HAL open science

Quantification et propagation des détectabilités des systèmes MBSE

Yunhui Hou, Vincent Idasiak, Kratz Frédéric

To cite this version:

Yunhui Hou, Vincent Idasiak, Kratz Frédéric. Quantification et propagation des détectabilités des systèmes MBSE. Congrès Lambda Mu 22 " Les risques au cœur des transitions " (e-congrès) - 22e Congrès de Maîtrise des Risques et de Sûreté de Fonctionnement, Institut pour la Maîtrise des Risques, Oct 2022, Paris Saclay, France. hal-03865286

HAL Id: hal-03865286

https://hal.science/hal-03865286v1

Submitted on 22 Nov 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Quantification et propagation des détectabilités des systèmes MBSE

HOU Yunhui
INSA Centre Val de Loire – Laboratoire PRISME EA 4229
88 Boulevard Lahitolle, 18000 Bourges
yunhui.hou@insa-cvl.fr

IDASIAK Vincent
INSA Centre Val de Loire – Laboratoire PRISME EA 4229
88 Boulevard Lahitolle, 18000 Bourges
vincent.idasiak@insa-cvl.fr

KRATZ Frédéric
INSA Centre Val de Loire – Laboratoire PRISME EA 4229
88 Boulevard Lahitolle, 18000 Bourges
frederic.kratz@insa-cvl.fr

Résumé — Dans cet article, nous proposons une extension de la Base de données de Défaillances Fonctionnelles (BDF) en SysML et de la méthode Médisis permettant de créer un lien entre un mode de défaillance et les fonctions de détection et de pronostics qui peuvent signaler directement son apparition. Ensuite, un algorithme de propagation de détectabilité est proposé en utilisant la BDF et cette extension. Finalement, un exemple illustrant l'application de notre algorithme est proposé.

Mots-clefs — AMDEC, détectabilité, MSBE, sûreté de fonctionnement, propagation fonctionnelle.

I. INTRODUCTION

Dans la méthode AMDEC, la criticité est obtenue en tenant compte d'un triplet de facteurs : gravité, occurrence et détectabilité. La détectabilité désigne la capacité de détection des défaillances, de contrôle et de surveillance du système. Elle est souvent évaluée de manière qualitative (détection certaine, détection par opérateur, difficilement détectable, indétectable) et est traitée comme une caractéristique du mode de défaillance sans tenir compte de la dépendance avec d'autres fonctions du système (détection, surveillance, pronostics) et de leur fiabilité [6]. De plus, les défaillances de détection et de prédiction font toujours partie des causes des événements indésirables ou des augmentations de coût d'utilisation du système [4]. Par exemple, pour qu'un système de surveillance ou de détection automatique fonctionne correctement, l'observabilité et la contrôlabilité du système doivent être absolument assurées. La disponibilité des fonctions concernant la récupération des mesures (capteurs ou détecteurs) doit aussi être prise en compte dans la propagation de détectabilité [5]. Pour une défaillance détectée par opérateur, la procédure et la périodicité de l'opération et plus loin la fiabilité du facteur humain jouent aussi un rôle important dans la propagation de détectabilité. Il y a donc une nécessité de quantifier et d'analyser la propagation fonctionnelle de la détectabilité.

L'objectif de cet article est d'étudier la quantification et la propagation de détectabilité et son lien avec les modes de défaillance et les mesures de sûreté de fonctionnement des fonctions concernées.

Dans notre article, nous allons présenter une méthode pour analyser les propriétés structurelles et la détectabilité des systèmes complexes dont chaque mode de défaillance à étudier sera modélisé en utilisant l'approche MBSE. La dépendance entre la détectabilité et les fonctions qui participent à la réalisation de la détection de la défaillance est identifiée par une extraction des informations de mode de défaillance et ses causes/effets à partir d'un métamodèle d'une base de données de défaillances fonctionnelles (BDF) en SysML de la méthode Médisis avec une extension adaptée [2,3].

Dans section III, une méthode de détermination et de propagation de détectabilité est présentée.

Un exemple sur un système surveillé de manière automatique est présenté dans Section IV.

II. MODELISATION

A. *Modélisation des défaillances fonctionnelles*

Notre approche est une extension de la méthodologie MéDISIS (Méthodologie D'intégration des analyses de Sûreté de fonctionnement à l'Ingénierie Système) [2,3]. MéDISIS propose une méthode de traitement de l'information et des connaissances pour les systèmes complexes et multi-technologiques. La méthode est basée sur deux métamodèles/bases de données : la Base de données des Dysfonctionnements des Fonctions (BDF) et la Base de Comportements Dysfonctionnels (BCD) alimentées automatiquement par le modèle de système en SysML avec correction manuelle par des experts.

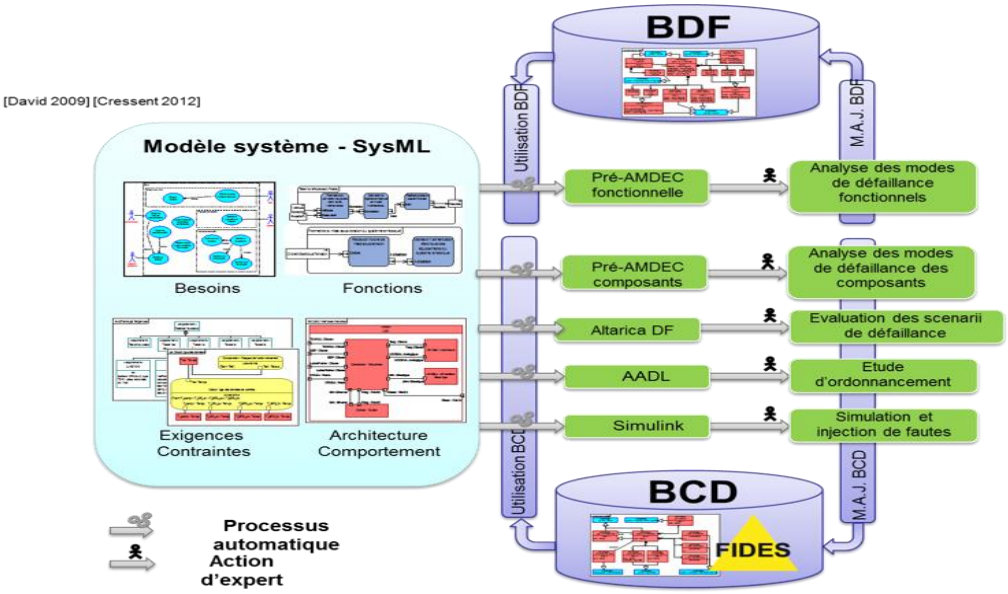

Figure 1 Modèle Médisis [2,3]

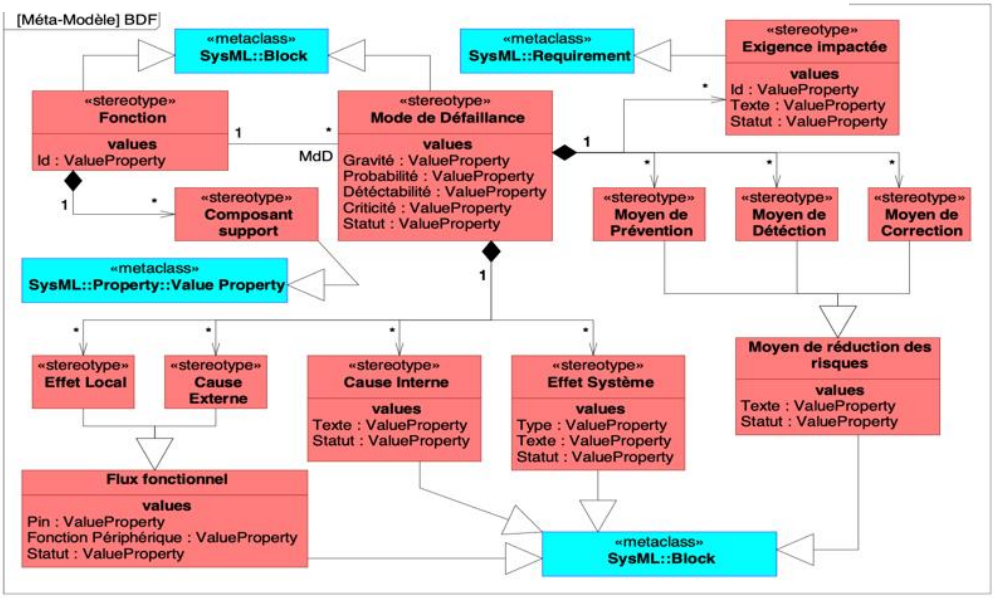

Figure 2 Méta-modèle BDF [1,2,3]

La BCD est définie pour pérenniser les données dysfonctionnelles des composants, réciproquement la BDF stocke les données dysfonctionnelles concernant les fonctions [1].

B. Modélisation des outils de détection et de pronostics dans un système

Du point de vue des composants, les outils de détection nous permettent de connaitre l'existence d'une défaillance ou d'une anomalie par surveillance sur une ou plusieurs mesures ou un phénomène. Ils sont souvent réalisés par l'utilisation de programmes et/ou d'algorithmes de diagnostics utilisant en général les données provenant des capteurs.

La maintenance et le pronostic sont associés à un autre groupe d'outils. Au contraire de celui des outils de détection, les outils de pronostic utilisent des mesures/informations concernant des signes avant-coureurs de défaillance du système ainsi que les données historiques ou le modèle du système. La qualité de connaissance est un facteur décisif pour la précision d'un outil de pronostic. L'autre différence entre les deux groupes d'outils est que les modes de défaillance de type de dégradation (légères) peuvent être signalées par les outils pronostics.

Pour un outil de détection et de pronostics, la signalisation des défaillances est sur un composant ou un ensemble de composants. Il faut ensuite localiser et isoler les causes de défaillance, mais cela n'est pas toujours possible, spécifiquement pour des outils de pronostics basés sur l'apprentissage automatique.

Du point de vue fonctionnelle, les fonctions de détection et de pronostics fournissent des indicateurs de défaut et de maintenance à la supervision. Ces indicateurs représentent les signes de dégradation et/ou de perte d'une ou plusieurs fonctions. Les outils de détection et de pronostic sont souvent mentionnés dans l'aspect sécurité ou dans l'aspect sûreté de fonctionnement à travers les exigences. Dans le cadre de BDF, les fonctions de détection et de pronostics sont vues comme un moyen de détection : une fonction de détection signale la détection d'une défaillance qui existe déjà ; une fonction de pronostics prévient/prédit l'arrivée d'une défaillance à partir d'un instant donné.

Comme toutes les fonctions, les fonctions de détections et de pronostics peuvent aussi tomber en panne. Si la défaillance de ces fonctions n'est pas signalée et qu'elle n'entraine pas de procédure de sécurité à court terme, il n'y aura alors aucun impact direct sur la fiabilité et la disponibilité de la fonction concernant les exigences principales surveillées. Au contraire, la détectabilité de fonctions surveillées dépend fortement des fonctions de détections et de pronostics.

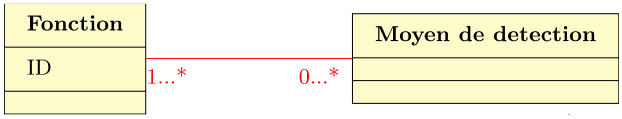

Figure 3 Extension de BDF : Détection

En général, l'approche proposée est de synthétiser un lien entre la ou les fonctions de détection ou de prévention (pronostics) et le ou les modes de défaillance qu'elles surveillent. Cette association signifie que la fonction référencée est le moyen de détection ou de pronostics d'un ou plusieurs modes de défaillance dont la détectabilité est étudiée.

III. PROPAGATION FONCTIONNELLE DE DETECTABILITE

Dans cette section, nous proposons une méthode de propagation à des fins de vérification/validation de la détectabilité ainsi que de calculer la détectabilité en fonction des mesures de sûreté de fonctionnement des fonctions de détection et de pronostic.

A. Étape 1 : Identification des fonctions concernant la détectabilité

Dans cette étape, les fonctions de détections et de prévision des défaillances sont identifiées. Parmi ces fonctions nous pouvons trouver :

- Une Fonction de type détection : alarme, test périodique, surveillance par opérateur ou par outil diagnostique

- Une Fonction de type prévision : programme de pronostics

B. Étape 2 : Création des connexions entre la rubrique de la détectabilité d'un mode de défaillance et la fonction qui représente le moyen de détection.

Dans cette étape, la connexion entre une fonction de détection ou de pronostics et le ou les modes de défaillance signalés par ces fonctions est définie par une association : « Détecté Par »/ « Détecté » (ou « DetectedBy »/ « Detect »). Pour chaque fonction de détection identifiée, on recherche les effets des défaillances possibles signalées directement en étudiant le texte de la description. Un des raccourcis est alors d'analyser les paramètres et les mesures capturées par la fonction de détection et de remonter jusqu'aux effets des défaillances impactant les mesures concernées. Cette méthode se base sur la propagation d'un flux fonctionnel de « detection/pronostic » de la classe « fonction » vers le package « Paramètre » et ceci jusqu'aux classes « effet local » et « effet système » d'un mode de défaillance.

Pour chaque fonction de détection et de pronostics, (noté *d*), nous sommes en mesure de trouver les modes de défaillance (notée m) d'une fonction (notée f) pouvant être détectées par *d*. Ces modes de défaillance d'une fonction sont notée *f.m.* Dès lors, nous définissons les opérations Detect() et DetectedBy() telles que

$$d \in \mathrm{DetectedBy}(f.m)$$

$$f.m \in \mathrm{Detect}(d)$$

Pour chaque couple, mode de défaillance $f.m$, on cherche les effets possibles (défaillance possible) sur les autres fonctions. Ces effets possibles sont représentés par des modes de défaillance des fonctions impactées $f_p.m'$ via le bloc « system effects » désigné par les experts selon des « use cases » et des cas spécifiques dans la BDF, et ainsi que ceux des fonctions périphériques impactées possibles

Cette relation est présentée par deux opérations Effect() et EffectedBy() telles que

$$f_p.m' \in \mathrm{Effect}(f.m)$$

$$f.m \in \mathrm{EffectedBy}(f_p.m')$$

Pour chaque mode de défaillance d'une fonction $f.m$, il est nécessaire de rechercher les coupes minimales possibles $c_1, \dots, c_n$ par les opérations Cause() et CausedBy() suivantes, via le bloc « Cause Interne » et « Cause Externe » définies telles que

$$c_1, \dots, c_n \in \mathrm{Cause}(f.m)$$

$$f.m \in \mathrm{CausedBy}(c_i)$$

Les coupes sont formées d'un ensemble de mode de défaillance des fonctions dont les défaillances satisfassent la condition du mode de défaillance étudié, *i.e.*

$$c_i = (f_{c1}.m_{i_1}, \dots, f_{c_n}.m_{i_n})$$

Fonction Detection $(f.m)$
Initialisation :
Pour chaque mode de défaillance m tel que DetectedBy$(f.m)$ n'est pas vide

$$f.m.\text{détectabilité} = 1 - \sum_{d \in \text{DetectedBy(f.m)}} (1 - d.\text{détectabilité})$$

Tant Qu'il existe des modes de défaillance avec détectabilité non attribuée.
Pour toutes les fonctions $f.m$ dont la détectabilité n'est pas attribuée
    $D1 = 0$
    Pour tous les causes $c \in Cause(f.m)$
        $D_c = 1$
        Pour chaque $e \in c$
            Si $e.$détectabilité est attribuée
                $D_c = D_c * (1 - e.\text{détectabilité})$
            FinSi
        FinPour
        $D_c = 1 - D_c$
        $D1 = D1 + D_c * \prod_{m_i \in c} m_i.frequence$
    FinPour
    $D1 = \frac{D1}{f.m.frequence}$
    $D2 = 0$
    Pour chaque effet $e \in Effect(f.m)$
        Si $e.$détectabilité est attribuée
            $D21 = 0$
            Pour chaque cause $c \in Cause(e)$
                Si $f.m \in c$
                $D21 = D21 + \prod_{m_i \in c, m_i \neq f.m} m_i.frequence$
                FinSi
            FinPour
            $D2 = D2 + e.\text{détectabilité} * D21$
        FinSi
    FinPour
    $f.m.\text{détectabilité} = 1 - (1 - D1)(1 - D2)$
FinPour
Fin TantQue
Fin

Figure 4 Algorithme : propagation de détectabilité

Les résultats des opérations précédentes correspondent à un élément unique ou bien à un ensemble d'éléments.

C. Étape 3 : Évaluation de disponibilité de fonction de détection et de pronostic de défaillance

Pour une fonction détection et de pronostic de défaillance, notée d, on cherche à attribuer une disponibilité de ses fonctions. La valeur de disponibilité est égale à 1 moins la somme des fréquences des modes de défaillance de cette fonction.

$$d.\text{disponibilité} = 1 - \sum_{\forall m = d.m} m.\text{fréquence}$$

Étape 4 : Procédure de propagation : Rechercher le moyen de détection dont dépend la fonction étudiée.

Dans cette section nous allons présenter l'algorithme nous permettant de définir la détectabilité d'une fonction quelconque.

Pour un mode de défaillance m d'une fonction quelconque f, la fonction de Detection() proposée (explicitée dans la Figure 4) propage la détectabilité par attribution des valeurs de détectabilité aux modes de défaillance touchés.

Dans notre algorithme proposé, si le mode de défaillance peut être signalé par une ou plusieurs fonctions de détection ou de pronostics, sa détectabilité est donnée directement par la disponibilité de ces fonctions, en supposant que ces fonctions de détections et de pronostics fonctionnent indépendamment, et forment une structure parallèle.

Si la défaillance ne peut pas être détectée ou prédite directement, l'algorithme cherche les effets et les causes de cette défaillance.

Pour les causes, si une des coupes minimales possède une cause (qui est aussi un mode de défaillance) avec détectabilité attribuée, alors la défaillance étudiée sera détectée si cette cause fait partie réellement des causes de la défaillance.

Pour les effets, si un des effets (qui est aussi un mode de défaillance) possède une détectabilité attribuée, alors sa détectabilité est égale à la probabilité que cet effet soit détecté correctement et que la défaillance étudiée fasse partie de ces causes en sachant que la défaillance existe.

Il est possible qu'une défaillance n'aie pas de fonction de détection automatique implémentée dans le système ou assuré par l'opérateur. Dans ce cas-là, l'algorithme donne une détectabilité de valeur zéro. Ensuite, il est possible de proposer une méthode d'amélioration ou de demander à un expert de proposer une valeur sur la détectabilité.

IV. Exemple d'illustration

A. *Présentation de l'exemple*

Dans cette section, nous illustrons la méthode présentée précédemment par un exemple dont le flux fonctionnel est présenté dans Figure 5.

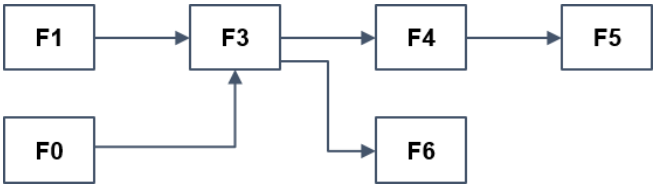

Figure 5 Exemple d'illustration : flux fonctionnel [1]

D'après le flux fonctionnel, les coupes des fonctions étudiées sont présentées sous la forme de résultats des opérations Cause() et Effect() définies ci-dessous :

$\text{Cause}(F_3.m_A) = \{(F_1.m_A, F_0.m_A)\}$

$\text{Cause}(F_3.m_B) = \{(F_1.m_B, F_0.m_B), (F_1.m_A, F_0.m_B)\}$

$\text{Cause}(F_4.m_A) = \{(F_3.m_A), (F_3.m_B)\}$

$\text{Cause}(F_5.m_B) = \{(F_4.m_A)\}$

$\text{Cause}(F_6.m_A) = \{(F_3.m_B)\}$

$\text{Effect}(F_3.m_A) = \{F_4.m_A\}$

$\text{Effect}(F_3.m_B) = \{F_4.m_A, F_6.m_A\}$

$\text{Effect}(F_4.m_A) = \{F_5.m_B\}$

$\text{Effect}(F_1.m_A) = \{F_3.m_A, F_3.m_B\}$

$\text{Effect}(F_1.m_B) = \{F_3.m_A, F_3.m_B\}$

$\text{Effect}(F_0.m_A) = \{F_3.m_A\}$

$\text{Effect}(F_0.m_B) = \{F_3.m_B\}$

Les défaillances des fonctions F_1 et F_0 sont causées par les éléments externes dont les fréquences sont connues telles que :

$$F_1.m_A.Fréquence = 0.02$$

$$F_0.m_A.Fréquence = 0.02$$

$$F_0.m_B.Fréquence = 0.02$$

La fonction F_3 est surveillée par une fonction de détection F_2 dont la disponibilité est donnée par :

$$F_2.disponibilité = 0.98$$

La fonction F_2 est capable de détecter les modes de défaillances m_A et m_B de la fonction F_3, *i.e.*

$$DetectedBy(F_3.m_A) = F_2$$

$$DetectedBy(F_3.m_B) = F_2$$

B. *Application numérique*

Dans cette section, l'algorithme de propagation de détectabilité présentée Figure 4 est appliqué sur l'exemple numérique.

Pour F_3 , les deux modes de défaillance sont détectées directement par la fonction F_2, nous obtenons donc que :

$F_3.m_A.détectabilité = 0.98$
$F_3.m_B.détectabilité = 0.98$

Lors de la première itération, les détectabilités de F_1 et F_0 sont attribuées par l'opération Effect().

Les défaillances sur F_1 et F_0 font partie des coupes de $F_3.m_A$ et $F_3.m_B$. Un élément dans une coupe peut être détecté si : une apparition de toutes les défaillances de manière simultanée de cette coupe apparait et provoque une défaillance avec détectabilité attribuée.

Par exemple, $F_1.m_A$ sera détecté quand elle provoque avec $F_0.m_A$ la défaillance $F_3.m_A$ ou quand elle provoque avec $F_0.m_B$ la défaillance $F_3.m_B$. Donc sa détectabilité est la somme de la probabilité sachant l'existence de $F_1.m_A$, $F_3.m_A$ soit provoqué par la coupe $(F_1.m_A, F_0.m_A)$, et que $F_3.m_A$ soit bien signalé, ou que $F_3.m_B$ soit provoqué par la coupe $(F_1.m_A, F_0.m_B)$ et que $F_3.m_B$ soit bien signalé. Donc nous obtenons

$F_1.m_A.détectabilité = 0.0392$
$F_1.m_B.détectabilité = 0.0392$

$F_0.m_A.détectabilité = 0.0196$
$F_0.m_B.détectabilité = 0.0196$

Les détectabilités de F_4 et F_6 sont attribuées par l'opération Cause() car les causes de défaillances de ces fonctions sont toutes sur F_3. La détectabilité d'une défaillance ayant une des causes détectables est déterminée par la détectabilité de cette cause et la probabilité que cette cause fasse partie réellement de la coupe qui a causé cette défaillance en sachant la cause détectable et que la défaillance étudiée existe. Par exemple, $F_4.m_A$ est causé par $(F_3.m_A), (F_3.m_B)$. Sachant l'existence $F_4.m_A$, la probabilité que cette défaillance ait été causée par $(F_3.m_A)$ ou $(F_3.m_B)$ est respectivement égale 0.5.

$$F_4.m_A.détectabilité = 0.5 * F_3.m_A.detectabilié + 0.5 * F_3.m_B.detectabilié$$

Donc nous obtenons

$F_4.m_A.détectabilité = 0.98$
$F_6.m_A.détectabilité = 0.98$

La détectabilité de $F_5.m_B$ est attribuée dans la deuxième itération où la détectabilité de la seule cause $F_4.m_A$ est déjà attribuée. Alors on obtient la détectabilité suivante :

$F_5.m_B.détectabilité = 0.98$

Cet exemple numérique à permis de présenter l'utilisation de l'algorithme de propagation de détectabilité proposé sur un cas simple.

V. CONCLUSIONS ET PERSPECTIVES

Dans cet article nous avons proposé une extension de la BDF et une méthode d'attribution de la détectabilité directe et de propagation de la détectabilité en utilisant les connaissances dans la BDF. Cette méthode nous permet de quantifier la détectabilité de mode de défaillance dans une AMDEC, et de créer un lien entre les fonctions qui satisfont les exigences principales et celles qui satisfont les exigences de sûreté de fonctionnement par des outils de détection et de pronostics.

Les futurs travaux se concentrent sur deux axes :
- intégrer des outils et des fonctions de détection et de pronostic ainsi que des procédures opérationnelles concernées (facteur humain, use case, …) dans notre métamodèle ;

- adapter la méthode proposée à la Base de Comportements Dysfonctionnels (BCD).

A long terme, nous espérons pouvoir intégrer les éléments de Prognostics and Health Management (PHM) dans notre approche Médisis.

REFERENCES

[1] Chieb, B. Ingénierie système pour la sûreté de fonctionnement de gamme de produits. Application aux système de levage sûrs Thèse de doctorat, Université d'Orléans, 2020.

[2] Cressent, R. Valorisation de l'Ingénierie Système à Base de Modèles, pour l'analyse de sûreté de fonctionnement des systèmes complexes critiques intégrant des COTS. Thèse de doctorat, Université d'Orléans, 2012.

[3] David, P. Contribution à l'analyse de sûreté de fonctionnement des systèmes complexes en phase de conception : application à l'évaluation des missions d'un réseau de capteurs de présence humaine. Thèse de doctorat, Université d'Orléans, 2009.

[4] Zio E. Reliability engineering: Old problems and new challenges[J]. Reliability engineering & system safety, 2009, 94(2): 125-141.

[5] Dakil M, Simon C, Boukhobza T. Generic methodology for the probabilistic reliability assessment of some structural properties: a graph theoretical approach[J]. International Journal of Systems Science, 2015, 46(10): 1825-1838.

[6] Stamatis D H. Failure mode and effect analysis: FMEA from theory to execution[M]. Quality Press, 2003.