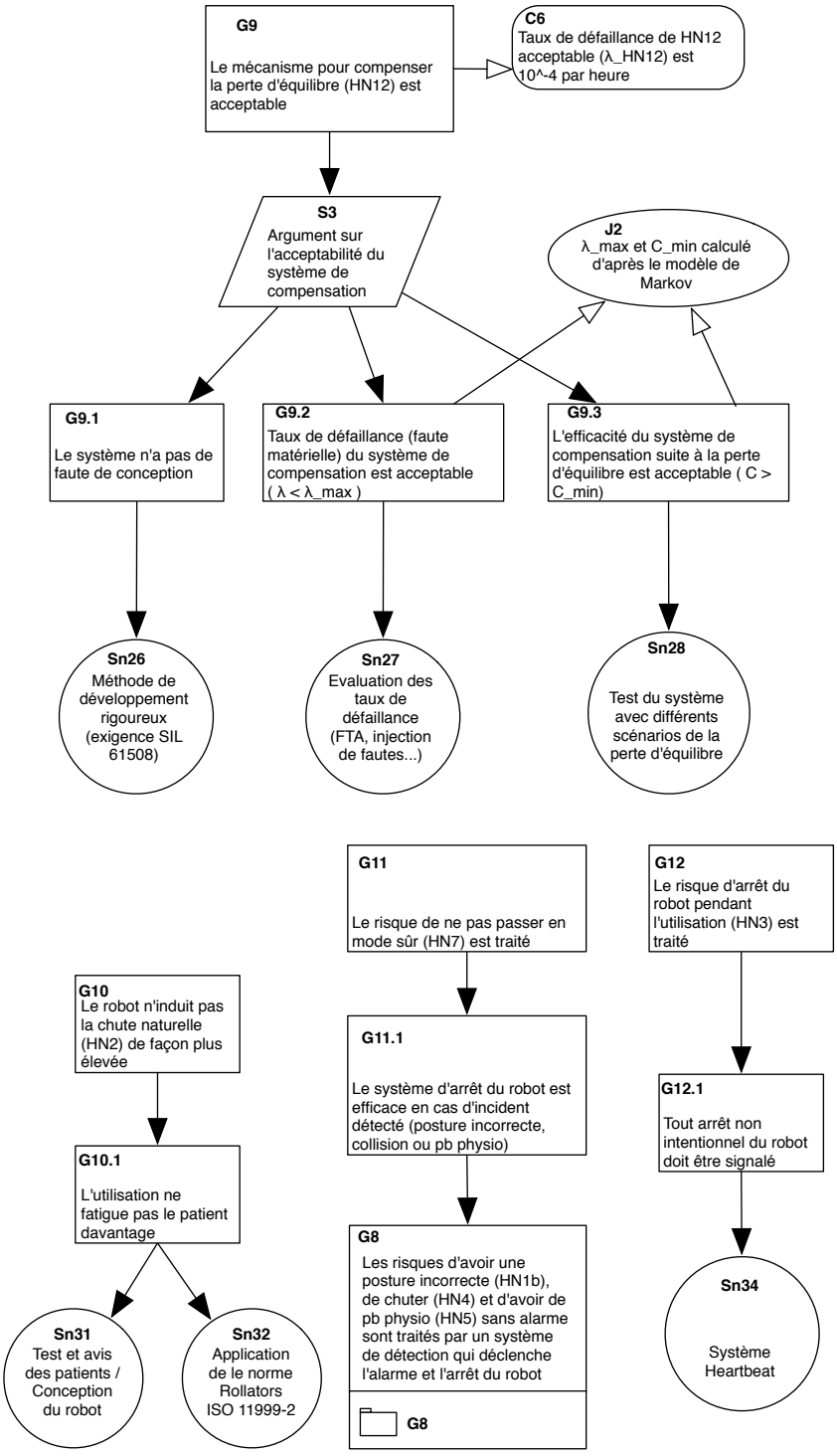

G6
Le risque de déambulation avec mauvaise posture (HN1a) due au robot est traité
G6.1
Le robot est doté d'un système de détection de mauvaise posture
G6.2
La détection de mauvaise posture implique toujours l'alarme et l'arrêt temporaire du robot
G6.1.1
Absence de faute de conception (HW/SW)
G6.1.2
Le taux de défaillance est acceptable
G6.1.3
Les imprécisions de la détection est acceptable
Sn38
Méthode de développement rigoureux (exigence SIL 61508)
Sn11
Méthode de développement rigoureux (exigence SIL 61508)
Sn12
Evaluation des taux de défaillance (FTA, injection de fautes...)
Sn13
Test faux positif/faux négatif

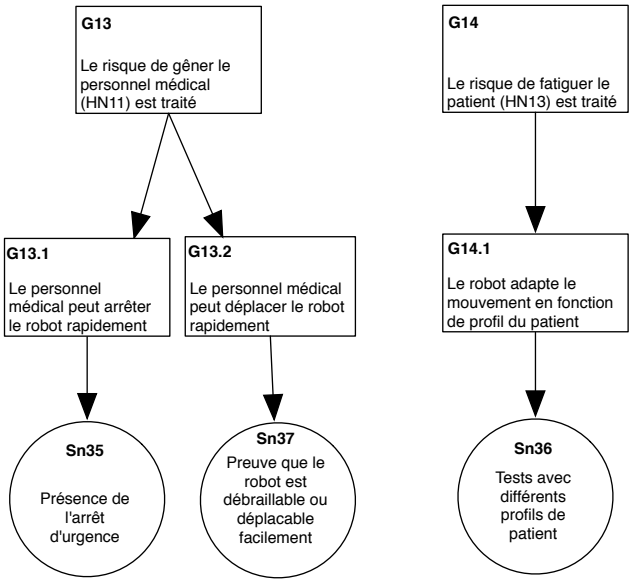

G8
Les risques d'avoir une posture incorrecte (HN1b), de chuter (HN4) et d'avoir de pb physio (HN5) sans alarme sont traités par un système de détection qui déclenche l'alarme et l'arrêt du robot
G8.1
Le système d'arrêt du robot est fiable et le mécanisme d'arrêt ne génère pas de nouveau danger
G8.6
La détection d'un problème entraîne toujours le déclenchement de l'alarme et l'arrêt temporaire du robot
G8.2
Les systèmes d'alarme sont fiables
G8.3
Le système de détection de chute (HN4) est acceptable
G8.4
Le système de détection de pb physio (HN5) est acceptable
G8.5
Le système de détection de posture incorrecte (HN1b) est acceptable

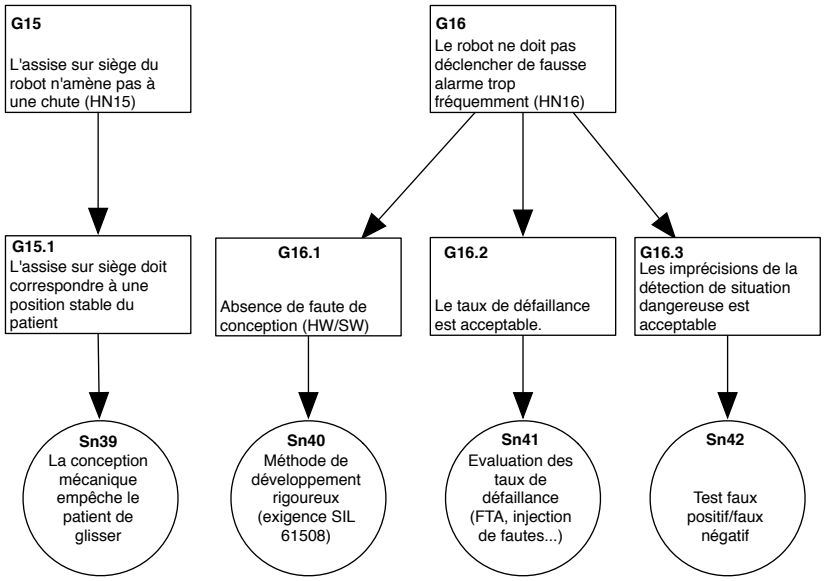

G8.1
Le système d'arrêt du robot est fiable et le mécanisme d'arrêt ne génère pas de nouveau danger
G8.2
Les systèmes d'alarme sont fiables
Sn33
Test d'arrêt dans les différents scénarios
Sn15
Conforme aux normes des machines
Sn16
Conforme aux normes des alarmes
G8.3
Le système de détection de chute (HN4) est acceptable
C6
Taux de défaillance acceptable ???
G8.3.1
Absence de faute de conception (HW/SW)
G8.3.2
Le taux de défaillance est acceptable
G8.3.3
Les imprécisions de la détection est acceptable
Sn17
Méthode de développement rigoureux (exigence SIL 61508)
Sn18
Evaluation des taux de défaillance
Sn19
Test faux positif/faux négatif


G8.4
Le système de détection de pb physio (HN5) est acceptable
C7
Taux de défaillance acceptable ???
G8.4.1
Absence de faute de conception (HW/SW)
G8.4.2
Le taux de défaillance est acceptable
G8.4.3
Les imprécisions de la détection sont acceptables
Sn20
Méthode de développement rigoureux (exigence SIL 61508)
Sn21
Evaluation des taux de défaillance
Sn22
Test faux positif/faux négatif


G8.5
Le système de détection de posture incorrecte (HN1b) est acceptable
C8
Taux de défaillance acceptable ???
G8.5.1
Absence de faute de conception (HW/SW)
G8.5.2
Le taux de défaillance est acceptable
G8.5.3
Les imprécisions de la détection sont acceptables
Sn23
Méthode de développement rigoureux (exigence SIL 61508)
Sn24
Evaluation des taux de défaillance
Sn25
Test faux positif/faux négatif
G8.6
La détection d'un problème entraîne toujours le déclenchement de l'alarme et l'arrêt temporaire du robot
Sn45
Tests démontrant que l'arrêt temporaire n'induit pas de dangers
Sn29
Méthode de développement rigoureux (exigence SIL 61508)


G9
Le mécanisme pour compenser la perte d'équilibre (HN12) est acceptable
C6
Taux de défaillance de HN12 acceptable (λ_HN12) est 10^-4 par heure
S3
Argument sur l'acceptabilité du système de compensation
J2
λ_max et C_min calculé d'après le modèle de Markov
G9.1
Le système n'a pas de faute de conception
G9.2
Taux de défaillance (faute matérielle) du système de compensation est acceptable (λ < λ_max)
G9.3
L'efficacité du système de compensation suite à la perte d'équilibre est acceptable (C > C_min)
Sn26
Méthode de développement rigoureux (exigence SIL 61508)
Sn27
Evaluation des taux de défaillance (FTA, injection de fautes...)
Sn28
Test du système avec différents scénarios de la perte d'équilibre
G11
Le risque de ne pas passer en mode sûr (HN7) est traité
G12
Le risque d'arrêt du robot pendant l'utilisation (HN3) est traité
G10
Le robot n'induit pas la chute naturelle (HN2) de façon plus élevée
G11.1
Le système d'arrêt du robot est efficace en cas d'incident détecté (posture incorrecte, collision ou pb physio)
G12.1
Tout arrêt non intentionnel du robot doit être signalé
G10.1
L'utilisation ne fatigue pas le patient davantage
G8
Les risques d'avoir une posture incorrecte (HN1b), de chuter (HN4) et d'avoir de pb physio (HN5) sans alarme sont traités par un système de détection qui déclenche l'alarme et l'arrêt du robot
G8
Sn34
Système Heartbeat
Sn31
Test et avis des patients / Conception du robot
Sn32
Application de le norme Rollators ISO 11999-2


G13
Le risque de gêner le personnel médical (HN11) est traité
G14
Le risque de fatiguer le patient (HN13) est traité
G13.1
Le personnel médical peut arrêter le robot rapidement
G13.2
Le personnel médical peut déplacer le robot rapidement
G14.1
Le robot adapte le mouvement en fonction de profil du patient
Sn35
Présence de l'arrêt d'urgence
Sn37
Preuve que le robot est débraillable ou déplacable facilement
Sn36
Tests avec différents profils de patient


G15
L'assise sur siège du robot n'amène pas à une chute (HN15)
G16
Le robot ne doit pas déclencher de fausse alarme trop fréquemment (HN16)
G15.1
L'assise sur siège doit correspondre à une position stable du patient
G16.1
Absence de faute de conception (HW/SW)
G16.2
Le taux de défaillance est acceptable.
G16.3
Les imprécisions de la détection de situation dangereuse est acceptable
Sn39
La conception mécanique empêche le patient de glisser
Sn40
Méthode de développement rigoureux (exigence SIL 61508)
Sn41
Evaluation des taux de défaillance (FTA, injection de fautes...)
Sn42
Test faux positif/faux négatif

Bibliographie

2006/42/EC (2006). Directive 2006/42/EC on machinery. Official Journal of the European Union L157.

93/42/EEC (1993). Council directive of the 14th of june 1993 concerning medical devices. Journal Officiel des Communautés Européennes (JOCE) L169.

Aguirre, F., Sallak, M., et Schon, W. (2013). Incertitudes aléatoires et épistémiques, comment les distinguer et les manipuler dans les études de fiabilité ? Dans *QUALITA2013*. Compiègne, France.

Albu-Schaffer, A., Eiberger, O., Grebenstein, M., Haddadin, S., Ott, C., Wimbock, T., Wolf, S., et Hirzinger, G. (2008). Soft robotics. *Robotics Automation Magazine, IEEE*, 15(3), p. 20–30.

Allenby, K. et Kelly, T. (2001). Deriving safety requirements using scenarios. Dans *Requirements Engineering, 2001. Proceedings. Fifth IEEE International Symposium on*, p. 228–235.

Anaheed, A., BaekGyu, K., Insup, L., et Oleg, S. (2012). A systematic approach to justifying sufficient confidence in software safety arguments. Dans *Computer Safety, Reliability, and Security Lecture Notes in Computer Science*, edité par O. Frank et D. Peter, tome 7612, p. 305–316. Springer Berlin Heidelberg.

Anaheed, A., Jian, C., Oleg, S., et Insup, L. (2013). Assessing the overall sufficiency of safety arguments. Dans *21st Safety-critical Systems Symposium (SSS'13), Bristol, United Kingdom*.

Arnold, J. et Alexander, R. (2013). Testing autonomous robot control software using procedural content generation. Dans *Computer Safety, Reliability, and Security*, edité par F. Bitsch, J. Guiochet, et M. Kaâniche, tome 8153 de *Lecture Notes in Computer Science*, p. 33–44. Springer Berlin Heidelberg.

Aven, T. (2010). Some reflections on uncertainty analysis and management. *Reliability Engineering & System Safety*, 95(3), p. 195 – 201.

Avižienis, A., Laprie, J., Randell, B., et Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 1(1), p. 11–33.

Bader, K., Lussier, B., et Schön, W. (2014). A fault tolerant architecture for data fusion targeting hardware and software faults. Dans *The 20th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC 2014)*, p. 10. Singapore.

Bensalem, S., Silva, L. d., Ingrand, F., et Yan, R. (2011). A verifiable and correct-by-construction controller for robot functional levels. *Journal of Software Engineering for Robotics*, 1(2).

Bishop, P., Bloomfield, R., et Guerra, S. (2004). The future of goal-based assurance cases. Dans *DSN Workshop on Assurance Cases : Best Practices, Possibles Obstacles, and Future Opportunities*. Florence, Italy.

Bjørn Axel Gran, R. F. et Thunem, A. P.-J. (2004). An approach for model-based risk assessment. Dans *23rd International Conference, SAFECOMP 2004, Potsdam, Germany*, p. 311–324. Springer Berlin / Heidelberg.

Blanquart, J.-P. (2010). *Survey of state of the art and of the practice in safety and diagnosability*. Rapport technique D_SP1_R5.8_M2, EADS Astrium Satellites, CESAR European Project.

Bloomfield, R. et Littlewood, B. (2003). Multi-legged arguments : the impact of diversity upon confidence in dependability arguments. Dans *Proceedings of International Conference on Dependable Systems and Networks (DSN'03)*. San Francisco, USA.

Caquas, J., Guiochet, J., Do Hoang, Q. A., et Pasqui, V. (2012). *MIRAS : Livrable L2.5. Plan de sécurité pour l'évaluation clinique d'un déambulateur robotisé*. Rapport de Contrat LAAS-CNRS 12364, LAAS.

CORAS (2014). A platform for risk analysis of security critical systems. `http://coras.sourceforge.net`.

Cyra, L. et Górski, J. (2011). Support for argument structures review and assessment. *Reliability Engineering and System Safety*, 96(1), p. 26–37.

Dardenne, A., Fickas, S., et van Lamsweerde, A. (1991). Goal-directed concept acquisition in requirements elicitation. Dans *Proceedings of 6th International Workshop on Software Specification and Design*, p. 14–21. Corno, Italy : Springer.

Dardenne, A., Fickas, S., et van Lamsweerde, A. (1993). Goal-directed requirements acquisition. Dans *Science of Computer Programming*, tome 20, p. 3–50.

DefStan 00-56 (2004). Defence standard 00-56 issue 3 : Safety management requirements for defence systems. UK Ministry of Defence.

Denney, E., Habli, I., et Pai, G. (2011). Towards measurements of confidence in safety cases. Dans *Proceedings of the 5th International Symposium on Empirical Software Engineering and Measurement (ESEM'11)*. Banff, Canada.

Díez, F. J. et Druzdzel, M. J. (2007). *Canonical probabilistic models for knowledge engineering*. Rapport technique, Research Center on Intelligent Decision-Support Systems. UNED. Madrid, Spain.

Do Hoang, Q. A. et Guiochet, J. (2012). *MIRAS : Livrable L2.4. Estimation des risques résiduels*. Rapport de Contrat LAAS-CNRS 12065, LAAS.

Do Hoang, Q. A., Guiochet, J., Kaaniche, M., et Powell, D. (2013). *Utilisation des réseaux bayésiens et de l'approche de Fenton pour l'estimation de probabilité d'occurrence d'événements*. Rapport technique, LAAS-CNRS.

Do Hoang, Q. A., Guiochet, J., Powell, D., et Kaâniche, M. (2012). Human-robot interactions : model-based risk analysis and safety case construction. Dans *Embedded Real Time Software and Systems (ERTS2 2012)*. Toulouse, France.

Durand, B. (2011). *Proposition d'une architecture de contrôle adaptative pour la tolérance aux fautes*. Thèse de doctorat, Université de Montpellier 2.

Fenton, N. et Neil, M. (2012). *Risk Assessment and Decision Analysis with Bayesian Networks*. CRC Press, Taylor and francis Group.

Fenton, N., Neil, M., et Caballero, J. G. (2007). Using ranked nodes to model qualitative judgments in bayesian networks. *IEEE Transactions onKnowledge and Data Engineering*, 19(10), p. 1420 –1432.

Filippini, R., Sen, S., et Bicchi, A. (2008). Toward soft robots you can depend on. *Robotics Automation Magazine, IEEE*, 15(3), p. 31–41.

Gacogne, L. (2003). *Logique floue et applications*. Conservatoire National des arts et métiers, Institut d'informatique d'entreprise d'Evry.

Glauser, D., Flury, P., Burckhardt, C., et Kassler, M. (1993). Mechanical concept of the neurosurgical robot Minerva. *Robotica*, 11(6), p. 567–575.

Goodenough, J. B., Weinstock, C. B., et Klein, A. Z. (2013). Eliminative induction : a basis for arguing system confidence. Dans *ICSE'13*, p. 1161–1164.

Gorski, J. et Jarzebowicz, A. (2005). Development and validation of a HAZOP-based inspection of UML models,. Dans *3rd World Congress for Software Quality, Munich, Germany*.

GSN-Standard (2011). GSN COMMUNITY STANDARD VERSION 1. `http://www.goalstructuringnotation.info` [Online ; accessed Decembre 18th 2014].

Guiochet, J., Do Hoang, Q. A., Kaaniche, M., et Powell, D. (2013). Model-based safety analysis of human-robot interactions : The MIRAS walking assistance robot. Dans *Rehabilitation Robotics (ICORR), 2013 IEEE International Conference on*, p. 1–7.

Guiochet, J., Martin-Guillerez, D., et Powell, D. (2010). Experience with model-based user-centered risk assessment for service robots. Dans *IEEE International Symposium on High-Assurance Systems Engineering (HASE'2010)*, p. 104–113. San Jose, CA, USA : IEEE Computer Society.

Guiochet, J., Motet, G., Baron, C., et Boy, G. (2004). Toward a human-centered UML for risk analysis - application to a medical robot. Dans *Proc. of the 18th IFIP World Computer Congress (WCC), Human Error, Safety and Systems Development (HESSD04)*, edité par C. Johnson et P. Palanque, p. 177–191. Kluwer Academic Publisher.

Haddadin, S. (2014). *Towards Safe Robots, Approaching Asimov's 1st Law*, tome Springer Tracts in Advanced Robotics, Vol. 90. Springer.

Haddadin, S., Suppa, M., Fuchs, S., Bodenmüller, T., Albu-Schäffer, A., et Hirzinger, G. (2011). Towards the robotic co-worker. Dans *Robotics Research*, edité par C. Pradalier, R. Siegwart, et G. Hirzinger, tome 70 de *Springer Tracts in Advanced Robotics*, p. 261–282. Springer Berlin Heidelberg. `http://dx.doi.org/10.1007/978-3-642-19457-3_16`.

Hansen, K. M., Wells, L., et Maier, T. (2004). HAZOP analysis of UML-based software architecture descriptions of safety-critical systems. Dans *Proceedings of NWUML*.

Harel, D. (1987). Statecharts : A visual formalism for complex systems. *Sci. Comput. Program.*, 8(3), p. 231–274.

Hawkins, R., Kelly, T., Knight, J., et Graydon, P. (2011). A new approach to creating clear safety arguments. Dans *Proceedings of 19th Safety Critical Systems Symposium*. Southampton, UK.

Hitchcock, D. (2005). Good reasoning on the toulmin model. *Argumentation*, 19(3), p. 373–391.

IEC61508 (2010). Functional safety of electrical/electronic/programmable electronic safety-related systems. Édition 2. International Electrotechnical Commission.

IEC61508-5 (2010). Functional safety of electrical/electronic/programmable electronic safety-related systems : Part 5 : Examples of methods for the determination of safety integrity level. International Electrotechnical Commission.

IEC61882 (2001). Hazard and operability studies (HAZOP studies) – Application guide. International Electrotechnical Commission.

IEC62304 (2006). Medical device software - software life cycle processes. International Electrotechnical Commission.

ISO10218-1 (2011). Robots for industrial environments – safety requirements – part 1 : Robot. International Organization for Standardization.

ISO12100 (2010). Safety of machinery - general principles for design - risk assessment and risk reduction. International Standard Organisation.

ISO13482 (2014). Robots and robotic devices – safety requirements for personal care robots. International Organization for Standardization.

ISO13849-1 (2006). Safety of machinery – safety-related parts of control systems – part 1 : General principles for design. International Organization for Standardization.

ISO/DIS31000 (2009). Risk management — principles and guidelines on implementation. International Standard Organisation.

ISO/FDIS14971 (2006). Medical devices - Application of risk management to medical devices. International Standard Organisation.

ISO/IEC-Guide73 (2009). Risk management - Vocabulary - Guidelines for use in standards. International Organization for Standardization.

Jarzebowicz, A. et Górski, J. (2006). Empirical evaluation of reading techniques for UML models inspection. *International Transactions on Systems Science and Applications*, 1(2), p. 103–110.

Johannessen, P., Grante, C., Alminger, A., Eklund, U., et Torin, J. (2001). Hazard analysis in object oriented design of dependable systems. Dans *2001 International Conference on Dependable Systems and Networks, Göteborg, Sweden*, p. 507–512.

Kelly, T. et McDermid, J. (1997). Safety case construction and reuse using patterns. Dans *16th International Conference on Computer Safety and Reliability (SAFECOMP97)*.

Kelly, T. P. (1998). *Arguing Safety – A Systematic Approach to Managing Safety Cases*. Thèse de doctorat, University of York.

Lano, K., Clark, D., et Androutsopoulos, K. (2002). Safety and security analysis of object-oriented models. Dans *SAFECOMP '02 : Proceedings of the 21st International Conference on Computer Safety, Reliability and Security*, p. 82–93. London, UK : Springer-Verlag.

Laprie, J. (2004). Sûreté de fonctionnement informatique : Concepts de base et terminologie. *Revue de l'Électricité et de l'Électronique*.

Littlewood, B. et Rushby, J. (2012). Reasoning about the reliability of diverse two-channel systems in which one channel is "possibly perfect". *IEEE Transactions on software engineering*, 38(5), p. 1178–1194.

Littlewood, B. et Wright, D. (2007). The use of multilegged arguments to increase confidence in safety claims for software-based systems : A study based on a BBN analysis of an idealized example. *IEEE Trans. Software Eng.*, 33(5), p. 347–365.

Lussier, B., Gallien, M., Guiochet, J., Ingrand, F., Killijian, M.-O., et Powell, D. (2007). Fault tolerant planning for critical robots. Dans *37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN07), Edinburgh, UK*.

Machin, M., Dufossé, F., Blanquart, J.-P., Guiochet, J., Powell, D., et Waeselynck, H. (2014). Specifying safety monitors for autonomous systems using model-checking. Dans *Computer Safety, Reliability, and Security*, edité par A. Bondavalli et F. Di Giandomenico, tome 8666 de *Lecture Notes in Computer Science*, p. 262–277. Springer International Publishing. `http://dx.doi.org/10.1007/978-3-319-10506-2_18`.

Mainprice, J., Sisbot, E. A., Siméon, T., et Alami, R. (2010). Planning safe and legible hand-over motions for human-robot interaction. *IARP Workshop on Technical Challenges for Dependable Robots in Human Environments*, 2(6), p. 7.

Martin-Guillerez, D., Guiochet, J., et Powell, D. (2010a). Experience with a model-based safety analysis process for an autonomous service robot. Dans *IARP Workshop on Technical Challenges for Dependable Robots in Human Environments*.

Martin-Guillerez, D., Guiochet, J., et Powell, D. (2010b). A UML-based method for risk analysis of human-robot interactions. Dans *2nd International Workshop on Software Engineering for Resilient Systems*. ACM.

Martin-Guillerez, D., Guiochet, J., Powell, D., et Zanon, C. (2010c). UML-based method for risk analysis of human-robot interaction. Dans *International Workshop on Software Engineering for Resilient Systems (SERENE'2010), London, UK*.

Mekki-Mokhtar, A., Blanquart, J.-P., Guiochet, J., Powell, D., et Roy, M. (2012). Safety trigger conditions for critical autonomous systems. Dans *18th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC 2012), Niigata, Japan*.

OMG (Consulté le 13 mars 2010). Unified Modeling Language Specification v2. `http://www.omg.org`, 2007.

OMG-ARM (2013). Structured assurance case metamodel (SACM), version 1. Object Management Group.

Pasqui, V., Saint Bauzel, L., Zong, C., Clady, X., Decq, P., Piette, F., Michel-Pellegrino, V., El Helou, A., Carre, M., Durand, A., Do Hoang, Q. A., Guiochet, J., Rumeau, P., Dupourque, V., et Caquas, J. (2012). Projet miras : robot d'assistance à la déambulation avec interaction multimodale. *Ingénierie et Recherche BioMédicale (IRBM)*, 33(2), p. 165–172.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems : networks of plausible inference*. Morgan Kaufmann Publishers Inc. San Francisco, USA.

PHRIENDS (2006). Physical Human-Robot Interaction : Dependability and Safety. Project supported by the European Commission under the 6th Framework Programme (STReP IST-045359),`www.phriends.eu`. Accessed : 2014-09-01.

Rey, A. (1991). *Le petit Robert 1*. Dictionnaires Le Robert.

Roger, W., Alan, W., Chris, H., et Mike, F. (2012). Building safer robots : Safety driven control. *The International Journal of Robotics Research*.

Sabetzadeh, M., D. Falessi, D., Briand, L., Alesio, S., McGeorge, D., Å hjem, V., et Borg, J. (2011). Combining goal models, expert elicitation, and probabilistic simulation for qualification of new technology. Dans *Proceedings of 13th International Symposium on High-Assurance Systems Engineering (HASE'11)*. USA.

Simon, C., Weber, P., et Evsukoff, A. (2008). Bayesian networks inference algorithm to implement dempster shafer theory in reliability analysis. *Reliability Engineering and System Safety, Elsevier*, 93(7).

Srivatanakul, T. (2005). *Security Analysis with Deviational Techniques*. Thèse de doctorat, University of York.

Stanton, N., Salmon, P., Walker, G., Baber, C., et Jenkins, D. P. (2006). *Human Factors Methods : A Practical Guide for Engineering And Design*. Ashgate Publishing.

Toulmin, S. (1958). The uses of argument. *Cambridge University Press*.

Weaver, R. et Kelly, T. (2004). The goal structuring notation - a safety argument notation. Dans *Proceedings of the Dependable Systems and Networks 2004, Workshop on Assurances Cases*.

Zhao, X., Zhang, D., Lu, M., et Zeng, F. (2012). A new approach to assessment of confidence in assurance cases. Dans *SAFECOMP Workshops*, p. 79–91.

Analyse et justification de la sécurité de systèmes robotiques en interaction physique avec l'humain

Résumé : Les systèmes s'adaptant à leur environnement et en interaction physique avec l'homme se développent de plus en plus dans des domaines comme le médical, l'assistance aux personnes ou le travail en usine. Ils diffèrent des systèmes classiques par leur capacité à s'adapter à l'environnement et à prendre des décisions en tenant compte de leur perception de l'environnement et notamment de l'homme. La défaillance de tels systèmes pouvant avoir des conséquences catastrophiques sur l'homme, l'analyse et la démonstration du niveau de confiance que l'ont peut leur accorder vis-à-vis de la sécurité-innocuité, et a fortiori leur certification, constituent aujourd'hui un vrai défi.

La construction d'argumentaire de sécurité (ou dossier de sécurité, ou *safety case*), est un des moyens permettant de préparer la certification de tels systèmes. Il s'agit principalement de justifier pour chaque danger comment il a été traité et ramené à un niveau acceptable. Malheureusement, dans le cas des systèmes robotiques, de nombreuses incertitudes subsistent, et il n'existe pas à l'heure actuelle de méthode systématique permettant la construction de tels dossiers de sécurité et la démonstration du niveau de confiance sous-jacent. L'objectif des travaux est de contribuer à la définition d'une telle méthode en partant d'une technique d'analyse du risque dédiée à l'analyse des interactions humain-robot, puis en s'appuyant sur des modèles formalisés permettant de construire l'argumentaire de sécurité et d'évaluer automatiquement le niveau de confiance dans cet argumentaire.

Mots clés : Sécurité des robots, analyse du risque, UML, argumentaire de sécurité, confiance, robotique de service

---

Safety analysis and justification of human-robot interactions

Abstract : Robotic systems that continuously adapt to their environment and physically interact with human are increasingly used in various fields like personal assistance or factory work. They are characterised by their ability to adapt to the environment, to take decision in the light of their perception of the environment and particularly of the human. As the failure of such systems may lead to catastrophic consequences, analysis and justification of the level of confidence in these systems with regards to safety, and furthermore their certification is a real challenge.

The construction of a Safety Case is one of the means that can be used to support the certification of such systems. It is aimed at describing and justifying how every hazard has been mitigated and its severity maintained as low as reasonably possible. However, for robotic systems that have to deal with many uncertainties, there is a lack of a systematic approach to support the construction of their Safety Case and the assessment of its underlying confidence. Our research aims at contributing to the development of such a systematic approach starting with a risk analysis focusing on human-robot interactions, followed by Safety Case construction from formalized models and finally an automatic assessment of the confidence in safety argumentation. As a case study, the safety of a rehabilitation robot for strolling is analysed and justified based on the approaches developed in this thesis.

Keywords : Robot safety, risk analysis, safety case, confidence, rehabilitation robot.