

Allocations de Fiabilité et de Sécurité par facteurs de faisabilité avec le logiciel Arbre-Analyst

Reliability & Safety allocations by feasibility factors with Arbre-Analyst software

MILCENT Frédéric
Naval Group
Nantes-Indret
frederic.milcent@naval-group.com

MAIRET Alexandre
Arts et Métiers ParisTech
Paris
alexandre.mairet@ensam.eu

CLEMENT Emmanuel
Thalès
Brest
emmanuel.clement@fr.thalesgroup.com

Résumé — Naval Group a développé un module complémentaire, pour le logiciel Arbre-Analyst, permettant de réaliser des allocations, globales ou partielles, à partir d'un objectif de haut niveau (navire) en répartissant les valeurs selon certaines caractéristiques des événements concernés. Ce module est capable d'allouer des probabilités ou des lambdas en fonction d'objectifs de probabilité de défaillances, de nombre de pannes ou de défaiabilité. Les résultats sont obtenus rapidement et sont en tous points conformes à ceux obtenus avec une résolution théorique.

Mots-clefs — *allocations, arbres de défaillances, poids, fiabilité, probabilité*

Abstract — Naval Group has developed an additional module for the Arbre-Analyst software, allowing global or partial allocations, based on a high-level objective (ship) by distributing the values according to certain characteristics of the concerned events. This module is capable of allocating probabilities or lambdas according to failure probability, number of failures or unreliability objectives. The results are obtained quickly and are in all respects consistent with those obtained with theoretical resolution.

Keywords — *allocations, fault trees, weight, reliability, probability*

I. OBJECTIF ET CONTEXTE

Lors des phases amont du développement d'un système, il est nécessaire de spécifier des objectifs de fiabilité et/ou de sécurité à des sous-systèmes ou à des équipements. Fixer ces objectifs s'avère parfois complexe notamment lorsque les interactions entre les différents sous-systèmes impliquent une combinatoire importante. Dans ces cas, il est donc nécessaire d'utiliser un logiciel qui permet d'effectuer automatiquement ces allocations sur la base d'une décomposition logique du scénario menant à un événement redouté ou à une indisponibilité du système (Arbres de Défaillances, Diagrammes de Fiabilité...).

Certains logiciels permettant de réaliser et d'exploiter des Arbres de Défaillances ou des Diagrammes de Fiabilité présentent des fonctionnalités d'allocations de fiabilité (taux de défaillances) ou de probabilité. Bien souvent ceux-ci proposent un principe d'équirépartition de l'allocation ne permettant pas de moduler les valeurs en fonction des caractéristiques de chaque événement (volume de matériel concerné, taux d'utilisation, réparabilité...) et ne permettent pas toujours de figer certaines valeurs (ce qui est nécessaire avec les éléments pour lesquels les performances de fiabilité ou de sécurité sont connues).

Naval Group souhaite réaliser, à partir d'arbres de défaillances, des allocations, globales ou partielles, issues d'un objectif de haut niveau (navire), en répartissant les valeurs selon certaines caractéristiques des événements concernés. N'ayant pas la possibilité de mettre en œuvre des allocations pondérées avec les logiciels disponibles dans son environnement informatique, Naval Group a fait le choix de développer un module complémentaire pour le logiciel Arbre-Analyst (Clément E. et al., 2022).

II. METHODOLOGIE

Le logiciel Arbre-Analyst utilise le format d'échange Open PSA (Epstein et al., 2008) pour la modélisation des arbres de défaillances. Ce format d'échange décompose le modèle en 5 niveaux dont notamment le niveau « stochastique » où sont définis les paramètres des lois et leur distribution, le niveau « arbre de défaillance » où la structure logique des arbres de défaillances est définie et le niveau « méta-logique » où sont définis notamment les causes communes de défaillances. Pour réaliser des

35 allocations, il est nécessaire d'intervenir au niveau stochastique pour faire varier les paramètres tout en conservant la logique
36 booléenne et les causes communes. Réaliser les allocations à partir des paramètres signifie qu'il n'est pas possible de dissocier
37 le traitement de plusieurs événements utilisant le même paramètre mais garanti, en revanche, un traitement cohérent
38 d'événements se rapportant à des matériels identiques.

39
40 Le logiciel Arbre-Analyst propose différents types de paramètres (probabilité, taux de défaillance / lambda, fréquence, taux de
41 réparation mu...) mais le module d'allocation développé a été limité à deux types de paramètres : probabilités et lambdas. Les
42 objectifs visés peuvent varier selon les exigences ; il peut s'agir d'objectifs de probabilités de défaillance (Q et Qavg), de
43 nombre d'occurrences (E) ou de défiabilité (F). Le module permet donc de fixer ces différents types d'objectifs.

44
45 La méthode retenue est une méthode d'allocation par facteurs de faisabilité (Rauzy A. et al., 2022). Cette méthode repose sur
46 une répartition selon des poids propres aux différents paramètres. Ces poids permettent de prendre en compte les
47 caractéristiques des événements (nombre de composants concernés, possibilité de maintenance, taux d'utilisation, niveau de
48 fiabilité accessible, coût, maturité ...). Le choix a été fait de ne retenir qu'une seule valeur pour le poids de chaque paramètre,
49 cette valeur peut être issue de plusieurs caractéristiques mais la consolidation sur une unique valeur devra être réalisée hors du
50 logiciel Arbre-Analyst.

51
52 Pour permettre au module de déterminer les paramètres à allouer ainsi que leurs poids, le champ de description associé aux
53 paramètres dans Arbre-Analyst intègre un marqueur spécifique (en l'absence de ce marqueur, le paramètre ne fera pas l'objet
54 d'allocation et sera donc fixe lors des calculs). La syntaxe à utiliser est la suivante :

55
56 #Alloc ← marqueur initial
57 1 ← poids (valeur numérique)
58 Alloc_End# ← marqueur final
59

60 La corrélation entre probabilité et lambda étant très variable (notamment par rapport au temps), la définition d'une référence
61 commune pour l'affectation des poids est complexe. De ce fait, le module ne propose pas d'allocation simultanée sur les deux
62 types de paramètres, il faudra donc sélectionner probabilités ou lambdas lors du lancement des calculs.

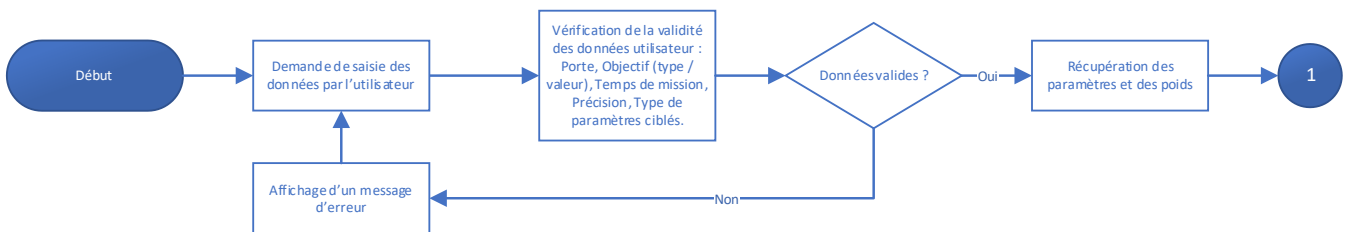
63 III. PRINCIPE DE FONCTIONNEMENT DU MODULE

64 Au lancement, le module invite l'utilisateur à sélectionner et renseigner les informations nécessaires pour l'allocation, à savoir :

- 65 • Porte de tête : porte sur laquelle les calculs doivent être réalisés.
- 66 • Type d'objectif : Q (probabilité de défaillances), Qavg (probabilité moyenne de défaillances), E (nombre
67 d'occurrences) ou F (défiabilité).
- 68 • Objectif : valeur numérique fixée comme objectif.
- 69 • Précision : écart acceptable par rapport à l'objectif (par défaut il s'agit de l'objectif divisé par 1000). Cette valeur
70 influe sur le temps de calcul : le nombre d'itérations sera plus élevé pour une valeur faible.
- 71 • Temps de mission : valeur numérique fixée pour le temps utilisé pour les calculs.
- 72 • Paramètres à allouer : probabilités ou lambdas.

73
74 Quelques vérifications préliminaires sont réalisées sur les informations renseignées par l'utilisateur (objectif ≤ 1 , temps > 0 ...)
75 afin d'éviter les erreurs. Une fois ces vérifications effectuées, la première étape consiste à explorer les événements de base de
76 l'arbre constitué par la porte de tête sélectionnée. Le but de cette exploration est d'analyser si les événements de base utilisent
77 un paramètre du type recherché (probabilité ou lambda) présentant également un marqueur d'allocation. Le cas échéant, le nom
78 du paramètre ainsi que le poids affecté à celui-ci sont stockés pour les besoins des calculs.

79



80
81 Fig. 1. Logigramme étapes 1 et préliminaires.

82 Une seconde étape est mise en œuvre afin de vérifier deux conditions préalables à la poursuite de l'allocation, à savoir que :

- 83 • L'objectif fixé est accessible ; un calcul avec une valeur de 0 pour les paramètres alloués (probabilités ou lambdas),
84 avec les éventuels autres paramètres (non alloués) inchangés, est réalisé. Si le résultat obtenu est supérieur à l'objectif,
85 celui-ci ne pourra pas être atteint car les valeurs non allouées entraînent le non-respect de l'objectif.
- 86 • L'allocation est nécessaire ; un calcul avec une valeur de 1 (pour les probabilités) ou une valeur équivalente à un
87 dixième du temps de mission (pour les lambdas) est réalisé. Si le résultat obtenu est inférieur à l'objectif, celui-ci

88
89
90

restera tenu quelle que soit la valeur de probabilité (ou avec une valeur très élevée pour les lambdas), l'allocation n'est donc pas nécessaire.

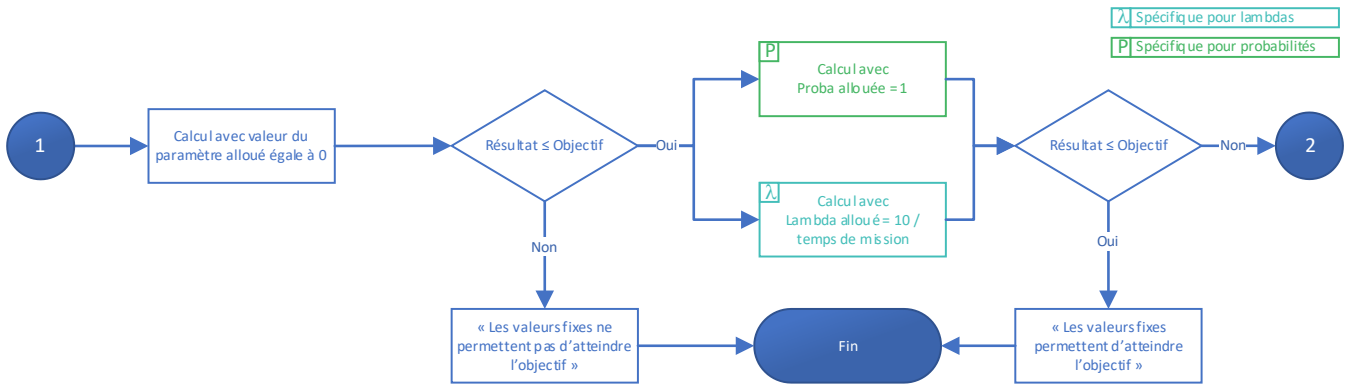


Fig. 2. Logigramme étape 2.

91
92

Lors de la 3^{ème} étape, le module cherche à déterminer les valeurs des paramètres alloués permettant d'obtenir un résultat inférieur à l'objectif dans un intervalle correspondant à la précision. Par exemple pour un objectif de 10^{-2} et une précision de 10^{-4} , le résultat sera considéré conforme s'il est compris dans l'intervalle $[9,90 \cdot 10^{-3} ; 10^{-2}]$. A partir de cette étape, une dichotomie est réalisée sur la base de 3 valeurs :

- Var : variable utilisée pour faire évoluer les paramètres lors de chaque itération. Les paramètres sont fixés en multipliant cette variable Var par le poids affecté au paramètre.
- DVaT : historisation de la dernière valeur de Var pour laquelle l'objectif était respecté (résultat < objectif).
- DVaNT : historisation de la dernière valeur de Var pour laquelle l'objectif n'était pas respecté (résultat > objectif).

La dichotomie est réalisée en deux phases :

- Phase 1 : Afin d'obtenir rapidement un ordre de grandeur permettant de respecter l'objectif fixé, des calculs sont réalisés en divisant la valeur Var par 10 lors de chaque itération jusqu'à obtenir un résultat inférieur à l'objectif.
- Phase 2 : La valeur de Var est ensuite affinée en fixant une valeur correspondant à la moyenne de DVaT et DVaNT jusqu'à obtenir une valeur dans l'intervalle [Objectif - Précision ; Objectif].

100
101
102
103
104
105
106

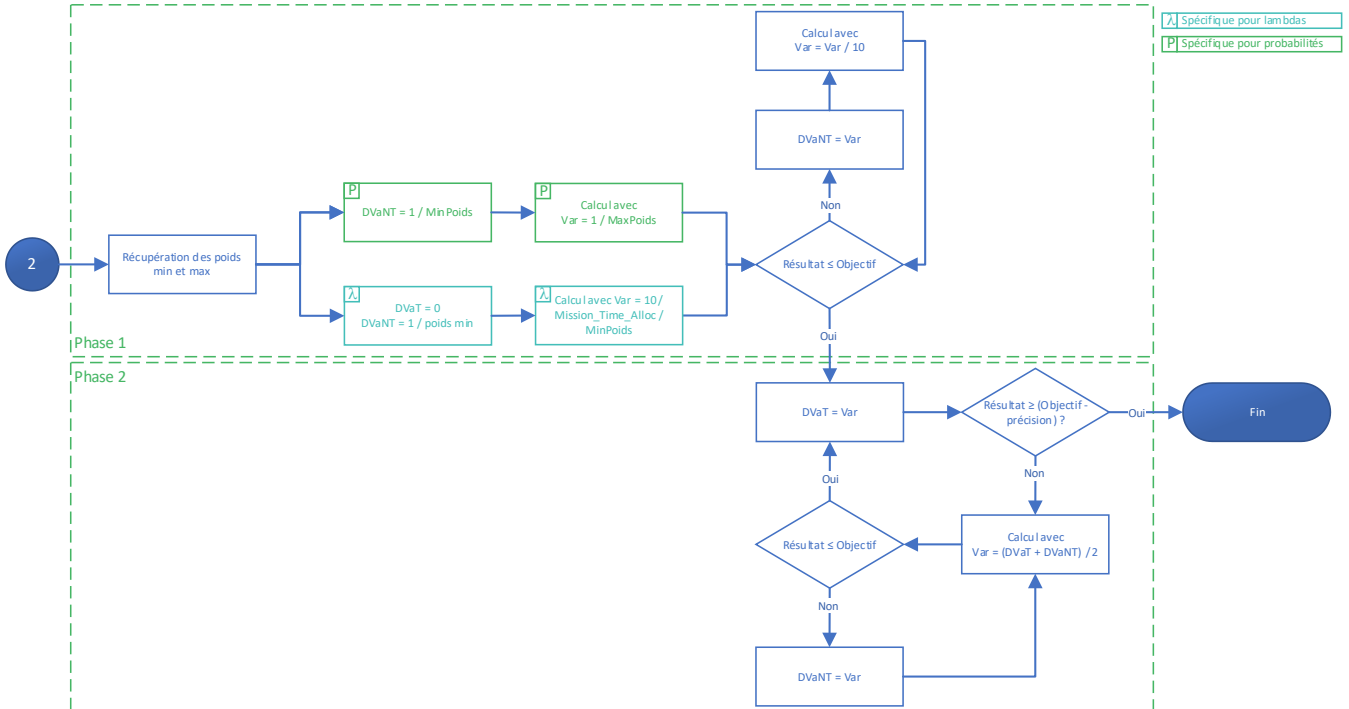


Fig. 3. Logigramme étape 3.

107
108

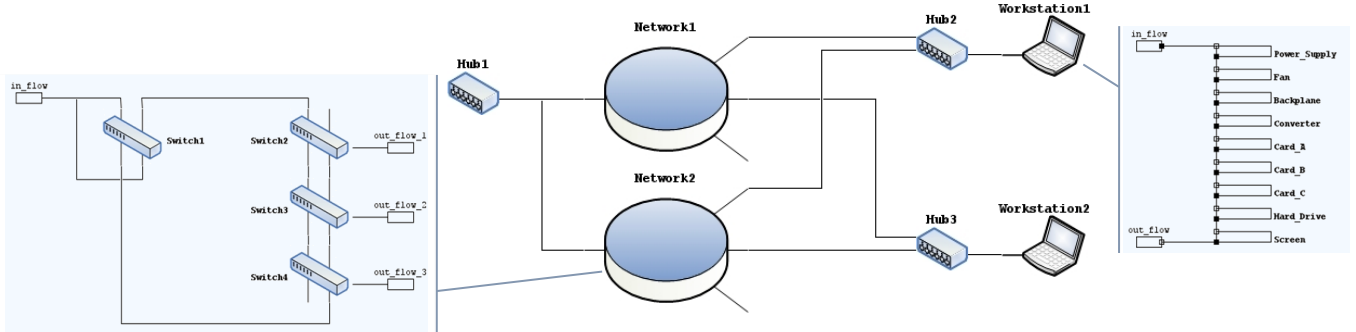
A l'issue de cette étape, une fenêtre s'affiche pour rappeler les hypothèses de calcul et donner les résultats de l'allocation.

109
110
111
112

113

IV. CAS D'APPLICATION

114 Afin d'illustrer la méthode employée, un système de surveillance (Milcent F. et al., 2014), similaire à certains systèmes présents
 115 sur les navires développés par Naval Group, est traité à titre d'exemple. Son rôle est de récupérer des informations (au niveau
 116 du Hub1) et de les transmettre, via un réseau informatique (dont le matériel est imposé et les données de fiabilité sont connues),
 117 aux opérateurs sur leurs stations de travail (Workstation1 et Workstation2). Il est considéré disponible si les informations sont
 118 fournies aux opérateurs des deux stations de travail. L'objectif est d'obtenir une indisponibilité inférieure à 10^{-2} au bout de 70
 119 jours (1680 h).



120

121

Fig. 4. Représentation graphique du cas d'application.

122 L'indisponibilité de ce système a été modélisée sous la forme d'un arbre de défaillances présenté ci-après. Deux lois ont été
 123 utilisées pour calculer les probabilités de défaillances : la loi Gamma-Lambda-Mu (1) pour les événements colorés en bleu et
 124 la loi exponentielle (2) pour les autres événements.

125

$$Q(t) = \frac{\lambda}{\lambda + \mu} (1 - e^{-(\lambda + \mu)t}) + \gamma e^{-(\lambda + \mu)t} \quad (1)$$

126

$$Q(t) = 1 - e^{-\lambda t} \quad (2)$$

127

128 Avec

129 $Q(t)$: Probabilité de défaillance à l'instant t

130 λ : taux de défaillance (h^{-1})

131 μ : taux de réparation (h^{-1})

132 γ : probabilité de refus de démarrage

133

134 Le taux de défaillance des switches des réseaux est imposé. Les taux de défaillances des autres matériels font l'objet d'une
 135 allocation pour ce cas d'application. Les poids affectés aux lambdas ont été choisi en fonction d'ordres de grandeur de taux de
 136 défaillances constatés pour ce type de matériel et ont été intégrés, avec les marqueurs d'allocation, dans le champ de description
 137 des paramètres. Les valeurs utilisées sont synthétisées dans le tableau ci-dessous.

138

TABLE I. PARAMÈTRES INITIAUX

Matériel	Loi	λ	Poids	μ	γ
Hub	Exponentielle	5.00E-06	5.00E-06	-	-
Backplane	Exponentielle	5.00E-06	5.00E-06	-	-
Card A	Exponentielle	1.00E-05	1.00E-05	1	0
Card B	Exponentielle	1.00E-06	1.00E-06	-	-
Card C	Exponentielle	8.00E-06	8.00E-06	-	-
Converter	Gamma-Lambda-Mu	1.00E-05	1.00E-05	0.5	0
Fan	Gamma-Lambda-Mu	5.00E-05	5.00E-05	1	0
Hard Drive	Exponentielle	5.00E-06	5.00E-06	-	-
Power Supply	Exponentielle	1.00E-06	1.00E-06	-	-
Screen	Gamma-Lambda-Mu	5.00E-05	5.00E-05	0.5	0
Switch	Exponentielle	1.00E-05	-	-	-

139

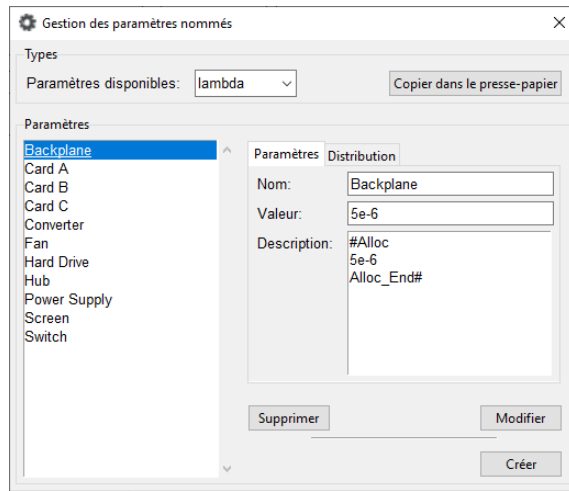
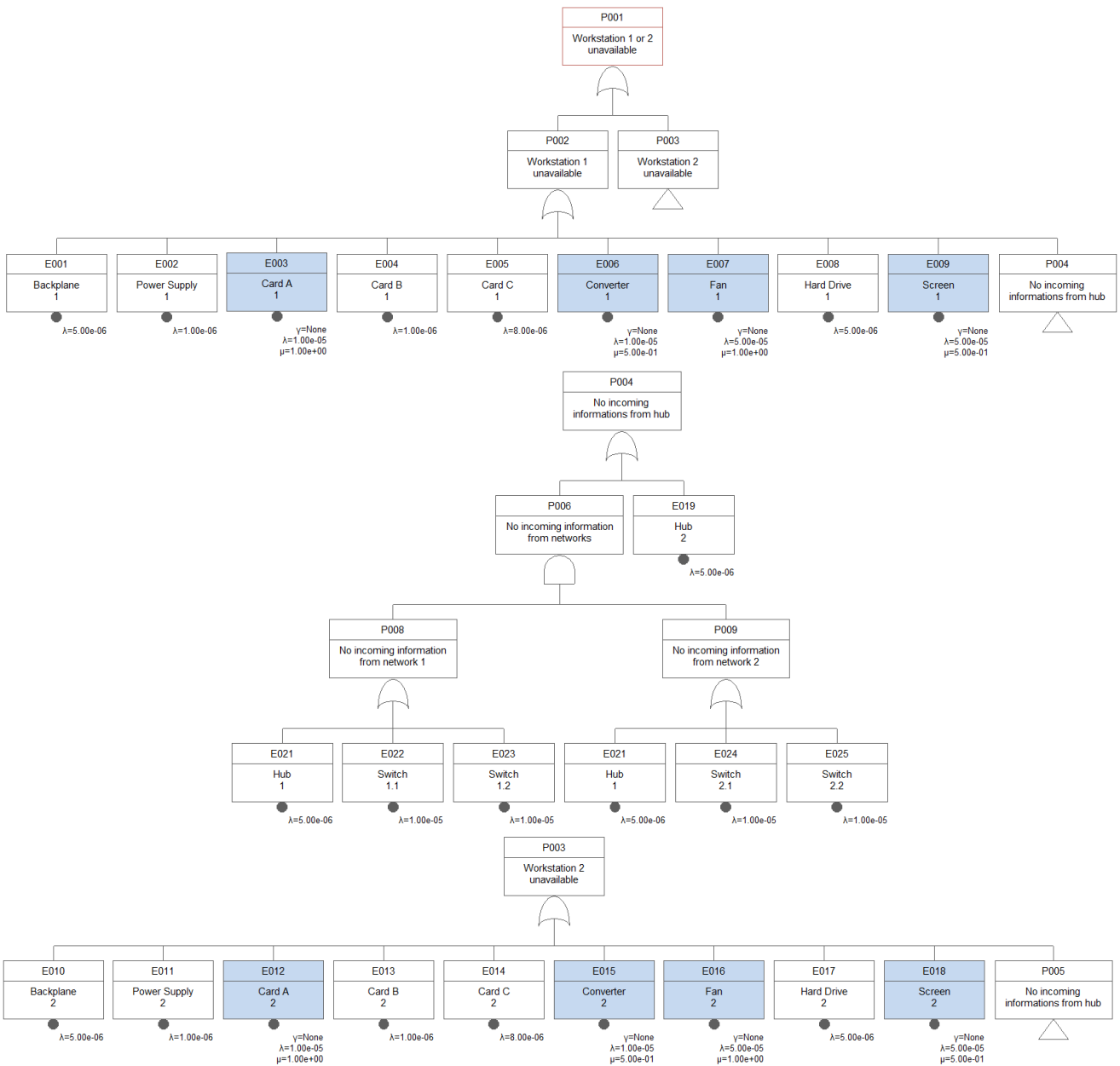


Fig. 5. Description d'un paramètre alloué



140
141

142

143

144

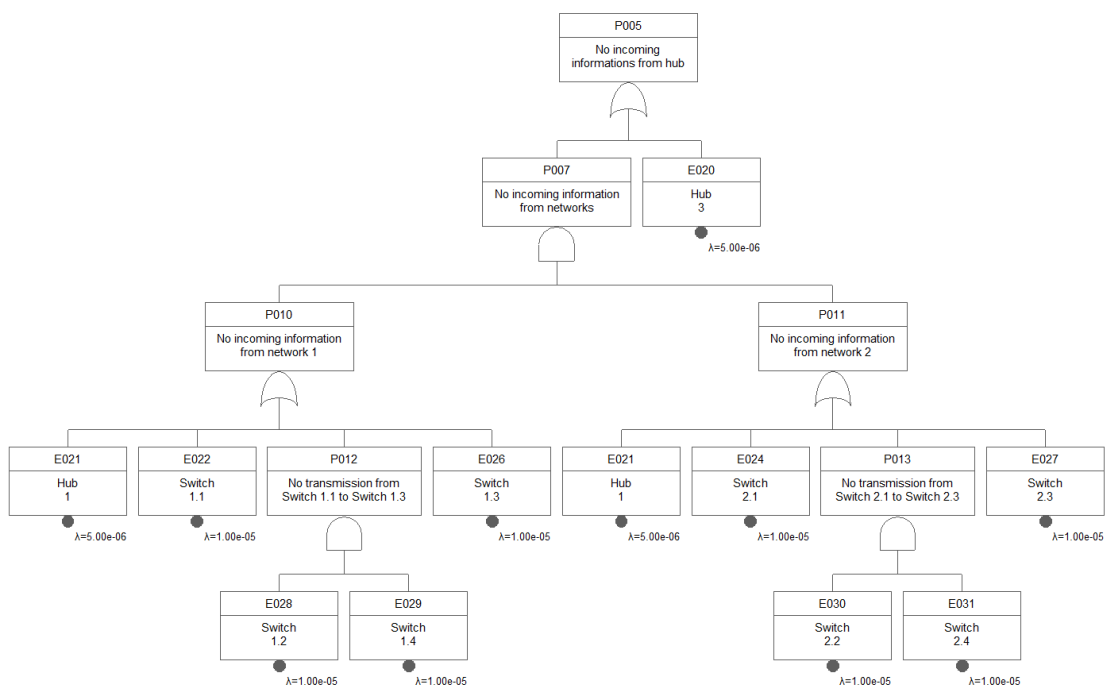


Fig. 6. Arbre de défaillances du cas d'application

145
146

147 Le module d'allocation est ensuite exécuté via le menu d'Arbre-Analyst. Les différents paramètres de calcul sont renseignés
148 dans la boîte de dialogue du module (une précision de 10^{-5} , équivalente à la valeur qui aurait été définie par défaut, a été choisie).
149

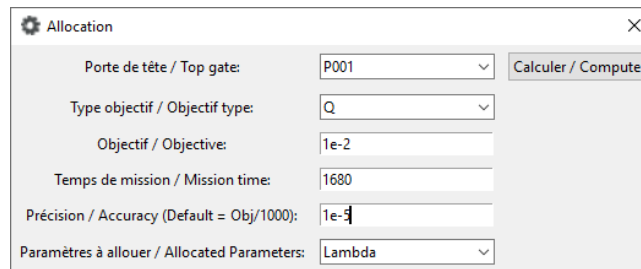


Fig. 7. Boîte de dialogue du module Allocation

150
151

152 Le résultat obtenu (affiché à $1,00 \cdot 10^{-2}$ mais de $9,996 \cdot 10^{-3}$ si l'on prend une valeur plus précise) permet de respecter l'objectif et
153 est bien compris dans l'intervalle $[9,99 \cdot 10^{-3}; 10^{-2}]$. Une fenêtre synthétise les différents lambdas alloués, rappelle les paramètres
154 de calcul et indique le temps de calcul (environ 9s pour ce cas d'application) et le nombre d'itérations (18 pour ce cas
155 d'application). Une fois le processus d'allocation terminé, les valeurs des paramètres ne sont pas réinitialisées, il est donc
156 possible de lancer un calcul « classique » avec les nouveaux paramètres obtenus.
157

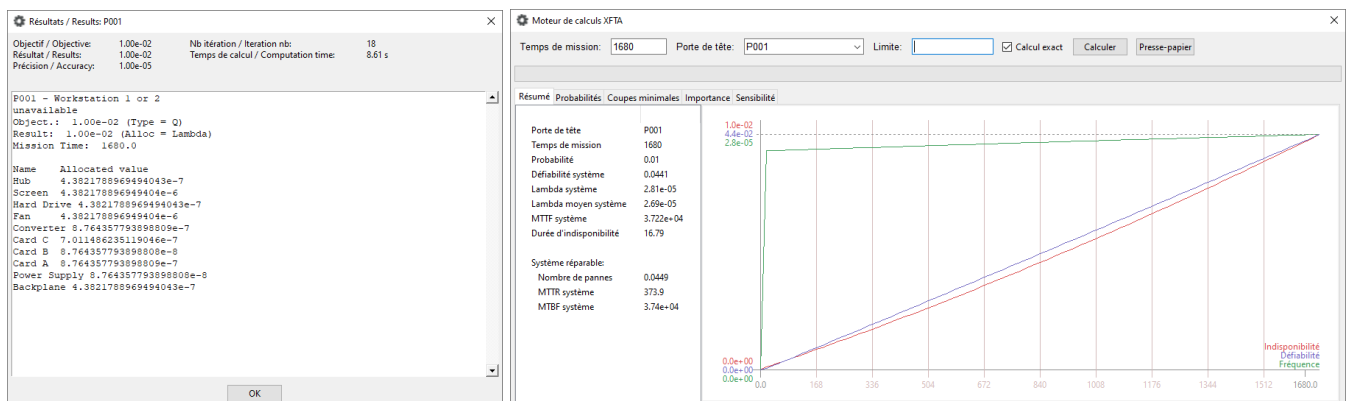


Fig. 8. Résultats obtenus

158
159

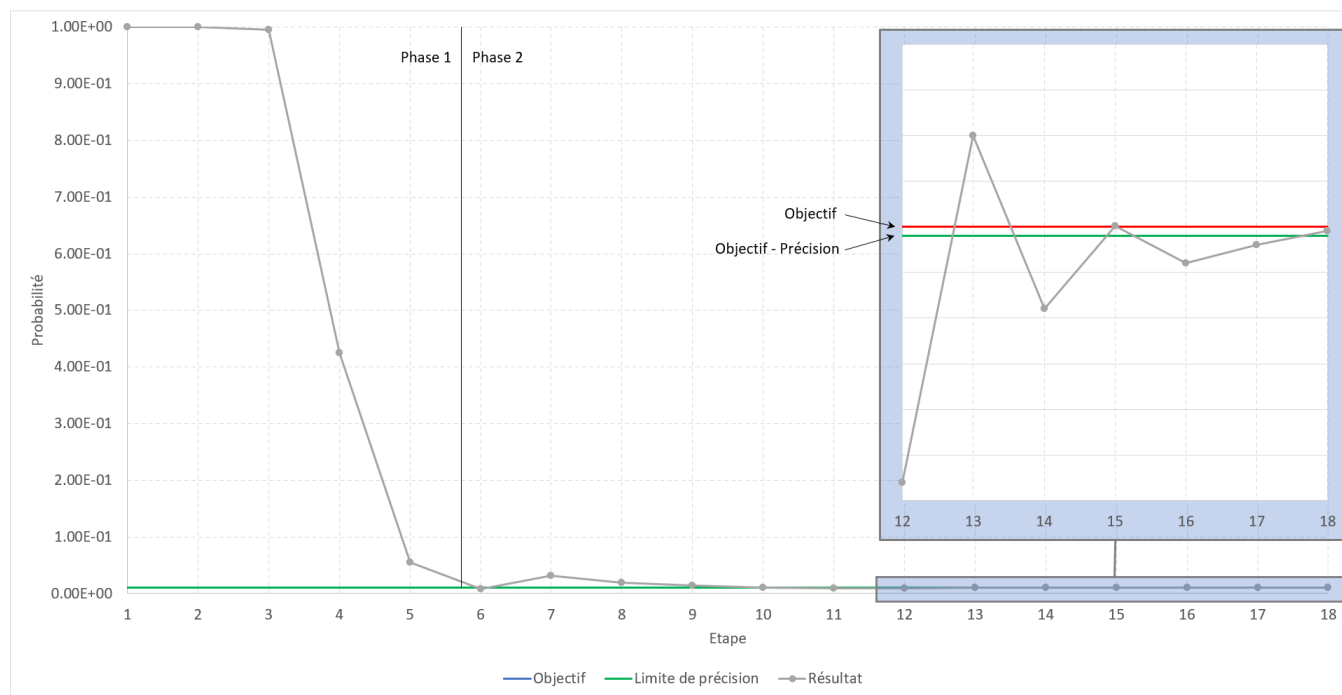
160 Afin de vérifier que le module fonctionne correctement, le processus d'allocation a été réalisé pas-à-pas en modifiant les
161 paramètres à chaque itération. Les détails des paramètres utilisés et des résultats obtenus pour chaque itération est présenté dans

162 le tableau ci-après. Cette analyse montre que l'ensemble des résultats est identique en tous points (allocations, résultat calculé,
 163 nombre d'itérations) à ceux obtenus directement avec le module.

164 TABLE II. DETAIL DES PARAMETRES UTILISES LORS DE CHAQUE ITERATION

Itération	Var	DVaT	DVaNT	Hub	Backplane	Card A	Card B	Card C	Convertir	Fan	Hard Drive	Power Supply	Screen	Switch	Résultat
Initiale	-	-	-	5.00E-06	5.00E-06	1.00E-05	1.00E-06	8.00E-06	1.00E-05	5.00E-05	5.00E-06	1.00E-06	5.00E-05	1.00E-05	-
Prélim 1	-	0	-	0	0	0	0	0	0	0	0	0	0	0	0.0019145
Prélim 2	-	0	1.00E+06	5.95E-03	5.95E-03	5.95E-03	5.95E-03	5.95E-03	5.95E-03	5.95E-03	5.95E-03	5.95E-03	5.95E-03	1.00E-05	1
1	5.95E+03	0	5.95E+03	2.98E-02	2.98E-02	5.95E-02	5.95E-03	4.76E-02	5.95E-02	2.98E-01	2.98E-02	5.95E-03	2.98E-01	1.00E-05	1
2	5.95E+02	0	5.95E+02	2.98E-03	2.98E-03	5.95E-03	5.95E-04	4.76E-03	5.95E-03	2.98E-02	2.98E-03	5.95E-04	2.98E-02	1.00E-05	1
3	5.95E+01	0	5.95E+01	2.98E-04	2.98E-04	5.95E-04	5.95E-05	4.76E-04	5.95E-04	2.98E-03	2.98E-04	5.95E-05	2.98E-03	1.00E-05	0.996
4	5.95E+00	0	5.95E+00	2.98E-05	2.98E-05	5.95E-05	5.95E-06	4.76E-05	5.95E-05	2.98E-04	2.98E-05	5.95E-06	2.98E-04	1.00E-05	4.25E-01
5	5.95E-01	0	5.95E-01	2.98E-06	2.98E-06	5.95E-06	5.95E-07	4.76E-06	5.95E-06	2.98E-05	2.98E-06	5.95E-07	2.98E-05	1.00E-05	5.55E-02
6	5.95E-02	5.95E-02	5.95E-01	2.98E-07	2.98E-07	5.95E-07	5.95E-08	4.76E-07	5.95E-07	2.98E-06	2.98E-07	5.95E-08	2.98E-06	1.00E-05	7.41E-03
7	3.27E-01	5.95E-02	3.27E-01	1.64E-06	1.64E-06	3.27E-06	3.27E-07	2.62E-06	3.27E-06	1.64E-05	1.64E-06	3.27E-07	1.64E-05	1.00E-05	3.18E-02
8	1.93E-01	5.95E-02	1.93E-01	9.67E-07	9.67E-07	1.93E-06	1.93E-07	1.55E-06	1.93E-06	9.67E-06	9.67E-07	1.93E-07	9.67E-06	1.00E-05	1.97E-02
9	1.26E-01	5.95E-02	1.26E-01	6.32E-07	6.32E-07	1.26E-06	1.26E-07	1.01E-06	1.26E-06	6.32E-06	6.32E-07	1.26E-07	6.32E-06	1.00E-05	1.36E-02
10	9.30E-02	5.95E-02	9.30E-02	4.65E-07	4.65E-07	9.30E-07	9.30E-08	7.44E-07	9.30E-07	4.65E-06	4.65E-07	9.30E-08	4.65E-06	1.00E-05	1.05E-02
11	7.63E-02	7.63E-02	9.30E-02	3.81E-07	3.81E-07	7.63E-07	7.63E-08	6.10E-07	7.63E-07	3.81E-06	3.81E-07	7.63E-08	3.81E-06	1.00E-05	8.95E-03
12	8.46E-02	8.46E-02	9.30E-02	4.23E-07	4.23E-07	8.46E-07	8.46E-08	6.77E-07	8.46E-07	4.23E-06	4.23E-07	8.46E-08	4.23E-06	1.00E-05	9.72E-03
13	8.88E-02	8.46E-02	8.88E-02	4.44E-07	4.44E-07	8.88E-07	8.88E-08	7.11E-07	8.88E-07	4.44E-06	4.44E-07	8.88E-08	4.44E-06	1.00E-05	1.01E-02
14	8.67E-02	8.67E-02	8.88E-02	4.34E-07	4.34E-07	8.67E-07	8.67E-08	6.94E-07	8.67E-07	4.34E-06	4.34E-07	8.67E-08	4.34E-06	1.00E-05	9.91E-03
15	8.78E-02	8.67E-02	8.78E-02	4.39E-07	4.39E-07	8.78E-07	8.78E-08	7.02E-07	8.78E-07	4.39E-06	4.39E-07	8.78E-08	4.39E-06	1.00E-05	1.0001E-02
16	8.73E-02	8.73E-02	8.78E-02	4.36E-07	4.36E-07	8.73E-07	8.73E-08	6.98E-07	8.73E-07	4.36E-06	4.36E-07	8.73E-08	4.36E-06	1.00E-05	9.96E-03
17	8.75E-02	8.75E-02	8.78E-02	4.38E-07	4.38E-07	8.75E-07	8.75E-08	7.00E-07	8.75E-07	4.38E-06	4.38E-07	8.75E-08	4.38E-06	1.00E-05	9.98E-03
18	8.76E-02	8.76E-02	8.78E-02	4.38E-07	4.38E-07	8.76E-07	8.76E-08	7.01E-07	8.76E-07	4.38E-06	4.38E-07	8.76E-08	4.38E-06	1.00E-05	9.996E-03

165 La figure ci-après montre l'évolution du résultat obtenu en fonction des itérations. Cette représentation permet de mettre en
 166 évidence les deux phases de l'étape 3 : la recherche rapide d'un ordre de grandeur, puis une dichotomie plus fine afin d'obtenir
 167 un résultat compris dans l'intervalle souhaité.
 168
 169



170 Fig. 9. Evolution du résultat obtenu en fonction des itérations

172 V. RESULTATS ET CONCLUSIONS

173 Les allocations obtenues grâce au module sont en tous points conformes à ceux obtenus par une résolution théorique. Le nombre
 174 d'itérations varient en fonction de la précision souhaitée et le temps de calcul associé est de l'ordre de quelques secondes, pour
 175 des arbres de défaillances de taille réduite, à quelques minutes, pour des arbres de défaillances de grande taille. Le module
 176 d'allocations développé pour Arbre-Analyst permet de répondre à l'objectif de réaliser des allocations de fiabilité (lambdas) ou
 177 de probabilités par facteurs de faisabilité.

178 La création de ce module et la pertinence des résultats obtenus avec celui-ci offrent des perspectives de futurs développements
 179 autour du logiciel Arbre-Analyst à Naval Group : calculs spécifiques, passerelles pour assurer la continuité numérique avec
 180 d'autres analyses ou logiciels...

181 La liberté de diffusion et d'utilisation d'Arbre-Analyst ainsi que la possibilité de créer des modules complémentaires pour des
 182 besoins spécifiques sont des atouts majeurs qui permettent d'encourager une démarche de partage mutuel au sein de la

183 communauté d'utilisateurs de ce logiciel. Naval Group souhaite favoriser cette démarche collaborative en mettant à disposition
184 le module d'allocations (disponible sur le site dédié à Arbre-Analyst ou sur demande).

185 REFERENCES

186 [R01] Clément, É., Rauzy, A., Milcent, F., & Thomas, T. (2022). Arbre-Analyst -Présentation des évolutions du moteur de
187 calculs XFTA. HAL (le Centre Pour la Communication Scientifique Directe). [https://hal-
188 03878513](https://hal.archives-ouvertes.fr/hal-03878513)

189 [R02] Epstein, S., Rauzy A. (2008). Open-PSA Model Exchange Format (The Open PSA initiative). [https://open-
190 psa.github.io/joomla1.5/index.php](https://open-psa.github.io/joomla1.5/index.php)

191 [R03] Rauzy, A., Dupart, C., Morel, P., Thomas, P., (2010). Allocations préliminaires de disponibilité avec Aralia, Congrès
192 Lambda Mu 17, La Rochelle

193 [R04] Milcent, F., Prosvirnova, T., (2014). Optimisation de la maintenance d'une flotte de matériels basée sur une modélisation
194 dynamique, Congrès Lambda Mu 19, Dijon