



System-Analyst : nouveau moteur d'évaluations de modèles MBSA dynamiques de grandes tailles

System-Analyst : new engine for evaluating large dynamic models in the MBSA context

CLEMENT Emmanuel
Thales
Brest

emmanuel.clement@fr.thalesgroup.com

MILCENT Frédéric
Naval Group
Nantes

frederic.milcent@naval-group.com

LE BERRE Nicolas
Thales
Brest

nicolas.leberre@fr.thalesgroup.com

1 **Résumé** — Le déploiement de l'ingénierie dirigée par les modèles pour la réalisation d'évaluation de sûreté de fonctionnement (MBSA)
2 amène à devoir réaliser des modèles dynamiques de grande taille, fortement connexe et comprenant des boucles fermées (représentatives de
3 boucles d'asservissement de nos systèmes industriels). Pour ces modèles de grandes tailles, les moteurs d'évaluation existants ont des
4 difficultés à obtenir les performances de Sûreté de Fonctionnement dans des temps raisonnables. Pour répondre à cela, nous avons initié des
5 travaux pour la conception et le développement de nouveaux moteurs de simulations qui sont librement distribués avec l'outil MBSA «
6 System-Analyst ». Cette communication vise à présenter succinctement les concepts sous-jacents, les types d'évaluations disponibles
7 aujourd'hui et les travaux prévus par la suite sur ces nouveaux moteurs d'évaluations. Nous présenterons deux applications réelles
8 industrielles l'une en lien avec les activités de Naval Group et l'autre en lien avec les activités de Thales DMS. Ces deux applications nous
9 permettront de comparer d'une part les résultats obtenus et, d'autre part, le temps nécessaire pour obtenir ces résultats, avec deux outils
10 MBSA disponibles sur le marché : SIMFIA NEO et AltaRica Wizard de l'association OpenAltaRica.

11 **Mots-clefs** — *simulation, évaluations dynamiques, MBSA, modélisation*

12 **Abstract** — The deployment of model-driven engineering for the realization of RAMST assessment requires the creation of large-scale,
13 highly interconnected dynamic models that include closed loops (representative of feedback loops in our industrial systems). These large-
14 scale models present challenges for existing evaluation engines in terms of efficiently obtaining the RAMST performance results within
15 reasonable periods. To address this, we have initiated work on the design and development of new simulation engines. These new evaluation
16 engines aim to handle large-scale, highly interconnected, and looped models. They are freely distributed with the MBSA (System-Analyst)
17 tool. This communication aims to briefly present the underlying concepts, the types of evaluations currently available, and the planned future
18 work. We will display two real industrial applications, one related to Naval Group's activities and the other related to Thales DMS's activities.
19 These two applications will allow us to compare the results obtained and the time required to obtain these results using two MBSA tools:
20 SIMFIA NEO and OpenAltaRica.

21 **Keywords** — *simulation, évaluations dynamiques, MBSA, modélisation*

22 I. INTRODUCTION

23 La complexité croissante des systèmes entraîne, notamment en sûreté de fonctionnement, la nécessité de progresser dans les
24 méthodes d'analyses en développant l'ingénierie dirigée par les modèles (MBSA). Le MBSA (Model-Based Safety Assessment)
25 permet, par la réalisation de modèles de plus haut niveau, c'est-à-dire proches des architectures fonctionnelles et physiques des
26 systèmes, d'assurer le partage et la cohérence des hypothèses, des données d'entrée et des résultats des analyses entre les
27 différents acteurs de la Sûreté de fonctionnement. Cela conduit à une cohérence, à une validation et à une capitalisation des
28 modèles et des analyses. C'est dans ce cadre qu'intervient cette communication.

29 Cette démarche s'inscrit dans une démarche plus globale d'organisation des travaux FMDTS (Clement E. et al., 2022)
30 (Fiabilité, Maintenabilité, Disponibilité, Testabilité et Sécurité) autour d'un modèle MBSA unique, validé et partagé tel que
31 présenté dans la Figure 1.

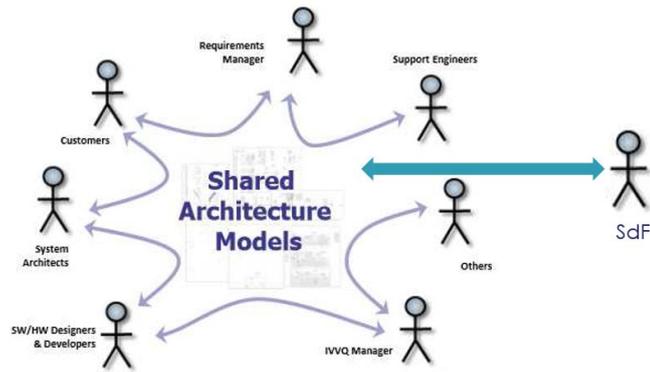


Figure 1: Architecture partagée

II. MÉTHODOLOGIE

A. Contexte

Thales DMS et Naval Group travaillent activement au déploiement de l'ingénierie dirigée par les modèles dans le cadre des activités de sûreté de fonctionnement. Ce déploiement vise d'une part à partager un point de vue unique sur les architectures de nos systèmes entre les ingénieurs de la sûreté de fonctionnement et les autres ingénieries de spécialités dont notamment l'ingénierie système. D'autre part, l'adoption de modèles pour la conduite des activités de sûreté de fonctionnement permet de formaliser, de capitaliser et d'améliorer la confiance dans les études menées. De plus, la donnée présente dans ces modèles est ensuite valorisée pour notamment construire des services à forte valeur ajoutée pour nos clients.

Cependant, nous sommes confrontés à deux limites importantes. D'une part, les systèmes industriels, que nous développons, comprennent des fonctions fortement asservies. Or, la modélisation de ces fonctions asservies nécessite de réaliser des modèles avec des boucles fermées. Dans certains cas, notamment à Thales, nous nous retrouvons avec des modèles qui sont fortement bouclés. Beaucoup d'outils logiciels MBSA disponibles sur le marché (basé sur les automates de mode ou les systèmes à transitions gardées) ne permettent pas d'évaluer des modèles bouclés sans devoir mettre en place des artefacts de modélisation (Prosvirnova T. et al., 2022).

Par ailleurs, la deuxième limite à laquelle nous sommes confrontés est que les outils de simulation employés dans les logiciels MBSA ne permettent pas de réaliser certaines évaluations dans des temps raisonnables sur des modèles de taille industrielle. Les évaluations réalisées pour obtenir des résultats de sûreté de fonctionnement notamment la génération de séquences minimales critiques nécessite l'utilisation d'algorithmes de complexité à minima exponentielle (en fonction de l'ordre maximal des séquences attendues).

Afin d'accompagner le déploiement des outils MBSA dans nos entités industrielles, nous avons démarré des travaux qui visent à mitiger les deux limites présentées plus haut. Les travaux consistent en la proposition:

1. D'une première implémentation optimisée pour les CPU (processeurs de nos ordinateurs) objet de cette communication;
2. D'une deuxième implémentation portée sur les GPU (cartes graphiques de nos ordinateurs). L'architecture des GPU ouvre la possibilité de paralléliser de façon importante les évaluations et ainsi doit permettre d'obtenir des gains significatifs de performance comme décrit dans l'article (Buzer L. et al., 2023).
3. D'une troisième implémentation portée sur des FPGA (Circuit logique programmable). L'architecture spécifique des FPGA ouvre la voie pour obtenir des gains importants tels que présentés dans l'article (Barbone, M. et al., 2021).

La figure 2 ci-après représente la roadmap proposée et les gains espérés par chacune des implémentations.

Les travaux présentés ici présentent un état des lieux des résultats obtenus avec la première implémentation. Ces travaux ne portent pas sur l'amélioration des algorithmes dans le but de réduire leur complexité mais portent sur l'optimisation de leur implémentation en code informatique.

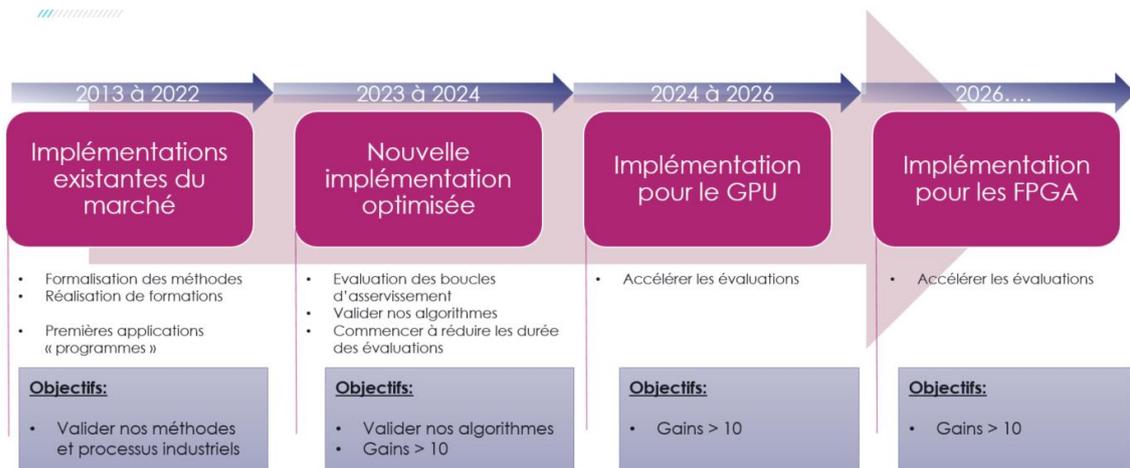


Figure 2 : Roadmap travaux implémentations de moteurs d'évaluations

B. Principes retenus

1) Objectifs

Les objectifs visés par ces travaux sont les suivants :

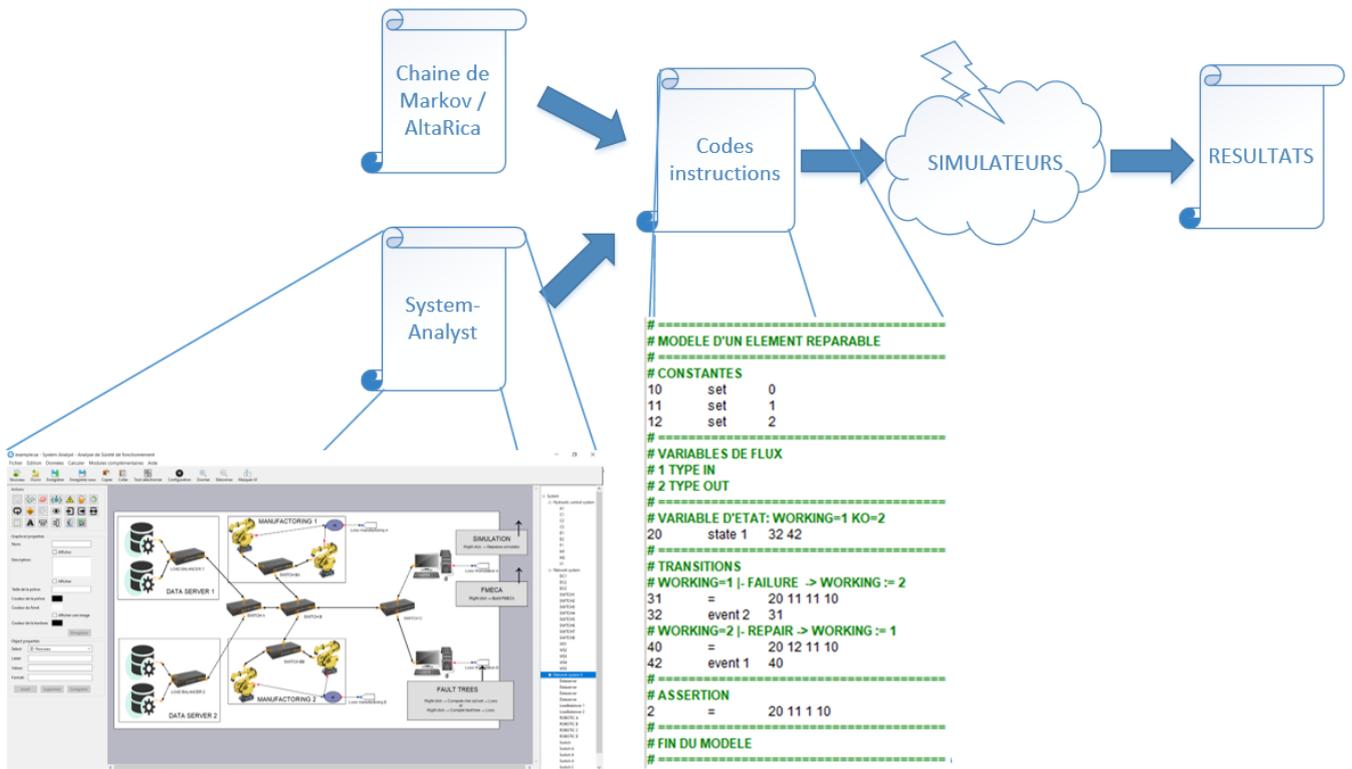
1. Permettre l'évaluation de modèles dont la sémantique correspond aux systèmes à transition gardées ;
2. Permettre la réalisation des évaluations suivantes :
 - a. Simulations pas-à-pas ;
 - b. Simulations stochastiques ;
 - c. Génération de séquences stochastiques ;
 - d. Génération des séquences critiques exhaustives ;
 - e. Génération des séquences critiques minimales ;
 - f. Génération des coupes minimales.
3. Obtenir des résultats correspondants aux attendus et cohérents de l'état de l'art ;
4. Traiter naturellement les systèmes à boucles fermées (boucle d'asservissement) ;
5. Obtenir un gain significatif en terme de durée des évaluations et de consommation de mémoire ;
6. Permettre le déploiement sur un ordinateur personnel et sur des serveurs de calculs indépendamment de leur architecture matérielle et de leur système d'exploitation.

2) Solution logicielle

Pour permettre d'atteindre les objectifs fixés, le choix du langage de programmation de l'implémentation logicielle a été une étape importante. Le choix s'est porté sur le langage de programmation Rust. Rust est un langage de programmation compilé multi-paradigme qui met l'accent sur la performance, la sûreté des types et la concurrence. Rust est parmi les langages les plus rapides et économes en mémoire. Il est sans environnement d'exécution, ni ramasse-miettes, il peut dynamiser des services à hautes performances, s'exécuter dans des systèmes embarqués, et s'intégrer facilement à d'autres langages.

92 3) Utilisation générale de ce nouveau moteur d'évaluations

93 Le mode d'utilisation générale du moteur d'évaluations est décrit dans la figure 3 ci-après.



94

95

Figure 3: Etapes d'utilisation du moteur d'évaluations

96 Le code « d'instructions » à évaluer est construit sur la base d'une liste des variables constitutives du modèle. Chaque variable
97 de la liste comprend les paramètres suivants :

- 98 1. Identifiant de la variable ;
- 99 2. Type et paramètres de la variable ;
- 100 3. Liste des identifiants des prédécesseurs de la variable.

101 Les « instructions » d'un modèle correspondant à un élément basique « réparable » est présenté ci-après en figure 4.

102

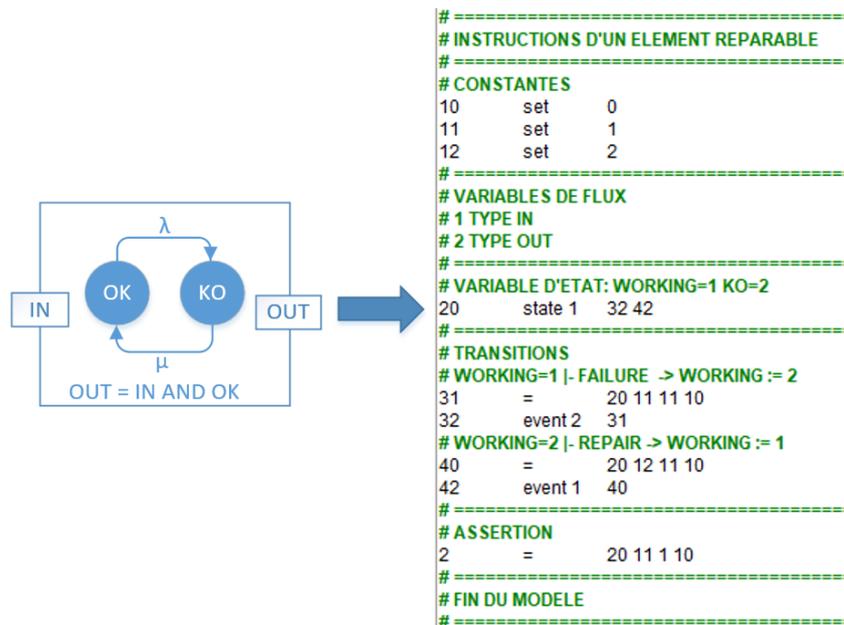


Figure 4: Représentation d'un élément réparable en « instruction » pour le simulateur

103
104

133 Ces artefacts de modélisation peuvent notamment prendre les formes suivantes :

- 134 • Ouvrir la boucle ce qui a pour inconvénient d'éloigner le modèle de l'architecture de la solution;
- 135 • Ajouter un évènement de type « Dirac » dans la boucle ce qui polluera la génération des séquences (ou coupes
- 136 minimales) avec des évènements « artificiels » (quoiqu'il est possible de les masquer dans certains logiciels) et
- 137 complique la construction et la validation du modèle.

138 Les étapes pour permettre l'évaluation de modèles à boucles fermées sont les suivantes :

- 139 • Construire une structure de données des variables basée sur un graphe dirigé cyclique ;
- 140 • Autoriser la réévaluation des variables dans une même session d'évaluation ;
- 141 • Ordonner la mise à jour des variables de telle sorte que la cohérence du modèle est assurée ;
- 142 • Ajouter des mécanismes pour borner la mise à jour des variables du modèle.

143 En implémentant ces étapes dans le moteur d'évaluations, nous arrivons à simuler des modèles fortement connexes et
144 fortement bouclée de façon naturelle et performante sans nécessiter de devoir ajouter, aux modèles, des artefacts de modélisation.

145 *D. Application aux modèles de grandes tailles*

146 Nous entendons par modèle de grandes tailles des modèles qui sont constitués d'un grand nombre de variables et/ou de
147 transitions. L'ordre de grandeur, dans nos cas industriels, s'exprime à minima en centaine pour les transitions et en millier pour
148 les variables.

149 Pour permettre des évaluations efficaces des modèles, les travaux ont porté sur 5 activités distinctes :

- 150 • Simplifier/factoriser les variables du modèle avant toute évaluation;
- 151 • Structurer efficacement en mémoire les variables ainsi que leurs données;
- 152 • Evaluer efficacement et au strict nécessaire les variables lors du tirage d'une nouvelle transition ;
- 153 • Paralléliser massivement les algorithmes d'évaluation du modèle telles que les simulations stochastiques, les
- 154 générations de séquences critiques minimales.

155 Les résultats des travaux présentés dans cette publication concernent la première version de ce nouveau moteur d'évaluations
156 de modèles dynamiques dysfonctionnels. Des travaux d'optimisation sont actuellement en cours pour proposer courant 2025 une
157 nouvelle version intégrant des optimisations supplémentaires.

158 III. APPLICATION INDUSTRIELLE N°1: MODELISATION D'UNE FLOTTE D'EQUIPEMENTS

159 *1) Présentation de l'application industrielle*

160 Naval Group conçoit et réalise des navires armés, des systèmes de combat et des infrastructures portuaires. Ce type de
161 systèmes présente les caractéristiques suivantes :

- 162 • Le niveau de disponibilité doit être important ce qui nécessite la mise en place de redondances (actives et/ou
- 163 passives) ;
- 164 • Il présente deux phases d'exploitation bien distinctes :
 - 165 ○ En mer, le système est totalement autonome ;
 - 166 ○ A quai, il dispose de tous les moyens mis à sa disposition par l'infrastructure portuaire.
- 167 • Les capacités de maintenance diffèrent selon la phase dans laquelle se trouve le système ;
- 168 • Les systèmes sont présents sur plusieurs bâtiments (notion de flotte) ;
- 169 • Le délai d'approvisionnement de certains matériels est parfois assez important ;
- 170 • Les équipes de maintenance sont limitées en nombre et partagées avec les autres systèmes présents à bord du
- 171 bâtiment.

172 Le cas d'application industrielle présenté ici est un système de surveillance (Milcent F. et al., 2014), présent sur les navires
173 développés par Naval Group. Son rôle est de récupérer des informations (au niveau du Hub1) et de les transmettre, via un réseau
174 informatique, aux opérateurs sur leurs stations de travail (Workstation1 et Workstation2). Il est considéré disponible si les
175 informations sont fournies aux opérateurs des deux stations de travail.

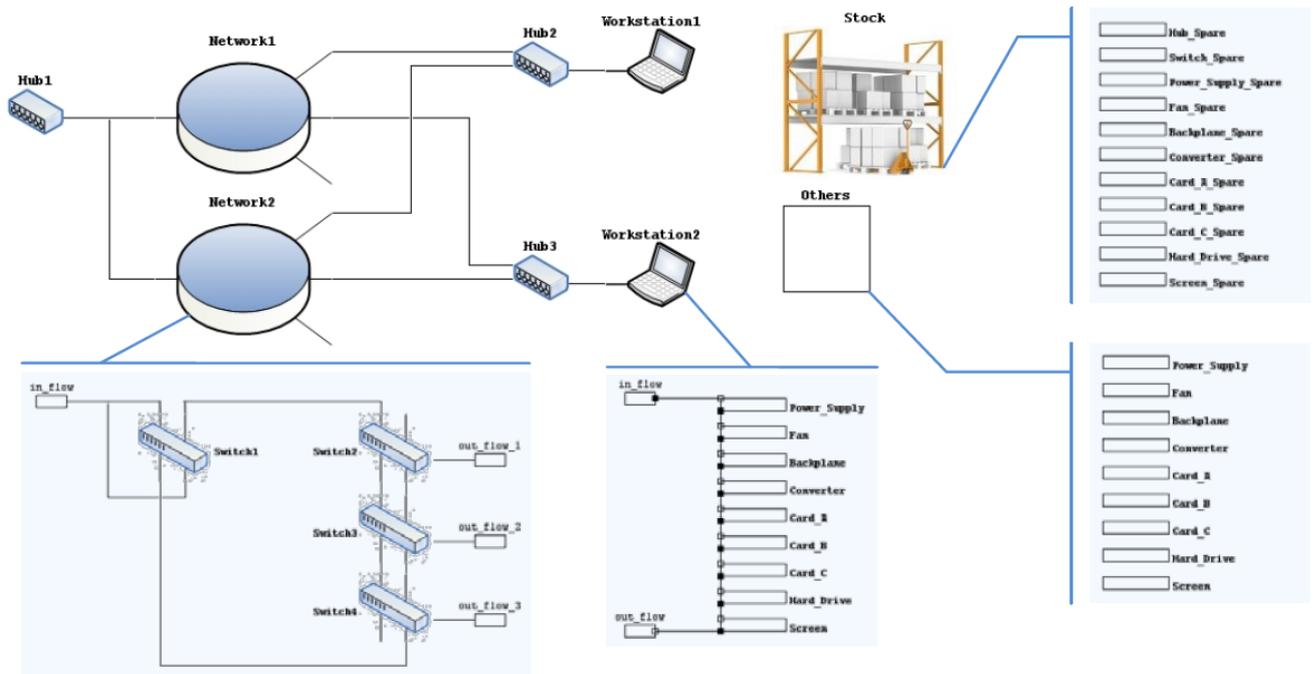


Figure 7: Système de surveillance d'un navire

Ce cas d'application industrielle est intéressant pour notre étude car il est fortement dynamique et comporte un nombre que nous estimons important de transitions et de variables. Le tableau ci-après présente quelques métriques.

TABLE I. STATISTIQUES DU MODELE

STATISTIQUES DU MODELE	
Variables	5669
Transitions	236
Observateurs	1

2) Présentation des outils MBSA

La comparaison proposée porte sur la réalisation et l'évaluation de ce cas industriel sur deux outils MBSA :

1. SIMFIA NEO (De Bossoreille X. et al., 2018) en version 1.4.0 ;
2. SYSTEM-ANALYST (Breton S. et al., 2018) en version 2.2.

Le matériel sur lequel les applications ont été exécutées est un simple ordinateur portable professionnel classique.

3) Présentation des résultats obtenus

Les analyses ont porté sur l'évaluation de la disponibilité du système d'intérêt. Pour cela, deux évaluations ont été réalisées :

- Une simulation stochastique pour évaluer la performance en terme de disponibilité;
- Une génération des séquences critiques minimales pour s'assurer de la validité du modèle.

Les résultats obtenus par les simulations stochastiques ainsi que les temps de calculs et l'utilisation de la mémoire sont présentés dans le tableau ci-après.

TABLE II. RESULTATS DES SIMULATIONS STOCHASTIQUES

SIMULATEUR STOCHASTIQUE						
	Temps de mission	Nombre histoires	Stocks	Dispo	Durée des calculs	Mémoire occupée
System-Analyst	1680	1,0E+04	Tout à 1	92,7%	1s	36Mo
	14820			39,3%	1s	36Mo
	1680	1,0E+05		93,0%	1s	36Mo
	14820			39,3%	3s	36Mo
	1680	1,0E+06		93,0%	10s	36Mo
	14820			39,3%	30s	36Mo
SIMFIA NEO	1680	1,0E+04	Tout à 1	92.8%	2s	> 100 Mo
	14820			39.5%	26s	> 100 Mo
	1680	1,0E+05		93,0%	31s	> 100 Mo
	14820			39.3%	4min	> 300 Mo
	1680	1,0E+06		92.99%	4min	> 300 Mo
	14820			39.4%	40min	> 300 Mo

194

195

196 Les résultats obtenus par les générations de séquences minimales ainsi que les temps de calculs et l'utilisation de la mémoire
197 sont présentés dans le tableau ci-après.

198

TABLE III. RESULTATS DES GENERATIONS DE SEQUENCES

GENERATEUR SEQUENCES MINIMALES						
	Ordre max	Combinaisons d'événements	Stocks	Quantité de séquences minimales	Durée des calculs	Mémoire occupée
System-Analyst	2	56 169	Tout à 1	21	<1s	36Mo
	3	13 312 053		35	<1s	36Mo
	4	3 154 956 561		47	5s	36Mo
SIMFIA NEO	2	56 169	Tout à 1	21	14s	< 100 Mo
	3	13 312 053		35	46s	< 200 Mo
	4	3 154 956 561		47	49min	> 300 Mo

199

200 4) Synthèse

201 La réalisation de ce cas industriel a permis deux choses :

- 202 • De valider les résultats obtenus par le nouveau moteur d'évaluations avec les résultats obtenus par Simfia Neo ;
- 203 • De comparer la durée et la consommation de mémoire des simulateurs stochastiques et des générateurs de séquences
204 minimales.

205 Ces résultats ne permettent évidemment pas de tirer une conclusion générale tant les spécificités des modèles à évaluer et les
206 spécificités des implémentations logicielles peuvent faire varier la mesure réalisée. Cependant et dans ce cas précis, nous
207 observons que les résultats des analyses obtenus dans les deux cas sont parfaitement similaires et conformes à l'attendu. En
208 revanche, les gains de performance obtenus sur les durées les plus longues avec le nouveau moteur d'évaluations est présenté
209 dans la table ci-après.

210

TABLE IV. PRESENTATION DES GAINS OBTENUS PAR LE NOUVEAU MOTEUR D'ÉVALUATIONS

COMPARAISON DES PERFORMANCES				
	SIMULATEUR STOCHASTIQUE		SEQUENCES MINIMALES	
	DUREE	MÉMOIRE	DUREE	MÉMOIRE
System-Analyst	30s	36Mo	6s	36Mo
Simfia Neo	40mn	>300Mo	49mn	> 300Mo
Gains	x80	x10	x490	x10

212

213 Nous observons une diminution de la consommation de mémoire de l'ordre d'un facteur 10 dans tous les cas. Pour les durées
214 des évaluations, nous observons une durée divisée par 80 dans le cadre de la simulation stochastique et d'un facteur de l'ordre
215 490 pour la génération des séquences minimales.

216

217 IV. APPLICATION INDUSTRIELLE N°2 : MODELISATION DETAILLEE D'UN SYSTEME COMPLEXE

218 1) Présentation de l'application industrielle

219 Thales accompagne les forces armées, les gouvernements et les organisations internationales pour les aider à acquérir et à
220 maintenir la sécurité, la supériorité tactique et l'indépendance stratégique quelle que soit la nature de la menace. C'est dans ce
221 cadre que s'inscrivent les activités de la division DMS de Thales. Thales DMS conçoit, produit et soutien des systèmes de mission
222 de défense dans le domaine marin, terrestre, aéroporté et spatial.

223 Le système industriel retenu pour ce cas d'application est un système de guerre électronique d'un avion d'armes. Ce système
224 est constitué de 150 URL (Unité Remplaçable en Ligne). Il est constitué de près de 100 000 composants électroniques et ne
225 comporte aucun COTS (Composants achetés sur étagère).

226 Ce cas d'application industrielle est intéressant pour notre étude car il comporte un nombre que nous estimons important de
227 transitions et de variables. Le tableau ci-après présente quelques métriques.

228

TABLE V. STATISTIQUES DU MODELE

STATISTIQUES DU MODELE	
Variables	47 692
Transitions	12 873
Observateurs	511

229

230 L'objectif ici est de générer une AMDEC et de quantifier les performances de la fonction de maintenance intégrée en terme
231 de taux de détection des pannes et de localisation des pannes. La méthode employée, à partir d'un modèle « MBSA » est décrite
232 dans les communications (Clement E. et al., 2018), (Rachid S. et al., 2020) et (Clement E. et al., 2022).

233 Pour mener à bien cette analyse, l'évaluation du modèle est réalisée par l'intermédiaire du simulateur pas-à-pas. Ainsi, les
234 valeurs d'un ensemble d'observateurs sont mesurés suite au tirage de chacune des transitions du modèle.

235 De plus, certaines pannes complexes du système nécessitent, pour un bon diagnostic, d'évaluer l'ordre de séquençement des
236 modes de défaillances. Pour cela, une génération des séquences à l'ordre 2 est réalisée.

237 2) Présentation des outils MBSA

238 La comparaison proposée porte sur la réalisation et l'évaluation de ce cas industriel sur deux outils MBSA :

- 239 1. OpenAltaRica en version 1.4.1;
- 240 2. System-Analyst en version 2.2.

241 Le matériel sur lequel les applications ont été exécutées est un simple ordinateur portable professionnel classique.

242 3) Présentation des résultats obtenus

243 Les analyses ont porté sur l'évaluation de la performance de la maintenance intégrée du système d'intérêt en terme de taux
244 de détections et de taux de localisation. Pour cela, deux évaluations ont été réalisées :

- 245 • Une simulation pas-à-pas pour évaluer la performance en terme de testabilité;
- 246 • Une génération des séquences à l'ordre 2 pour analyser les effets d'un séquençement dans le temps de pannes
247 complexes.

248 Les résultats obtenus ainsi que les temps de calculs et l'utilisation de la mémoire sont présentés dans le tableau ci-après.

GENERATION AMDEC/TESTABILITE					
	SIMULATION PAS-A-PAS		GENERATION SEQUENCES		
	DUREE	MÉMOIRE	QUANTITE	DUREE	MÉMOIRE
System-Analyst	2mn	25Mo	2489	136s	25Mo
OpenAltaRica	1h14mn	>10Go	2489	>24h	>1,5Go
Gains	x37	x400		> x635	x60

250

251 4) Synthèse

252 La réalisation de ce cas industriel a permis deux choses :

- 253 • De valider les résultats obtenus par le nouveau moteur d'évaluations avec les résultats obtenus par l'outil
- 254 OpenAltaRica Wizard ;
- 255 • De comparer la durée et la consommation de mémoire des simulateurs pas-à-pas et des générateurs de séquences
- 256 minimales des deux outils.

257 Comme le premier cas industriel, les résultats ne permettent pas de tirer une conclusion générale tant les spécificités des
 258 modèles à évaluer et les spécificités des implémentations logicielles des outils peuvent faire varier la mesure réalisée. Cependant,
 259 nous observons ici que les résultats des analyses obtenus avec les deux outils sont parfaitement similaires et conformes à
 260 l'attendu.

261 Nous observons une diminution de la consommation de mémoire de l'ordre d'un facteur supérieur à 60 dans tous les cas.
 262 Pour les durées des évaluations, nous observons une durée divisée par 37 dans le cas de la simulation pas-à-pas et d'un facteur
 263 de l'ordre 635 pour la génération de séquences critiques.

264 Les outils d'évaluations de l'association AltaRica sont en développement actif. Les prochaines versions vont améliorer leur
 265 performance de façon très significative notamment pour le générateur de séquences dont la version actuelle est la toute première
 266 version et vise, en premier lieu, à valider son fonctionnement. Les résultats présentés ici ne seront donc plus représentatifs avec
 267 les prochaines versions des outils OpenAltaRica.

268

269 V. CONCLUSION ET PERSPECTIVES

270 L'investissement dans le déploiement de l'ingénierie dirigée par les modèles pour les analyses des risques devient d'autant
 271 plus intéressant à mesure que le système industriel à analyser est complexe et/ou important. Or, la durée des évaluations entraîne
 272 des limites notamment sur la taille des modèles. Ces limites sont un frein au déploiement du MBSA dans nos entreprises.

273 Les travaux entrepris sur le développement de ce nouveau moteur d'évaluations de modèles dynamiques dysfonctionnels
 274 orientés « performance » démontrent qu'il est possible de mitiger la durée des évaluations de modèles de grandes tailles malgré
 275 la complexité de type « exponentielle » des algorithmes sous-jacents.

276 Les résultats présentés dans cette communication sont évidemment à prendre avec des pincettes tant les spécificités des
 277 modèles à évaluer et les spécificités des implémentations logicielles peuvent faire varier la mesure réalisée. Cependant et dans
 278 les deux cas industriels présentés ici, nous observons, dans la version actuelle de ce nouveau moteur d'évaluations de modèles
 279 MBSA, une accélération par un facteur 80 des simulations stochastiques et par un facteur 588 des générations de séquences
 280 critiques minimales.

281 Ce moteur d'évaluations MBSA est librement diffusé avec l'application System-Analyst (Breton S. et al., 2018). System-
 282 Analyst est un environnement de modélisations et de simulations MBSA libre de diffusion et libre d'utilisation.

283 Les travaux de développement de ce moteur d'évaluations MBSA vont se poursuivre sur les prochaines années tels que
 284 représentés dans la figure 2. Ces travaux visent à explorer des pistes pour améliorer l'efficacité des outils d'évaluations dans le
 285 cadre de nos applications industrielles. Dans un cadre plus général, il est bien évidemment recommandé d'utiliser des outils
 286 d'évaluations du marché reconnus, validés et qualifiés par l'usage tels que les outils de la plateforme OpenAltaRica.

287 Par ailleurs, les travaux présentés ici seront partagés avec l'association AltaRica et les résultats de l'ensemble des concepts
 288 explorés ici seront mis à la disposition des membres de l'association AltaRica.

289 Finalement, la directive CSRD (Corporate Sustainability Reporting Directive) est une directive européenne qui impose aux
 290 entreprises concernées de publier un rapport de durabilité annuel contenant des informations sur leurs performances
 291 environnementales, sociales et de gouvernance (ESG). Parmi ces informations figurent les émissions de gaz à effet de serre et la
 292 consommation d'énergie. Or, le poids du système d'information des entreprises dans leur bilan ESG est de plus en plus important.
 293 L'étude (Pereira R. et al., 2021) montre l'impact important que peut avoir le choix du langage de programmation ainsi que
 294 l'efficacité de l'implémentation dans la consommation énergétique d'un système d'information. A l'échelle de la communauté
 295 de la maîtrise des risques, bien sûr que le gain d'une amélioration de l'efficacité des logiciels, sur le bilan ESG d'un industriel,

296 ne sera pas dimensionnant. Il s'agit cependant d'une démarche qui doit être adoptée par tous les acteurs d'un système
297 d'information dont bien évidemment les éditeurs et les utilisateurs d'applicatifs « métier » de la maîtrise des risques.

298

299

REFERENCES

300 Breton, S., Le-Com, P., Thomas, T., Clement, E., (2018). System-Analyst – un outil pour l'analyse des risques, libre de
301 diffusion et compatible avec Arbre-Analyst et Open-AltaRica, Congrès Lambda Mu 21, Paris

302 Buzer, L., Cazenave, T., (2023). GPU for Monte Carlo Search, LION 17, Nice

303 Barbone, M., Howard, A, Tapper, A., Chen, D., Novak, M., Luk, W., (2021). Demonstration of FPGA Acceleration of Monte,
304 Carlo Simulation 20th International Workshop on Advanced Computing and Analysis Techniques in Physics Research, South
305 Korea

306 Clement, E., Riera, D., Batteux, M., (2018). Analyse des performances de diagnostic d'un système au moyen d'une
307 modélisation AltaRica 3.0, Congrès Lambda Mu 21, Paris

308 Rachid, S., Clement, E., Le Berre, N., Batteux, M., (2020). Processus de réalisation d'AMDEC basé sur les outils Open-
309 AltaRica, Congrès Lambda Mu 22, Paris

310 Clement, E., Le Berre, N., Cardine, A., (2022). Algorithme embarqué pour la localisation de pannes à partir d'un jumeau
311 numérique, Congrès Lambda Mu 23, Paris

312 Prosvirnova, T., Seguin, C., Frazza, C. et al., (2022). Stratégies de modélisation AltaRica de la propagation de défaillances
313 dans les systèmes dynamiques, Congrès Lambda Mu 23, Paris

314 Milcent, F., Prosvirnova, T., (2014). Optimisation de la maintenance d'une flotte de matériels basée sur une modélisation
315 dynamique, Congrès Lambda Mu 19, Dijon

316 De Bossoreille, X., Machin, M., Sagaspe, L., (2018). Un nouvel outil de safety pour maîtriser la complexité des systèmes,
317 Congrès Lambda Mu 21, Paris

318 Pereira R., Couto M., Ribeiro F., et al., (2021). Ranking Programming Languages by Energy Efficiency, Portugal

319 Site Internet de l'outil System-Analyst : www.system-analyst.fr

320 Site Internet de l'outil OpenAltaRica : www.altarica-association.org

321 Site Internet de l'outil SimFia Neo : www.protect.airbus.com/fr/safety/simfianeoo

322