



HAL
open science

ANALYSE DE SURETE DE FONCTIONNEMENT D'UNE LIGNE DE PRODUITS : DE L'INGENIERIE SYSTEME A L'AMDEC PARAMETRIQUE

V. Idasiak, B. Chieb, Frédéric Kratz

► **To cite this version:**

V. Idasiak, B. Chieb, Frédéric Kratz. ANALYSE DE SURETE DE FONCTIONNEMENT D'UNE LIGNE DE PRODUITS : DE L'INGENIERIE SYSTEME A L'AMDEC PARAMETRIQUE. Congrès Lambda Mu 21 “ Maîtrise des risques et transformation numérique : opportunités et menaces ”, Oct 2018, Reims, France. hal-02075399

HAL Id: hal-02075399

<https://hal.science/hal-02075399v1>

Submitted on 21 Mar 2019

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.

ANALYSE DE SURETE DE FONCTIONNEMENT D'UNE LIGNE DE PRODUITS : DE L'INGENIERIE SYSTEME A L'AMDEC PARAMETRIQUE

DEPENDABILITY ANALYSIS OF A PRODUCT LINE: FROM SYSTEM ENGINEERING TO PARAMETRIC FMECA

Chieb B., Idasiak V. et Kratz F.
Laboratoire PRISME – INSA Centre Val de Loire
88 boulevard Lahitolle, Bourges 18000, France
{nom.prénom}@insa-cvl.fr

Résumé

De plus en plus d'industriels développent des lignes de produits adoptant le concept de ligne de produits afin de rationaliser leur conception. Sans outil d'analyse adéquat, mener une étude de sûreté de fonctionnement d'une ligne de produits conduit à réaliser des études de sûreté de fonctionnement pour chaque produit. Ce qui nous amène à nous interroger sur la nécessité de refaire les analyses de parties déjà traitées, et donc sur la nécessité de développer une ligne de produits sûrs.

Notre objectif est de fournir un support d'analyse de type AMDEC paramétrique permettant de conduire une analyse au niveau de la ligne de produits. Cette AMDEC paramétrique permettra la déclinaison rapide des analyses pour chaque produit.

Summary

Increasingly industrials are developing product lines that adopt the product line concept in order to streamline their design. Without adequate analysis tools, conducting a functional dependability study of a product line leads to carrying out dependability studies for each product. This leads us to question the need to redo the analysis of the parts already treated, and therefore the need to develop a safe product range.

Our goal is to provide parametric FMECA analysis support for conducting a product line level analysis. This parametric FMECA then allows the rapid extending of analyzes for each product.

1. Introduction

Dans le domaine de l'ingénierie des systèmes, la plupart des méthodologies pour la conception et l'analyse des systèmes complexes sont développées pour un produit unique. Face aux contraintes économiques, le développement d'un produit doit être rapide, à faible coût et doit permettre d'offrir un large choix d'options sûres au client final. Ceci amène l'industriel à développer des stratégies basées sur la définition de ligne de produits. La validation des qualités attendues de chaque produit dépend des options de celui-ci. Il est, dans ce cas, difficile de réaliser une étude de sûreté de fonctionnement avant que le produit soit finalisé. Les travaux de notre équipe consistent à permettre le déclenchement des études de sûreté de fonctionnement au plus tôt dans le cycle d'ingénierie et si possible dès que l'information d'Ingénierie est disponible. Nous basons nos travaux sur l'emploi de modèles structurés et de formalismes standards utilisés par l'industrie.

De fait, l'ingénierie Système Basée sur les Modèles (ISBM) a montré largement son efficacité pour faciliter la réalisation de produit ou système et son affinité avec les outils ou les méthodes de sûreté de fonctionnement. Cependant, définir une ligne de produits impacte naturellement les organisations, notamment les processus et les méthodes d'Ingénierie Système (IS) et les techniques d'analyse de sûreté de fonctionnement. En effet, les produits d'une même gamme vont se différencier par des implémentations physiques ou logiques différentes, voire même par existence de l'ensemble de fonctionnalités disjointes. Ces variabilités sont au centre de notre étude, du choix de leur représentation dans le modèle de la ligne de produit à leur contribution pour la construction de l'AMDEC paramétrée. Dans cet article, nous commencerons par présenter le processus d'IS et ses activités afin de mettre en évidence les concepts élémentaires utilisés dans notre méthodologie. Puis, nous présenterons deux démarches d'analyses de sûreté de fonctionnement appliquées à une gamme de produits industriels définie par un modèle de ligne de

produits ou bien simplement produit par produit. Pour finir, nous réaliserons une étude comparative des deux démarches proposées et nous conclurons.

2. Processus d'ingénierie système

Lors de la conception d'une ligne de produits qui permet la définition de produits ce différenciant par des implémentations physiques ou logiques différentes, voire même par l'existence d'ensemble de fonctionnalités disjointes, un nouveau concept d'IS apparaît : les points de variation. Ces derniers induisent des structures physiques et logiques différentes pour chaque produit pouvant être gérées au niveau de la ligne de produit par le concept de variante. Les concepts de la modélisation de variante sont connus en informatique sous le nom de SPL (Software Product Line) [Lee et al., 2002], ils introduisent notamment le concept Feature Model. La technique de modélisation Feature Model est basée sur des concepts logiques entre les fonctionnalités du modèle. Elle est utilisée pour définir la similitude et la variation dans les modèles et elle fournit un support à la gestion cohérente des variabilités. Cette technique a été adaptée par [Weilkiens, 2016] pour être utilisée dans la modélisation en SysML. Elle a donné naissance à une démarche nommée VAMOS : « **V**ariant **M**odeling with **S**ysML ». Nous pouvons retenir que dans le cas d'une ligne de produits, la production d'une analyse AMDEC sera soumise à des phénomènes permanents de modifications du modèle du système à étudier. Les outils actuels de l'IS gèrent l'impact des modifications du modèle pour ses propres outils de Validation et Vérification mais ne transmet pas ses informations aux outils externes. C'est donc aux processus d'exploitation des informations du modèle d'IS de réaliser l'identification des variabilités et d'identifier leurs impacts dans l'analyse dont ils ont la charge.

Les méthodes d'ingénieries ont pour objectif de cadencer, de manière rationnelle, ces activités en tenant compte de la

dynamique des projets et des organisations. Nous ne devons pas considérer la chronologie de production des données, si nous voulons fournir une solution indépendante des méthodes d'IS. Ainsi, ces activités peuvent être réalisées en parallèle ou en cascade selon le contexte industriel de chaque projet. Notre proposition est indépendante d'outil ou de langage de modélisation. Le modèle du processus d'IS représenté sur la figure 1 a six activités connectées par leurs flux d'informations échangées. Ces activités vont générer des informations (Data Store) qui seront utilisées par les outils d'analyse.

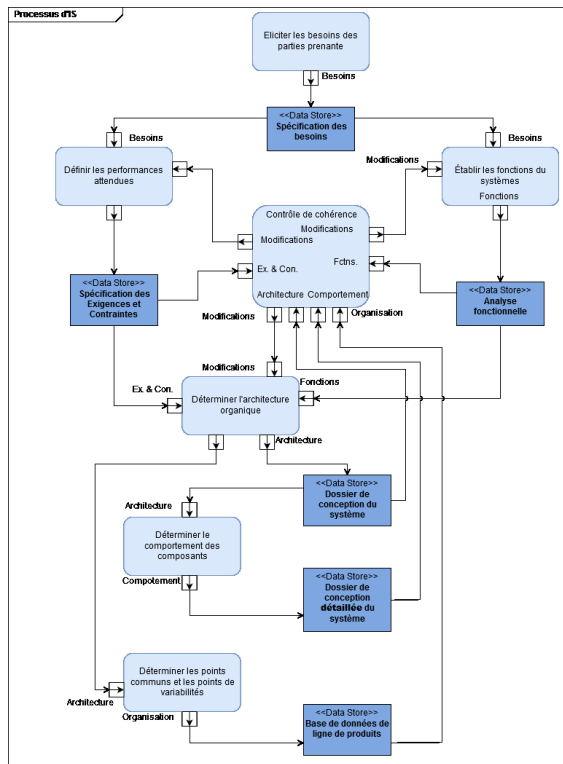


Figure 1. Activités du processus d'IS adopté pour une ligne de produits

Les données identifiées dans chaque activité dans le processus d'IS sont les suivantes :

2.1. Elicitation des besoins fonctionnels du système et de son environnement :

Cette activité permet d'identifier l'ensemble des besoins (**B**) satisfait par le système. Les besoins généraux sont détaillés par des besoins plus spécifiques formant ainsi une structure hiérarchique par la relation *R (Refined by)*. Dans un modèle SysML, les besoins sont représentés dans le diagramme de cas d'utilisation (i.e. Use case Diagram). Importés dans l'AMDEC, ces besoins porteront l'information de ce que le système ne pourrait plus réaliser en cas de défaillance.

2.2. Analyse fonctionnelle du système :

Cette activité permet d'identifier l'ensemble des fonctions système (**F**) qui sont également définies dans une structure hiérarchique, où une fonction peut être détaillée par plusieurs fonctions plus précises en utilisant la fonction *R (Refined by)*. Les fonctions de haut niveau existent pour satisfaire un besoin, ce qui est représenté par la relation *StB (Stated by)* entre ces fonctions et les besoins.

Les fonctions de bas niveaux sont associées à un composant qui sera le support de leur exécution et elles sont définies par la relation *ReB (Realised by)*.

Les fonctions sont représentées en modèle SysML dans des diagrammes d'activités (Activity Diagram).

Quelques soient leurs niveaux de détail (bas-niveau ou haut-niveau), les fonctions transforment des flux qu'elles s'échangent ce qui nous permet de définir :

Si une fonction f_x génère un flux utilisé par la fonction f_y , nous pouvons définir les relations successeur et prédécesseur entre les fonctions f_x et f_y .

On note : $f_x \rightarrow f_y$ (f_x est le prédécesseur de f_y), et donc f_y est successeur de f_x . L'identification de ces fonctions et des relations permettra de proposer dans l'AMDEC la propagation éventuelle d'un dysfonctionnement à travers les couples (fonction, flux).

2.3. Définition des exigences et des performances attendues :

Cette activité permet d'identifier les exigences (**E**) à partir desquelles un système est modélisé. Les exigences fonctionnelles sont liées à des fonctions de bas-niveau par la relation *SaB (Satisfy by)*.

Les exigences, dans un modèle SysML, sont représentées dans un diagramme d'exigences (Requirement Diagram). Importés dans l'AMDEC, les exigences non satisfaites porteront une information plus précise de ce que le système peut être amené à ne plus réaliser en cas de défaillance.

2.4. Définition des composants et de leur organisation :

Cette activité permet d'identifier l'ensemble des composants (**S**) définissant le système, qui est également défini par une structure hiérarchique de composition, *R (Refined by)*. Les composants sont alloués à des fonctions de bas niveaux f avec la relation *ReB (Realised by)* afin d'assurer la cohérence entre la structure fonctionnelle et la structure organique. L'identification de cette structure de composants permet d'identifier dans l'AMDEC, les modes de défaillances possibles en cas de défaillance physique des composants.

Description du comportement des composants :

Cette activité permet d'identifier le comportement du système et de ses composants. La représentation des comportements des composants se fait avec les diagrammes de séquence et d'état (Sequence et State machine Diagram). Ces informations ne sont pas utilisées pour la réalisation de l'AMDEC.

2.5. Identification des points communs et des points de variabilités :

Cette activité permet de construire une base de données de la ligne de produits en identifiant les points communs (les artefacts obligatoires : fonctions ou composants) qui seront réutilisés pour tous les produits de la ligne de produits. Sont également identifiés, les points de variabilités (les éléments optionnel *OP*, les dépendances aux contraintes comme : *Requires*, *Excludes*, *OR*, et *Alternative*) qui peuvent être configurés. Cette identification permet d'avoir une AMDEC paramétrée. Ces caractéristiques sont définies dans le modèle Feature Model ainsi que dans l'organisation et les relations du modèle SysML (Package Diagram).

2.6. Contrôle de cohérence de tous les éléments :

Cette activité a pour but de contrôler les cohérences entre l'ensemble des entités issues des activités du processus d'IS et d'assurer la compatibilité de ce qui a été réalisée grâce à ces activités avec les exigences définies par les parties prenantes. Les entités qui peuvent être contrôlées et modifiées sont les suivantes : fonctions, composants, besoins et exigences. L'activité de contrôle de cohérence est facilitée par le renseignement des relations de traçabilité réparties au travers de l'ensemble des activités d'IS.

La figure 2 illustre le méta-modèle des entités issues des activités du processus d'IS présentées en figure 1. Nous basons notre outil de génération de la pré-AMDEC (AMDEC préliminaire) sur ce méta-modèle. Celui-ci comporte une partie de base, contenant quatre entités principales (Need, Function, Component et Requirement) et l'entité qui représente le flux fonctionnel comme défini précédemment. La deuxième partie contient les entités qui sont dédiées à la définition de la ligne de produits et qui sont représentées par le cadre en pointillé. Toutes les relations de hiérarchisation et d'allocation sont également reprises dans les associations de ce méta-modèle.

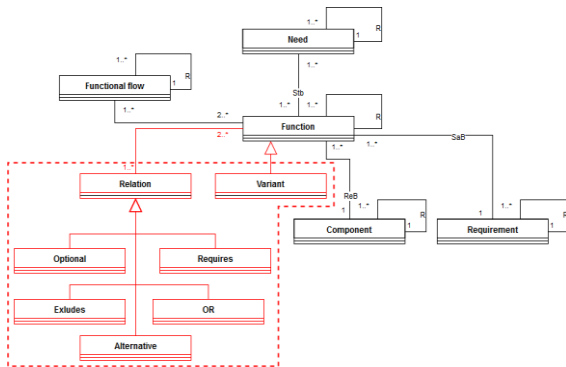


Figure 2. Meta-modèle des entités du processus d'IS.

3. Démarches d'analyse

Après avoir expliqué quand et où l'information du modèle système de ligne de produits était créée (cf. Figure.1) et après avoir expliqué comment cette information était structurée (cf. Figure.2), nous présentons comment cette information est utilisée pour alimenter une AMDEC dans deux démarches basées sur des modèles en SysML et en Feature Model.

La première démarche est basée multi-modèles qui représentent plusieurs versions du modèle SysML, où chaque version décrit un produit spécifique de la ligne de produits. La deuxième démarche est basée sur un modèle générique de la ligne de produits (modèle de variabilité) représenté en SysML pour l'aspect structurel, fonctionnel et comportemental, et en Feature Model pour l'aspect gestion de ces variabilités.

3.1. Démarche basée multi-modèles

Le processus de génération d'AMDEC (Figure. 3) est un processus de la méthodologie MéDISIS ([David et al., 2009] et [Cressent et al., 2012]). Ce processus automatique est basé sur trois phases : la génération du modèle d'analyse, la sélection des modes de défaillance et la détermination du comportement dysfonctionnel du système. Ces trois phases sont décrites par la figure 3 et reposent sur deux méta-modèles : celui des informations disponibles dans un modèle d'IS (cf. Figure.2) [Kajdan et Idasiak, 2015], et celui de l'AMDEC ciblée [Chieb et al., 2017]. La première phase consiste à générer automatiquement une pré-AMDEC en utilisant le modèle d'ingénierie système qui contient : chaque élément à étudier (fonction ou composant), ses prédécesseurs et successeurs selon les flux fonctionnels ou physiques. Cela est obtenu grâce à des règles qui permettent d'identifier le type d'information pour la génération automatique de la pré-AMDEC. Prenons l'exemple d'une pré-AMDEC fonctionnelle :

Fonction	Organe	Mode de Défaillance	Cause locale	Effet local	Besoin	Besoins système	Exigence fonctionnelle
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)

1) La liste des fonctions étudiées

Soit F_e l'ensemble des fonctions de bas niveau faisant l'objet de l'étude de sûreté de fonctionnement. Une fonction f_e sera élément de F_e si elle n'est détaillée par aucune autre fonction. Elle satisfait la condition suivante :

$$\nexists f \in F \mid f_e \mathbf{R} f \text{ ou bien } \{(f, f_e) \in F \times F_e \mid f_e \mathbf{R} f\} = \emptyset.$$

\mathbf{R} : relation « *Refined by* » définie précédemment.

F_e : ensemble des fonctions étudiées du système (les fonctions de bas niveau).

F : ensemble des fonctions du système.

2) La liste des composants supports

On considère un composant s comme un composant support si et seulement si la fonction f_e est réalisée par ce composant. En termes mathématiques :

$$\exists s \in S \mid f_e \mathbf{ReB} s.$$

\mathbf{ReB} : relation « *Realised by* » définie précédemment.

S : ensemble des composants du système.

3) Le mode de défaillance

Les informations sur les modes de défaillances ne sont pas incluses dans le modèle. Il appartient donc à l'expert de sûreté de fonctionnement de renseigner les modes de défaillance possibles pour chaque fonction étudiée. Il peut utiliser la base de données de comportements dysfonctionnels (David et al., 2009) fournie ou réutiliser une pré-AMDEC similaire avec la capitalisation des données. Dans tous les cas, pour initier la définition du comportement dysfonctionnel des fonctions étudiées, l'expert précise le mode de défaillance m de chacune de ces fonctions $f_e \in F_e$ et qui sera représenté par la suite par le couple (f_e, m) .

4) Cause

On considère la fonction f_c comme cause possible d'une défaillance d'une fonction f_e si et seulement si la fonction f_c est le prédécesseur de la fonction étudiée f_e , autrement dit, f_c envoie un flux fonctionnel vers f_e . En termes mathématiques :

$$\exists f_c \in F \mid f_c \rightarrow f_e.$$

F : ensemble des fonctions du système.

5) Effet local

On considère que la fonction f_e a un effet possible de défaillance sur la fonction f_p si et seulement si la fonction f_p est le successeur de la fonction étudiée f_e , autrement dit, f_p reçoit un flux fonctionnel de f_e . En termes mathématiques :

$$\exists f_p \in F \mid f_e \rightarrow f_p.$$

F : ensemble des fonctions du système.

6) Besoins

On considère un besoin b_e , comme un besoin réalisé par une fonction f ($b_e \mathbf{StB} f$) si et seulement si, cette fonction f (fonction de haut niveau) est détaillée jusqu'à f_e . En termes mathématiques :

$\exists b_e \in B \mid b_e \text{ StB } f \text{ et } \{\forall f_i \in F \mid f (R^{(-1)})^* f_i\} = \emptyset \text{ et } \exists f_e \in F_e \mid f_e (R^{(-1)})^* f.$

B : ensemble des besoins du système.

StB : relation « Stated by » définie précédemment.

$(R^{(-1)})^*$: relation inverse de « Refined by » itérée * fois.

7) Besoins système

L'identification des besoins système se fait à travers le suivi des chemins hiérarchiques (définis par la relation R) parmi l'ensemble des besoins. L'ensemble P des chemins entre les besoins de plus haut niveau et les besoins de bas niveau est donné par :

$P = \{(b_1, b_2, \dots, b_k) \mid (b_i, b_{i+1}) \text{ et } \{b \mid b_k R b\} = \emptyset, b_k \text{ StB } f \text{ et } \{\forall f_i \in F \mid f (R^{(-1)})^* f_i\} = \emptyset \text{ et } \exists f_e \in F_e \mid f_e (R^{(-1)})^* f \text{ et } i \in [1; k-1], k \in \mathbb{N}\}.$

b_k : besoin de bas niveau tel que $b_k \text{ StB } f_e$.

b_1 : besoin de haut niveau, c.-à-d. b_1 ne détaille aucun autre besoin.

8) Exigence fonctionnelle

On considère e_f comme une exigence fonctionnelle si et seulement s'il existe une fonction f_e qui satisfait cette exigence e_f . En terme mathématiques :

$\exists e_f \in E \mid e_f \text{ SaB } f_e.$

SaB : relation « Satisfay by » définie précédemment.

E : ensemble des exigences

La seconde phase consiste en un filtrage des lignes de la pré-AMDEC. En effet, l'expert sélectionne les éléments (fonctions ou composants) à prendre en compte dans l'étude avec les modes de défaillance associés, ainsi que leurs effets possibles.

La troisième phase, quant à elle, consiste à déterminer le comportement dysfonctionnel (table de propagation et AMDEC consolidée) du système afin de générer l'AMDEC finale, l'outil propose notamment de connecter les effets locaux d'un élément aux causes d'un autre élément.

Cette phase consiste à propager les modes de défaillance des éléments prédécesseurs causant les défaillances vers l'élément étudié. L'outil génère des tables de propagation de défaillance. L'analyste réalise alors des choix permettant d'associer la défaillance d'un élément prédécesseur à la défaillance de l'élément étudié ; lors d'une propagation les modes de défaillance peuvent évoluer. Afin de construire les chemins des effets locaux pour permettre de suivre la propagation d'une défaillance, en partant de la cause interne d'une fonction étudiée, les chemins des effets locaux identifiés correspondent chacun à une succession des éléments successeurs de l'élément étudié, associé à leurs modes de défaillance.

Pour cette démarche, un modèle de base V_0 (Produit version 0) est nécessaire pour déclencher l'étude de sûreté de fonctionnement. Cela permet la génération automatique d'une Pré-AMDEC V_0 .

La pré-AMDEC doit être complétée par l'expert de sûreté de fonctionnement à l'aide de l'outil de détermination des modes de défaillance.

Lors de la prise en compte de variabilité, de nouvelles lignes d'AMDEC sont générées, donnant ainsi la version V_i sur la base de la version précédente $V_{(i-1)}$ (cf. Figure. 3).

La capitalisation des décisions est portée par l'AMDEC, le modèle du système et la réutilisation de ses décisions sont pilotés par l'outil.

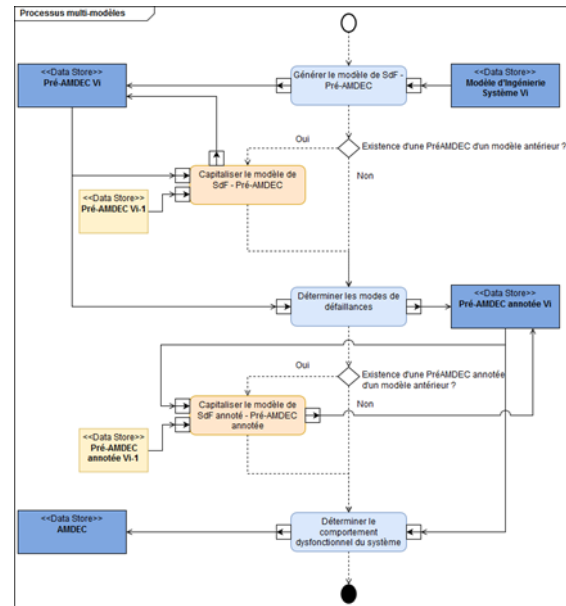


Figure 3. Processus de la démarche basée multi-modèles.

3.2. Démarche basée modèle de variabilité

Cette démarche consiste à développer une approche par ligne de produits avec la modélisation explicite des points communs et des variabilités des artefacts du système et la dérivation de produits à partir d'une base commune de la ligne de produits.

Dans cette démarche, un modèle de ligne de produits doit être défini. Ce modèle représentera les caractéristiques communes et les caractéristiques variantes de la ligne de produits. Ces caractéristiques peuvent être représentées avec Feature Model. La conception d'une architecture de ligne de produits consiste à implémenter les fonctions système spécifiées dans un modèle de fonctionnalité de ligne de produits à l'aide d'outils basés sur des langages comme SysML.

Le modèle de ligne de produits en SysML contient tous les éléments définis dans le modèle Feature Model et les décrit d'un point de vue fonctionnel et / ou organique. Ce modèle est composé de trois parties, une partie Core qui représente les éléments communs de l'ensemble des produits de la gamme qui sont identifiés par la relation 'MA' dans le modèle en Feature Model et la partie variante qui représente les éléments variant de l'ensemble des produits de la gamme et qui sont identifiés par la relation 'OP' et les relations orthogonales ('Alternative' et 'OU'). La troisième partie est composée d'éléments de la partie Core en interface avec des éléments de la partie variant. Ils inter-changent des flux entre eux.

En effet, les concepts de gestion de variabilité ajoutés dans la description de la ligne de produits vont être représentés dans deux nouvelles colonnes de la pré-AMDEC paramétrique. Ces nouvelles colonnes, appelées « RH » et « RO », soulignent l'association entre les variantes et les scénarios de configuration valides. Ces concepts présentent les relations de dépendance (hiérarchique et orthogonale), afin de pouvoir construire une pré-AMDEC qui corresponde à un modèle de produit dans la ligne de produits. Une base de données permet de stocker et de retrouver l'intégralité des informations permettant de maintenir et gérer en configuration l'ensemble des produits de la ligne de produits.

Dans la pré-AMDEC paramétrique, la colonne de relation hiérarchique reprend les relations d'optionnalité 'OP' ou d'obligation 'MA'. Ces deux propriétés permettent d'identifier les éléments invariants représentés par le signe 'MA' appartenant à la partie Core et les éléments variant représentés par le signe 'OP' appartenant à la partie variante. Quant à la colonne des relations orthogonales, celle-ci représente les relations d'implication 'RE' et d'exclusion 'EX' entre les éléments à étudier (les éléments variables). Ces deux propriétés permettent d'expliquer les dépendances entre les éléments. En particulier, nous avons les relations 'OR' et 'XOR' qui sont également considérées comme des relations orthogonales.

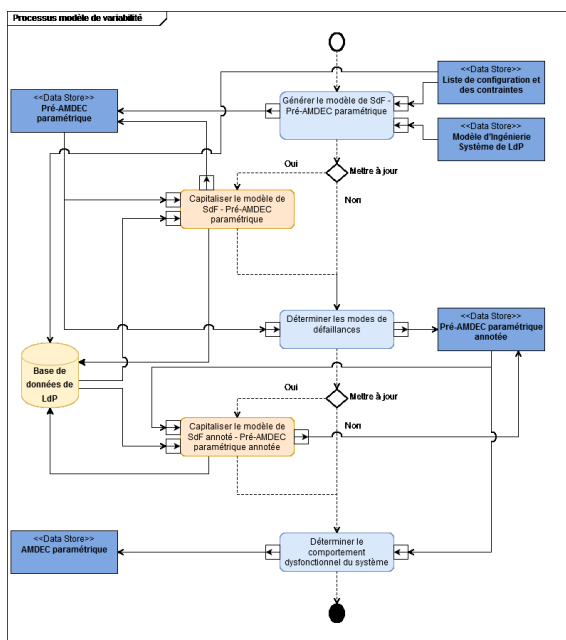


Figure 4. Processus de la démarche basée modèle de variabilité.

Grâce à cette représentation globale de l'AMDEC, nous serons en mesure d'effectuer l'analyse de la sûreté de fonctionnement des produits dérivés de la ligne de produits sans utiliser leur modèle, sur lequel nous pourrions appliquer le processus AMDEC de MéDISIS existant. Cependant, pour chaque produit, il serait nécessaire de refaire une partie d'analyse identique. Or, nous cherchons à optimiser les analyses de type AMDEC dans le cas d'une ligne de produits.

Les éléments parallèles qui ont les mêmes éléments d'entrée et éléments de sortie peuvent être identifiés implicitement. Étant donné que le principe de redondance, qui est représenté par la structure parallèle, permet de fournir plusieurs ressources redondantes pour réaliser la même fonction. Des variantes reliées à un point de variation peuvent ressembler à des systèmes en redondance. La relation « XOR » permet de distinguer les fausses redondances induites à certains points de variabilité dans l'analyse et la modélisation de la ligne de produits.

La prochaine étape est de générer l'AMDEC à partir de la ligne de produits, nous allons suivre les mêmes tâches que pour le premier processus. La tâche de filtrage consiste à sélectionner les lignes de pré-AMDEC utilisées pour l'analyse (les éléments qui composent les lignes de cette pré-AMDEC seront tous pris en compte dans la phase de la détermination des modes de défaillance). Après avoir déterminé les modes de défaillance liés à chaque élément de la ligne de produit dans la pré-AMDEC paramétrique, la

phase suivante est consacrée à la génération des tables de propagation. Les mêmes conditions sont prises en ce qui concerne le premier processus. L'association entre les modes de défaillance et l'élément étudié ainsi que leurs causes et effets peuvent être spécifiques à chacune de ces configurations. Après avoir obtenu la ligne de produits AMDEC, la dernière étape peut être effectuée. Cette dernière étape consiste à obtenir une AMDEC pour un produit spécifique à partir de l'AMDEC paramétrée en utilisant les listes de configuration et les nouvelles colonnes « RH » et « RO ».

4. Comparaison des démarches

Afin d'analyser l'organisation structurée de l'ensemble des éléments constituant le modèle de ligne de produits, nous avons introduit les notions de « Core », « Variante » et « Core-variant » du modèle de ligne de produits ce qui permet d'étudier l'influence des variabilités sur une étude AMDEC. Ces notions sont inspirées de la modélisation variante en SysML définie par [Weilkiens, 2016].

Le défi consiste donc à séparer la partie variante de la partie commune et à gérer les dépendances.

Les éléments de la ligne de produits sont décomposés en trois parties (Chieb et al., 2018):

- **Partie Core** : contient les éléments invariants du système et contient tous les éléments utilisés dans toutes les configurations du système. Ces éléments ne dépendent d'aucun élément variant ;
- **Partie Core-Variant** : contient les éléments invariants du système qui sont liés à au moins à un élément variant ;
- **Partie Variants** : contient les éléments de variante du système, ils n'existent que dans certaines configurations.

La table 1 présente la classification des lignes d'AMDEC, en fonction des informations pouvant y être introduites : soit par une variabilité d'une ligne de produits ou suite à la mise à jour d'un modèle du système et par conséquent la prise en compte d'un nouveau mode de défaillance.

Si l'on généralise la nature de l'information traitée dans une AMDEC en fonction de sa dépendance au changement du système, le modèle du système comporte trois types d'éléments qui représentent le triplet d'une ligne d'AMDEC (cf. table 1) :

- E_s : élément étudié ;
- E_c : élément cause de défaillance ;
- E_e : élément effet de défaillance.

Table 1. Classification des lignes d'AMDEC.

E_s	E_c	E_e	Type de ligne	Type ensemble
Core	Core ou CI	Core	T1	{Core}
Core	Core ou CI	Variante	T2	{Core} U {Variante}
Core	Variante	Core	T3	{Core} U {Variante}
Core	Variante	Variante	T4	{Core} U {Variante}
Variante	Core	Core	T5	{Core} U {Variante}
Variante	Core	Variante	T6	{Core} U {Variante}
Variante	Variante ou CI	Core	T7	{Core} U {Variante}
Variante	Variante ou CI	Variante	T8	{Variante}

Note : CI : Cause Interne (spécifique au composant).

Toutes modifications du modèle d'IS (ajout d'une variation dans la démarche basée multi-modèle ou changement de configuration dans la démarche basée modèle de variabilité)

impactent l'analyse de sûreté de fonctionnement, car elles modifient la structure fonctionnelle et/ou organique du produit. Afin d'étudier l'impact de ses variations, il est nécessaire de classer les lignes de l'AMDEC en fonction du triplet (cause, élément, effet) étudié. Ce qui permet de définir l'ensemble des lignes d'AMDEC impactées, ainsi que les informations susceptibles d'être réutilisées. L'identification et la conduite des modifications est alors possible par notre outil.

Dans la partie « Core », les éléments du triplet sont invariables et les informations de ce type de ligne d'AMDEC (T1) sont donc réutilisables. Les lignes d'AMDEC de type T1 comporte uniquement des éléments de la partie « Core » du système. Ces lignes de type T1 sont identifiées lors de la capitalisation des données d'une version d'AMDEC à une autre dans la démarche basée multi-modèles. Le gain de temps est obtenu ici, grâce la réutilisation des analyses précédentes attachées aux éléments « Core ».

Les éléments (triplet) constituant les lignes de type T8 appartiennent tous à la partie « variant ». Cette partie de l'analyse pourra être capitalisé pour tous les systèmes utilisant cette variante. Les outils de Versionning sont appropriés à réutilisation de ses informations.

Toutes les autres lignes sont mixte (Core et Variant) et n'ont pas le même impact sur la constitution de l'AMDEC finale. Prenons l'exemple de la ligne T3. Dans le cas de la démarche multi-modèles, et grâce à la capitalisation des données, l'expert n'a plus qu'à sélectionner les effets potentiels de défaillance de l'élément. Dans le cas de la démarche modèle de variabilité, et grâce à la configuration, l'expert peut optimiser le choix de l'effet de défaillance de l'élément.

Dans le cas des lignes type T4, l'élément étudié appartient à la partie « Core ». Donc l'expert peut réutiliser les informations renseignées sur cet élément. Si l'on dispose des informations sur la variabilité (E_c , E_e) une nouvelle ligne peut être générée en intercalant ceux-ci avec l'élément étudié.

Les lignes de type T6 sont symétriques aux lignes de type T3, ainsi si l'on dispose de l'analyse de la variabilité, cela revient à concaténer l'élément « Core » (ici cause) aux lignes d'AMDEC de la variabilité pour les éléments E_s de la variabilité. L'expert n'ayant plus qu'à sélectionner les lignes pertinentes.

Lorsque les produits sont définis dans la Ligne de produits, il est difficile de ne pas répéter la même étude de type AMDEC pour les éléments communs de ces produits. Afin d'optimiser le temps nécessaire pour produire l'AMDEC pour chaque produit, nous avons d'abord étudié comment les modèles de la Ligne de produit influent sur les informations nécessaires à la modélisation d'IS, puis sur leur analyse de sûreté de fonctionnement.

Le tableau suivant montre les caractéristiques de chaque démarche.

Table 2. Caractéristiques des deux démarches.

Caractéristiques	Démarche basée multi-modèles	Démarche basée modèle de variabilité
Réutilisation des données	x	x
Suivi de modifications	x	x
Base de données de Ligne de Produits		x
Capacité de prendre en compte une variabilité	x	x
Gain de temps	+	++

Les points forts des démarches sont la capacité de réutiliser les données d'analyse et de prendre en compte un nouvel élément qui représente une variabilité dans le système. Une caractéristique spécifique à la démarche basée sur un modèle variabilité est d'avoir une base de données des artefacts communs et des variabilités. Ces résultats nous ont permis d'avoir un gain très important sur le temps d'analyse et d'être en mesure de réutiliser des parties de cette analyse pour un autre système de produit de la même ligne de produits.

5. Cas d'étude

Le FAST BRAKE-PL est un système de levage doté d'un frein de sécurité opérationnel pour treuil de levage (FOST) (Chieb et al., 2017). Ce système a été choisi pour évaluer les deux démarches sur la base d'un modèle de ligne de produits proposé pour soutenir l'analyse automatique de la sûreté de fonctionnement de plusieurs produits dans la ligne de produits. Les résultats de l'application de l'approche proposée aux produits FAST BRAKE-PL ont été utilisés comme preuve de concept.

Le système se compose de trois parties : le sous-système de pièce de transmission qui comprend les deux accouplements élastiques 1 et 2 avec limiteur de couple qui constitue une alternative aux accouplements élastiques 1, au réducteur de chaîne et au tambour. Le sous-système de contrôle comprend le moteur électrique, deux capteurs de vitesse (un pour le tambour et un pour le moteur) ; cette partie ne contient aucune variation. Le sous-système de sécurité est composé d'un automate de sécurité pour assurer le freinage lorsqu'il y a un écart de vitesse entre le tambour et le moteur, et enfin, le frein standard et le frein de sécurité.

A partir du modèle de la ligne de produits de la figure 5, nous obtenons six configurations possibles valides. Les six configurations sont obtenues en utilisant les contraintes et les relations appliquées dans le modèle de ligne de produits. Nous concentrons notre étude uniquement sur les trois configurations principales.

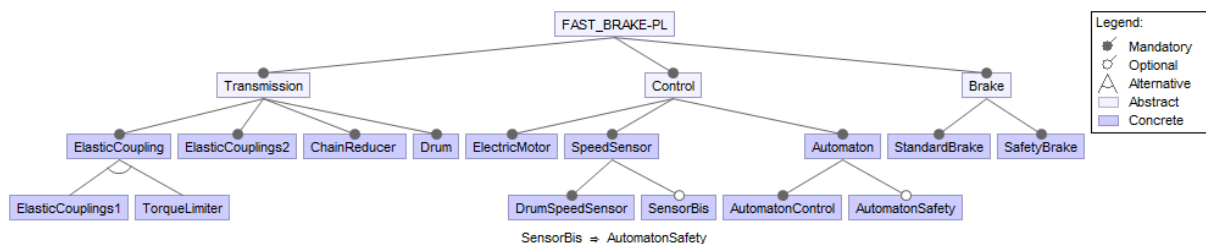


Figure 5. Modèle de FASTE BRAK-PL en Feature Model.

Dans la suite de notre étude de cas, nous avons appliqué la démarche basée modèle de variabilité. Nous avons suivi toutes les étapes de ce processus. A la fin du processus, une AMDEC paramétrique de la ligne de produits AMDEC est obtenue. Le tableau 3 donne un aperçu de l'AMDEC paramétrique du système FASTE BRAK-PL. Il contient la colonne ID qui représente l'identifiant de la ligne et la colonne « Élément » qui contient le nom de l'élément étudié, la colonne « cause », la colonne « effet local » et la colonne du mode de défaillance. Cette AMDEC paramétrique comprend la colonne de note pour la sélection des lignes valides, les colonnes HR et OR pour les relations liées à la variabilité et à la fin le type de ligne pour la classification des lignes d'AMDEC.

Pour obtenir une AMDEC d'un produit à partir de l'AMDEC paramétrique de la ligne de produits, nous avons fait le choix des lignes qui sont spécifiques au produit et procédons ensuite à l'étape de propagation des défaillances.

Par exemple, si nous analysons, la relation "implique" qui est utilisée pour gérer la variation des éléments optionnels « automate de sécurité » et « capteur bis ». Les lignes AMDEC impactées ont plusieurs types (T6, T7 et T8).

Le choix entre deux éléments est présenté dans la pré-AMDEC de notre étude de cas par la relation « XOR ». Par exemple, la présence de l'élément « accouplement élastique 1 » exclut la présence de l'élément « limiteur de couple 1 » et vice-versa. En effet, l'analyse est faite dans le cas d'une configuration de produit sur un seul élément parmi l'ensemble des éléments alternatifs. La relation alternative est appliquée aux éléments optionnels. Ce qui permet à l'expert d'optimiser l'analyse sur les lignes sélectionnées pour cette configuration. Ainsi, l'expert peut réutiliser les informations d'analyse existantes et il peut projeter les analyses qui partagent des caractéristiques communes (i.e. un point de variabilité). Par exemple, dans le système étudié, les éléments « Accouplement élastique 1 » et « limiteur de couple » partagent des caractéristiques structurelles communes matérialisées par la connexion souple entre moteur électrique et réducteur à chaîne. Cela aide l'expert à prendre des décisions génériques dans un premier temps puis compléter l'analyse plus tard.

L'économie de temps lors de la réalisation des AMDEC est de l'ordre de 50% grâce l'outil MéDISIS.

Table 3. AMDEC paramétrique partielle.

Id	Element	Cause	Local Effect	Failure mode	Note	HR	OR	Line type
01	brake	IC	drum	LF	ok	MA	/	T1
02	e_coupling1	IC	ch_reducer	LF	ok	OP	ec1 xor tl	T7
03	sensor_bis	IC	saf_automat	LF	ok	OP	as=>sb	T8

04	spd_sensor	IC	automat	LF	ok	MA	/	T1
05	ele_motor	IC	e_coupling1	LF	ok	OP	ec1 xor tl	T2
06	trq_limiter	ele_motor	ch_reducer	LF	ok	OP	ec1 xor tl	T6

6. Conclusion

Nous avons présenté dans cet article, deux démarches outillées de génération d'AMDEC, adaptées aux lignes de produit. Ces outils permettent de prendre en charge l'aspect itératif et évolutif des systèmes ainsi que leurs grandes complexités. La première exploite la capacité de la méthode MéDISIS à suivre les évolutions d'un modèle d'IS, un produit offrant une variante et alors vu comme une modification du modèle d'origine. Nous avons adapté celle-ci à l'analyse des systèmes type ligne de produits, en intégrant la variabilité dans le méta-modèle des entités du processus d'IS, permettant la détection de celle-ci dans le modèle du système et surtout dans une AMDEC paramétrée par les relations définissant les variabilités.

Les deux démarches permettent un gain de productivité en limitant les nouvelles études quant aux impacts dus aux évolutions de définition technique d'un système. En effet, la capitalisation, rendue possible par ces démarches. Offrant également une base de données commune aux différents produits de la ligne de produits.

La deuxième démarche permet de plus d'initier les analyses de types AMDEC dès la définition de la ligne de produit.

La démarche outillée présentée, peut également être appliquée à la situation ou problématique suivante : Un concepteur définit des alternatives lors de la conception d'un système, certaines s'excluant l'une l'autre et d'autres impliquant des obligations structurelles ou fonctionnelles (i.e. relation XOR, OR, RE, EX). Il est alors possible d'utiliser notre solution afin de choisir, plus rapidement, l'architecture optimale du point de vue sûreté de fonctionnement.

Une interface de l'outil MéDISIS avec la démarche basée multi-modèles est développée afin de faciliter l'utilisation des scripts. Pour la deuxième démarche basée modèle de variabilité l'interface est en cours d'achèvement.

Remerciements

Ces travaux sont financés dans le cadre du projet FUI FOST par les pôles de compétitivité : Mer Méditerranée, Nuclear Valley et ViaMéca. Nous remercions pour leur collaboration les partenaires industriels du projet : FOC transmissions, SERIEL Automatismes et ADC.

Références

David, P., Idasiak, V., & Kratz, F., Reliability study of complex physical systems using SysML. Reliability Engineering & System Safety, 95(4), 431-450. (2010).

Cressent, R., David, P., Idasiak, V., & Kratz, F., Designing the database for a reliability aware Model-Based System Engineering process. Reliability Engineering & System Safety, 111, 171-182. (2013).

Chieb, B., Idasiak, V. & Kratz, F., Analyse de sûreté de fonctionnement d'une ligne de produits à l'aide de son modèle, QUALITA, Bourges, 2017.

Chieb, B., Idasiak, V. & Kratz, F., (2018). Dependability analysis of a product line using its model. ESREL, Trondheim, Norway, 2018.

Kajdan, R., & Idasiak, V. (2015). Model-based systems engineering and failure analysis: Experience feedback. ESREL, Zürich, 2015.

Lee, K., Kang, K. C., and Lee, J., Concepts and Guidelines of Feature Modeling for Product Line Software Engineering, In: ICSR-7: Proceedings of the 7th International Conference on Software Reuse, London, UK, Springer-Verlag, 2002.

Pohl, K., Bockle, G., and van der Linden, F., Software Product Line Engineering: Foundations, Principles, and Techniques. Springer, 2005.

Weilkiens, T. Variant modeling with SysML. MBSE4U, 2016