Expérimentation de la méthodologie MBSA au domaine Spatial

Teddy Bilba, David Mailland, Léa Dumont, Martial Schaff

Institut pour la Maîtrise des Risques
Sûreté de Fonctionnement - Management - Cindyniques

Expérimentation de la méthodologie MBSA au domaine Spatial

BILBA Teddy
Thales Alenia Space
Toulouse, France

MAILLAND David
Thales Alenia Space
Toulouse, France

DUMONT Léa
SOM LIGERON Toulouse
Colomiers, France

SCHAFF Martial
Thales Alenia Space
Toulouse, France

***Résumé* - Cet article présente les résultats des études effectuées au sein de Thales Alenia Space et partagées avec Ligeron afin d'étudier l'intérêt de la mise en place de la méthodologie MBSA (Model Based Safety Assessment) dans le domaine de la conception de systèmes spatiaux.**

La conclusion présente les résultats des tests et suggère des améliorations pouvant être apportées aux outils supportant cette méthodologie.

Mots clefs – MBSA, domaine Spatial, Sûreté de Fonctionnement

I. INTRODUCTION

Le domaine spatial évolue avec l'arrivée de projets innovants comme les ballons stratosphériques autonomes du projet Stratobus de Thales Alenia Space, les charges-utiles de Télécommunication flexibles ou l'essor des programmes de Navigation, faisant de la sécurité des utilisateurs un critère de performance essentiel du Système.

Thales Alenia Space faisant partie des leaders de ces nouveaux marchés est conscient de devoir remettre en cause les méthodes d'analyse traditionnelles de Sûreté de Fonctionnement, de les éprouver et proposer à ses clients et aux autorités de certification des approches de sécurité robustes et fiables.

La méthodologie MBSA (Model-Based Safety Assessment, analyse de sécurité basée sur les modèles) suscite un vif intérêt de la part du monde de la Recherche. Elle consiste à modéliser la propagation de défaillances à l'intérieur d'un Système à l'aide d'un langage formel appliqué à un modèle de ce même Système. Relativement récente elle promet des échanges renforcés entre l'ingénierie Système et responsable Safety par l'utilisation de modèles partagés.

Les objectifs de cet article sont de valider, ou non l'intérêt de la méthodologie MBSA pour le domaine spatial puis de suggérer des adaptations ou des améliorations potentielles.

II. CONTEXTE

La méthodologie MBSA semble à priori destinée aux études de Sécurité (au sens « Safety »). Dans le cadre de cet article, l'intérêt en a cependant été évalué pour l'ensemble des analyses de Sûreté de Fonctionnement et de Sécurité du domaine spatial.

A. Rappel des analyses RAMS typiques du domaine Spatial

L'AMDEC est sans doute l'analyse la plus connue, elle consiste à déterminer l'effet de la panne d'un composant sur le véhicule spatial ou le système. Analyse montante par excellence, elle est dans ce cas construite en plusieurs étapes à partir des analyses AMDEC des segments ou des équipements.

L'AMDEC fonctionnelle est pratiquée dans les projets de la Navigation par Satellites avec comme finalité, similaire alors aux Analyses fonctionnelle des dangers (FHA) du domaine aéronautique, d'allouer une criticité à chacune des fonctions du Système en considérant ses modes de défaillance potentiels et leurs conséquences sur le fonctionnement même du dit Système.

L'analyse par arbres de défaillance (FTA) étudie l'ensemble des pannes ou combinaisons de pannes pouvant mener à un événement redouté. Dans une perspective autre, de maîtrise du risque opérationnel sur les services et les coûts de possession, l'approche consiste à établir le bloc diagramme de disponibilité ou la chaine fonctionnelle d'un segment sol par exemple, dans le but d'évaluer la performance de disponibilité opérationnelle.

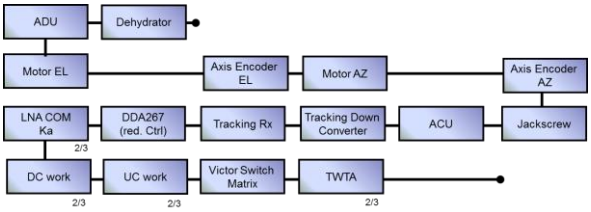

Fig. 1. Bloc diagramme de fiabilité de la chaine fonctionnelle de communication d'une antenne en Bande Ka.

Dans le cas de systèmes non réparables, typiquement au niveau d'un satellite, l'analyse de fiabilité sera suffisante. Celle-ci consiste à calculer la probabilité qu'un Satellite soit toujours capable d'assurer sa mission après un certain temps.

Dans le domaine spatial, l'ingénieur RAMS est également responsable de la démonstration de la tenue de certaines exigences de performance telles que la disponibilité, l'intégrité ou encore la continuité d'un système hautement reconfigurable. Afin d'appréhender cette complexité, l'analyste RAMS développe un modèle spécifique (sous Excel en Visual Basic), a recours aux réseaux de Petri ou utilise une méthode de calcul exacte comme les chaines de Markov.

En ce sens l'arrivée de nouvelles approches basées sur les modèles a donc été accueillie très favorablement par les analystes RAMS, habitués aux modélisations de Systèmes complexes à l'aide d'outils adaptés.

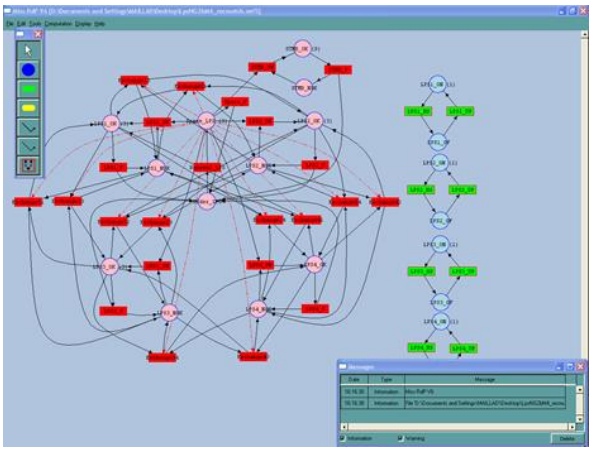

Fig. 2. Réseau de Petri évaluant la disponibilité d'un système présentant une importante flexibilité dans ses reconfigurations possibles.

B. Principe de la méthodologie MBSA.

Semblable à la méthode de Descartes, l'analyse MBSA réduit la complexité en identifiant les briques de bases, éléments simples à manipuler. En instanciant ces éléments et en décrivant les interactions entre eux, il est possible de créer un modèle complexe du Système et de ses reconfigurations en cas de panne. Cette méthodologie présente plusieurs avantages, notamment l'approche incrémentale qui réduit la complexité du système.

Une fois le système décomposé en blocs élémentaires et les chemins de dépendances explicités, l'humain cède alors sa place à la machine, dont il utilise la puissance de calcul pour étudier un très grand nombre de scénarios et effectuer des calculs de probabilité par exemple.

C. Le code AltaRica.

Le langage formel utilisé dans le cadre de cette étude est l'AltaRica. Ce langage a été conçu par le Laboratoire Bordelais de Recherche en Informatique ([3]) et existe maintenant dans sa 3ème version. D'un point de vue structurel, il s'assimile aux langages orientés-objet.

AltaRica repose sur le principe de machine d'états. Plusieurs états se comportant différemment sont définis et il est possible de passer d'un état à un autre par des transitions gardées, comme dans l'exemple ci-après.

```
domain States{WORKING, STANDBY, FAILED}

class GenericComponent
        // *** State variables
    parameter States initial = WORKING;
    States self (init = initial);
        // *** Flow variables
    Boolean outputFlow (reset = false);
        // *** Events with associated laws
    parameter Real lambda = 7.2e-4;
    parameter Real mu = 1.5e-2;
    event failure (delay = exponential(lambda));
    event start(delay = Dirac(1.0));
    event repair (delay = exponential(mu));
    // *** Changes on state variables
transition
    failure : self == WORKING -> self := FAILED;
    start : self == STANDBY -> self := WORKING;
    repair : self == FAILED -> self := STANDBY;
    // *** Changes on flow variables
assertion
    outputFlow := (self == WORKING);
end
```

Fig. 3. Exemple de code AltaRica décrivant 3 états d'un composant.

III. EXPERIMENTATION SUR EGNOS

A. Choix d'une fonction à modéliser et mise en œuvre.

Un système identifié pour évaluer l'apport de la méthodologie est une partie restreinte d'EGNOS (European Geostationary Navigation Overlay Service) : la modélisation du comportement du Check-Set et du Processing Set (CPF) et leur interaction avec les RIMS (Ranging and Integrity Monitoring Station) de type C afin de garantir aux utilisateurs l'intégrité du signal provenant des satellites GPS.

Le problème vient des distorsions anormales des signaux, appelés Evil Wave Forms ou EWF ([5]) qu'un satellite est susceptible d'émettre suite à une panne partielle à bord et qui pourraient avoir des conséquences dramatiques pour un avion en train d'atterrir, puisque le pilote pourrait penser qu'il est plus ou moins haut sur la piste qu'il ne l'est réellement. Pour parer à ce problème, les RIMS C ont pour but de surveiller la qualité du signal GPS. Elles transmettent alors le résultat de leurs tests internes au CPF qui décide d'utiliser ou non le signal suivant un système de vote majoritaire. Les RIMS C ont une certaine probabilité de non-détection et peuvent aussi être indisponibles suite à une panne par exemple, réduisant alors la robustesse du CPF. L'objectif est donc de modéliser le comportement du CPF et de trouver les événements redoutés, autrement dit, les cas menant à une validation par le CPF d'un signal erroné et d'en quantifier la probabilité.

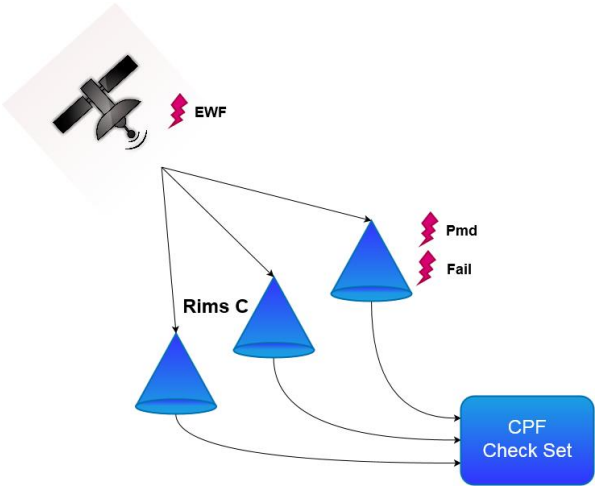

Fig. 4. Modèle retenu pour l'expérimentation.

Le comportement du satellite est simple à modéliser : il a une certaine probabilité d'émettre un EWF qu'il transmet aux RIMS. En terme AltaRica, il s'agit donc d'un flux comme vu précédemment.

En ce qui concerne la RIMS C, elle peut se trouver dans plusieurs états. Elle peut fonctionner normalement, être dans un état de « non-détection », ne pas fonctionner (défaillance) ou alors tout simplement ne pas être utilisée car elle se trouve en dehors de la zone de visibilité du satellite par exemple. 3 transitions sont donc décrites lui permettant de passer d'un état à un autre avec une certaine loi de probabilité dépendant des paramètres intrinsèques de la RIMS (MTBF, MTTR, Pmd etc.).

La figure ci-dessous présente le code AltaRica ayant permis de modéliser une RIMS C.

```
domain RimsState {Functioning, Misdetection, NonFunctioning, NOT_USED}

class RimsC
    parameter RimsState initialState = Functioning;
    RimsState State (init = initialState);
    Boolean statusBitGood (reset = true);
    Boolean monitoringSatellite (reset = true);
    Boolean satelliteSignalWithoutEWF (reset = true);
    event failure (delay = exponential(       ));
    event repair (delay = exponential(    ));
    event misdetection (delay = constant(       ));
transition
    failure: (State != NonFunctioning and State != NOT_USED) -> State :=
NonFunctioning;
    repair: State == NonFunctioning -> State := Functioning;
    misdetection: State == Functioning -> State := Misdetection;
assertion
    statusBitGood := (if (State == Functioning) then
(satelliteSignalWithoutEWF == true) else (State == Misdetection));
    monitoringSatellite := (State != NonFunctioning and State != NOT_USED);
end
```

Fig. 5. Figure 5. Code AltaRica d'une RIMS C.

Le CPF reçoit quant à lui toutes les données des RIMS et détermine si au moins l'une d'entre elle monitore le satellite. Ensuite, il calcule si plus de la moitié des RIMS C opérationnelles confirment l'intégrité du signal.

```
class CpfCS
    Boolean monitoringSignal (reset = true);
    Boolean useSignal (reset = true);
    Boolean inRims1statusBitGood, inRims2statusBitGood, inRims3statusBitGood
(reset = true);
    Boolean inRims1monitoringSatellite, inRims2monitoringSatellite,
inRims3monitoringSatellite (reset = true);

assertion
    monitoringSignal :=  (#(inRims1monitoringSatellite,
inRims2monitoringSatellite, inRims3monitoringSatellite) >= 1);
    useSignal := (#(inRims1statusBitGood, inRims2statusBitGood,
inRims3statusBitGood) > (#(inRims1monitoringSatellite,
inRims2monitoringSatellite, inRims3monitoringSatellite)/2));
end
```

Fig. 6. Code AltaRica du CPF.

Le modèle du système est finalement réalisé en :

- Instanciant le nombre désiré d'objets de chaque classe,
- Liant les entrées et sorties des objets créés
- Définissant un observable (événement redouté évoqué précédemment : le CPF valide un signal corrompu)

B. *Résultats.*

Ce que l'on souhaite évaluer dans cet exemple, c'est la probabilité d'envoi d'une information erronée en fonction du DOC (Depth of Coverage) : le nombre de RIMS C qui monitorent un Satellite donné. DOC *n* indique que *n* RIMS C reçoivent et traitent le signal du satellite. C'est la raison pour laquelle le modèle introduit l'état *NOT_USED* pour les RIMS afin de pouvoir calculer tous ces paramètres avec le même modèle. Ainsi, pour DOC 1, une seule RIMS est *Functioning* tandis que les autres sont *NOT_USED*.

Les résultats attendus sont les suivants :

Valeur DOC	Probabilité
DOC 1	α
DOC 2	β
DOC 3	γ
DOC 4 et plus	δ

Fig. 7. Probabilité d'erreur non détectée en fonction du DOC (calcul manuel)

Les valeurs obtenues en utilisant des outils basés sur la méthodologie MBSA sont :

Valeur DOC	Probabilité
DOC 1	α
DOC 2	β
DOC 3	γ
DOC 4	δ - ζ
DOC 5	ε

Fig. 8. Probabilité d'erreur non détectée en fonction du DOC (calcul par outil MBSA)

Celles-ci sont proches de celles attendues, confirmant ainsi que le modèle est correct.

Parallèlement, il a été possible de réaliser des simulations stochastiques avec des courbes type P(t) ou D(t) sur des systèmes simples, donnant des résultats satisfaisants.

Il est intéressant, dans un contexte « Safety » de disposer de l'arbre de défaillance. Une fois le modèle créé, il a été relativement simple d'instancier de nouvelles RIMS C, de générer l'arbre de défaillance de chaque DOC et enfin de calculer les probabilités pour plusieurs niveaux DOC (pour le DOC 5 par exemple), chose qui n'est pas facile à faire à la main.

La figure ci-dessous représente l'arbre généré en DOC 1 (le plus simple) avec le code présenté précédemment, c'est-à-dire que toutes les autres RIMS sont *NOT_USED*.

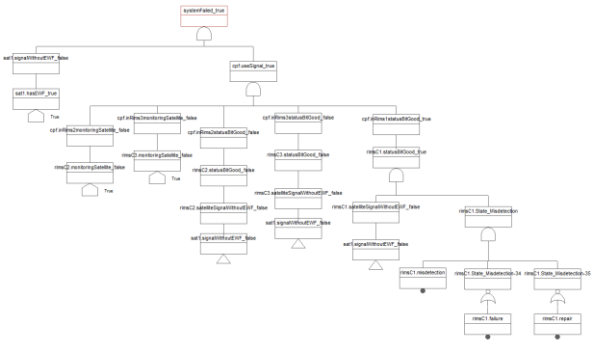

Fig. 9. Arbre de défaillance généré en appliquant la méthodologie MBSA au modèle de RIMS C en DOC 1 (plusieurs RIMS sont dans l'état NOT_used).

Il est possible de simplifier l'arbre précédent en n'instanciant qu'une seule RIMS C, de manière à éviter d'avoir des événements impliquant des RIMS non utilisées. Il faut pour cela réaliser un modèle dédié du Système en "DOC 1" (voir également le dernier chapître sur la lisibilité des arbres).

L'exemple développé dans ce chapitre démontre donc que s'ils sont bien utilisés, les outils basés sur la méthodologie MBSA sont capables de réaliser des analyses complexes différemment des analyses traditionnelles, apportant de ce fait un point de vue intéressant. Par ailleurs dans cet exemple les arbres des défaillances pour les différents DOC (de 1 à n) peuvent être obtenus en quelques minutes une fois le modèle initial implémenté puisqu'il suffit alors de changer l'état initial des RIMS pour passer d'une configuration DOC à une autre.

IV. Conclusion sur l'applicabilite de la methodologie MBSA au domaine spatial

En conclusion, la méthodologie MBSA laisse entrevoir de belles promesses et l'enthousiasme qu'elle suscite dans le monde de la recherche est compréhensible. Toutefois son application industrielle est difficile à mettre en œuvre.

Certes les outils basés sur cette méthodologie présentent des avantages tels que la communication renforcée avec l'ingénierie Système par l'utilisation d'un modèle commun. Par ailleurs et comme nous l'avons vu dans cet article, dans certains cas bien précis elle permet à l'ingénieur RAMS de dépasser sa fonction, venant en aide à l'architecte Système.

Toutefois, les expérimentations menées conjointement entre Thales Alenia Space et Ligeron, ont révélé un certain nombre de désavantages :

- Un seul cas a été identifié pour lequel l'approche MBSA (basée donc sur AltaRica) semble supérieure aux approches traditionnelles. Dans cet exemple, elle n'a pas permis de gagner du temps mais elle permettrait de réaliser des analyses plus complexes, tels que des trade-offs sur l'architecture ou sur les algorithmes de vote du CPF.
- Dans la plupart des cas étudiés lors des expérimentations, les analyses traditionnelles sont plus efficaces.
- L'Ingénieur en Sûreté de Fonctionnement doit consentir beaucoup d'efforts pour s'approprier la méthodologie MBSA, qui fait appel à des notions abstraites et à la programmation avec un langage formel. C'est donc une complexité supplémentaire qui se dresse devant lui, il n'est pas certain que l'industriel, très contraint par les délais (quelques heures pour réaliser une analyse) puisse prendre le temps de se documenter sur la méthode et les bonnes pratiques de la programmation en langage formel.
- La méthodologie MBSA pose un problème de détermination de la granularité des analyses lorsqu'il s'agit de connecter des modèles entre eux. Un modèle d'une partie d'EGNOS a été réalisé, mais s'il avait fallu représenter l'intégralité du Système, il aurait été nécessaire que les personnes en charge du codage se coordonnent entre elles régulièrement pour veiller à avoir le même niveau d'abstraction ou par exemple à ne pas avoir de conflit entre leurs variables.
- La modélisation sous AltaRica présente des limites pour la modélisation de certaines architectures redondées. Ce point est développé par la suite.

Il n'y a finalement pas de reproches à la méthodologie en elle-même. En revanche les outils qui permettent de l'utiliser ne semblent pas assez matures pour en tirer pleinement parti et être déployés industriellement à court terme sur de grands projets spatiaux.

Finalement, cette méthodologie souffre de la publicité abondante qui en est faite et les outils qui permettent de l'utiliser ont besoin de temps pour arriver à maturité. Vouloir imposer son utilisation à grande échelle pour effectuer des analyses de Sécurité pourrait conduire à un échec dans l'état actuel des choses.

En attendant de disposer d'outils plus évolués il pourrait être intéressant de revenir à une approche classique et à des outils maîtrisés comme par exemple les réseaux de Petri ou les chaines de Markov. Finalement ces techniques éprouvées sont également une approche basée sur les modèles et permettent, elles aussi, de quantifier un risque pour la sécurité du Système. De manière souvent bien plus directe qu'en passant par la programmation en AltaRica.

V. Suggestions d'ameliorations pour les outils bases sur le MBSA

Cette dernière partie suggère des points d'améliorations pour les outils basés sur le langage AltaRica, après six mois d'utilisation journalière.

A. Lisibilité des arbres de défaillance.

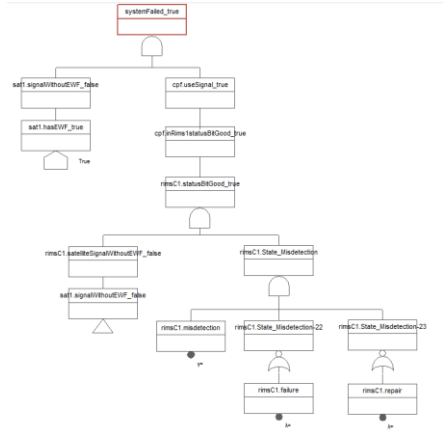

Fig. 10. Arbre de défaillance généré par les outils MBSA pour le modèle de RIMS C en DOC 1 (en instanciant uniquement une RIMS C).

Il existe une infinité de manières de représenter un arbre de défaillance. L'ingénieur RAMS doit expliciter les arbres de manière logique, compréhensible et donc vérifiable, dans la mesure du possible. Or les arbres générés automatiquement avec les outils AltaRica sont peu lisibles et par là même difficiles à valider. A titre d'exemple les deux figures ci-dessous montrent deux arbres de défaillance qui devraient théoriquement être identiques, l'un généré par les outils basés sur le MBSA et l'autre par un être humain, pour le même événement redouté du même Système.

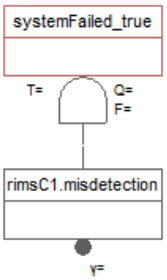

Fig. 11. Arbre de défaillance généré par un humain pour le modèle de RIMS C en DOC1.

La différence est frappante et on voit donc qu'une étape de traitement est nécessaire avant de pouvoir exploiter des arbres issus de la programmation en AltaRica. Avec des arbres plus complexes la logique de représentation est perdue.

B. Modélisation des boucles.

Les boucles de monitoring sont très utilisées sur la plupart des systèmes critiques. Il n'est à priori pas possible de les modéliser en AltaRica 2.0 mais l'AltaRica 3.0 le permettrait ([1], [4])

C. Modélisation d'une redondance froide.

Les redondances froides (un composant secours, hors tension, en parallèle du composant nominal) sont très utilisées dans le domaine spatial, pour des raisons de coûts et de consommation électrique. Or les modèles AltaRica sont des systèmes qui passent d'un état à un autre uniquement par des transitions. Comme explicité par [2] il n'y donc pas d'évolution continue possible entre deux événements et il est nécessaire de modéliser les aspects temporels par des transitions discrètes.

Par exemple, des obstacles ont été rencontrés pour la modélisation de redondances froides : L'initialisation de la simulation est réalisée sur l'état qui conduit à l'alimentation du composant nominal, ce qui induit que l'état menant à l'activation du composant secours est initialement désactivé. Lors de la génération de coupes minimales, déduites des différents chemins menant à un évènement redouté, l'algorithme ne prend donc pas en compte le fait que la défaillance du circuit secours engendrera l'apparition de l'évènement redouté. Ceci peut s'expliquer par le fait que l'outil de recherche des coupes minimales ne prend pas en compte le changement d'états nominaux des composants.

Il y a donc deux possibilités pour modéliser les redondances froides :

Soit les ignorer ou les simplifier fortement, ce qui fausse les résultats, et ne permet pas d'évaluer, par exemple une performance de disponibilité opérationnelle ou d'optimiser un stock de rechanges.

Soit les prendre en compte en ajoutant des composants fictifs dans le système afin de parvenir au résultat escompté : les calculs deviennent justes mathématiquement mais les arbres de défaillance ne sont plus corrects car ils font apparaître des éléments fictifs qui n'existent pas dans le système.

D'où la nécessité de créer plusieurs modèles et de les maintenir, l'un avec des éléments fictifs pour les calculs et l'autre uniquement pour générer les arbres. On s'éloigne des objectifs des outils basés sur la méthodologie MBSA qui devaient simplifier le travail de l'ingénieur en Sûreté de Fonctionnement :

- D'autre outils et méthodes tels que les réseaux de Petri ou les chaines Markov prennent facilement en compte ces aspects transitoires. Il est donc dommage de devoir créer un modèle par outil.
- Il est difficile moralement d'accepter de modifier le système tel qu'il existe réellement dans le but de parvenir au bon résultat. De plus cela entraine un risque d'erreur important pour les personnes reprenant le modèle et accroît la complexité.
- Ces composants fictifs se retrouvent aussi dans les arbres générés ce qui alourdit encore ces derniers.

D. Détermination du comportement du système par l'AMDEC.

Le fait de renseigner les répercussions des pannes simples issues de l'AMDEC du Système dans un outil basé sur la méthodologie MBSA, ne permet pas de générer les arbres de défaillance. Ce point devrait être clarifié dans la communication qui est faite sur les outils basés sur le MBSA.

VI. Remerciements

Les auteurs remercient Ahlem MIFDAOUI de l'Institut Supérieur de l'Aéronautique et de l'Espace pour son support pédagogique.

Ainsi que Rémi COLLE de Ligeron pour les tests réalisés avec les outils MBSA basés sur l'AltaRica.

VII. REFERENCES

[1] Batteux, Michel, Tatiana Prosvirnova, Antoine Rauzy, et Leïla Kloul. «The altarica 3.0 project for model-based safety assessment.» IFAC Proceedings Volumes, vol 46. Vol. 46. Elsevier, 2013. 127-132.

[2] De Brito, Gabriella, et Marion Morel. «Approche basée "modèle" pour l'analyse safety des systèmes avioniques critiques et des erreurs humaines.» Lambda Mu 20, 2016.

[3] Point, Gérald. «AltaRica : Contribution à l'unification des méthodes formelles et de la sûreté de fonctionnement.» PhD thesis. Université Bordeaux I, Jan 2000.

[4] Prosvirnova, Tatiana. «AltaRica 3.0 : une approche orientée modèles pour la Sûreté de Fonctionnement.» PhD thesis. Ecole Polytechnique, Nov 2014.

[5] R. Eric Phelts, Dennis M. Akos, Per Enge. « Robust Signal Quality Monitoring and Detection of Evil Waveforms », ION Proceedings 2000.

.