

## OPTIMISATION DE LA MAINTENANCE D'UNE FLOTTE DE MATERIELS BASEE SUR UNE MODELISATION DYNAMIQUE

### OPTIMIZATION OF THE MAINTENANCE OF A FLEET OF EQUIPMENT WITH A MODEL-BASED ANALYSIS

Frédéric MILCENT

DCNS

Rue du Pont Neuf – CS 81030

16600 RUELLE SUR TOUVRE

+33 (0)5 45 24 30 00

frederic.milcent@dcnsgroup.com

Tatiana PROSVIRNOVA

LIX, Ecole Polytechnique

Bat. Alan Turing

91120 PALAISEAU

+33 (0)1 77 57 80 91

prosvirnova@lix.polytechnique.fr

#### Résumé

L'objectif de cette communication est de montrer qu'il est possible d'utiliser les modélisations dynamiques pour optimiser la maintenance des systèmes. La méthode d'optimisation proposée est basée sur une modélisation en langage AltaRica. Ce langage permet de prendre en compte les principales contraintes de maintenance rencontrées sur les systèmes navals. Afin d'illustrer la méthode, celle-ci est appliquée sur un système similaire à certains présents sur les navires développés par DCNS.

#### Summary

The objective of this paper is to show that it is possible to use dynamic modeling to optimize maintenance of systems. The optimization method proposed is based on the modeling language AltaRica. This language allows to take into account the main constraints encountered on the maintenance of naval systems. To illustrate the method, it is applied to a system similar to some on board of vessels developed by DCNS.

#### Objectifs

Le recours à des modélisations dynamiques est de plus en plus fréquent pour évaluer la disponibilité et/ou la sécurité des systèmes. L'objectif de cette communication est de montrer que ce type de modélisation peut également être utilisé pour optimiser la maintenance des systèmes (stock de rechanges, personnel de maintenance, approvisionnement...).

#### Contexte

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

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

#### Méthode

Il existe plusieurs méthodes pour modéliser de façon dynamique un système : chaîne de Markov, Réseaux de Petri, BDMP, langages de modélisation (AltaRica, Figaro)... C'est le langage de modélisation AltaRica (Boiteau *et al.*, 2006) qui a été choisi car il dispose d'une richesse d'expression permettant de traduire l'ensemble des spécificités du processus de maintenance associé au système. Sa structure hiérarchique permet également une vérification relativement simple contrairement aux Réseaux de Petri ou aux chaînes de Markov qui génèrent souvent un nombre d'éléments graphiques (places ou états) assez importants et donc des difficultés en termes de vérification (Bouissou, 2008).

Le modèle est élaboré à partir d'une démarche (Riera *et al.*, 2012) permettant de valider que les aspects fonctionnels et dysfonctionnels ont bien été implémentés.

La démarche retenue repose sur plusieurs étapes :

- Constitution du modèle du système,
- Mise en place et optimisation du stock de rechange présent à bord,
- Prise en compte de la notion de flotte,
- Optimisation du stock de rechange présent à la base.

### 1. Constitution du modèle du système

Afin d'adapter les capacités de maintenance à la phase d'exploitation dans laquelle se trouve le système, chaque élément ou sous-ensemble du modèle possède, au minimum, un état de type "énuméré" nommé *situation* qui correspond à la phase dans laquelle se trouve l'élément ou le sous-ensemble et qui peut prendre les valeurs suivantes : *waiting* (en attente), *sea* (en mer) et *dock* (à quai).

La valeur *waiting* permet d'appliquer un décalage entre les départs de chaque navire. La transition (*delay*) entre les valeurs *waiting* et *sea* est caractérisée par une loi de type Dirac avec comme paramètre la durée du décalage souhaité. Les transitions (*patrol* et *docking*) entre les valeurs *sea* et *dock* sont caractérisées par des lois de type Dirac avec comme paramètre la durée de la phase en question. Toutes les transitions liées à cet état sont synchronisées au niveau supérieur jusqu'au niveau du navire.

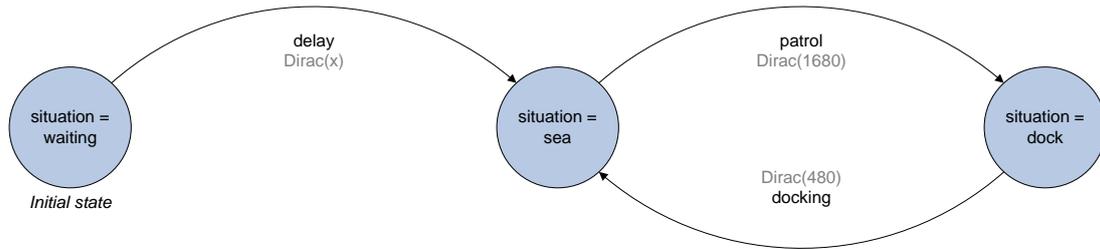


Figure 1. Graphe d'état "situation"

Les éléments soumis à défaillance comportent un état booléen supplémentaire (*working*). Cet état est représentatif du fonctionnement (ou de la panne) de l'élément concerné. Les transitions (*failure* et *repair*) entre les valeurs *true* et *false* sont caractérisées par des lois de type exponentielle avec comme paramètres le taux de défaillance  $\lambda$  et le taux de réparation  $\mu$ . La transition *repair* (entre *false* et *true*) est également conditionnée par l'état *working = true*.

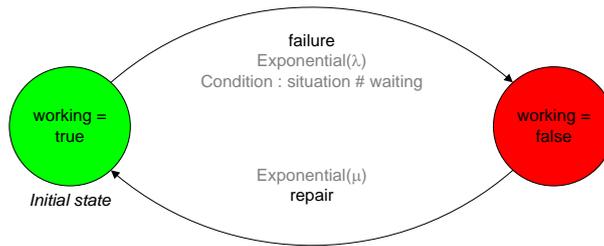


Figure 2. Graphe d'état "working"

```

node Example
flow
  in_flow:bool:in;
  out_flow:bool:out;
state
  working:bool;
  situation:{dock, sea, waiting};
event
  failure,
  repair,
  patrol,
  docking,
  delay;
trans
  (situation = waiting) |- delay -> situation := sea;
  (situation = dock) |- docking -> situation := sea;
  (situation = sea) |- patrol -> situation := dock;
  ((working = true) and (situation # waiting)) |- failure -> working := false;
  (working = false) |- repair -> working := true;
assert
  out_flow = ((if (working = true) then in_flow else false));
init
  working := true,
  situation := waiting;
extern
  law <event failure> = exponential(lambda);
edon
  
```

Figure 3. Code AltaRica d'un composant du système

## 2. Mise en place et optimisation du stock de rechange présent à bord

Pour modéliser le stock à bord, un composant est créé pour chaque type d'éléments du système. En plus de l'état *situation* commun à tous les composants du système, celui-ci comporte un état *quantity* qui représente le nombre d'éléments en stock et dont la valeur est comprise entre 0 et n (n étant le nombre initial de rechange). La transition *taking* décrémente la valeur de *quantity* lors du prélèvement d'un rechange grâce à une synchronisation avec la transition *repair* des éléments concernés du système. La transition *filling* incrémente la valeur de *quantity* lors de la remise à niveau du stock à bord lors du passage du navire à quai. Un observateur (*Stock*) est également intégré au composant afin d'obtenir la valeur finale de *quantity* et donc la quantité d'éléments en stock en fin de simulation.

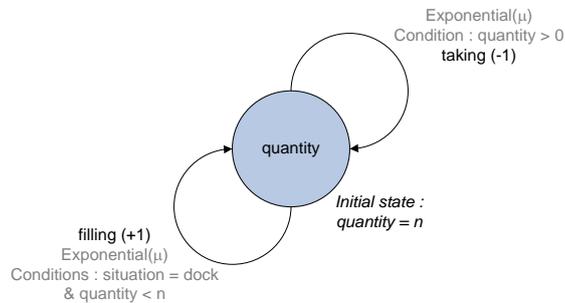


Figure 4. Graphe d'état "quantity"

```

node spare
state
  quantity:float;
  situation:{dock, sea, waiting};
event
  delay,
  docking,
  patrol,
  taking,
  filling;
trans
  (quantity > 0) |- taking -> quantity := (quantity - 1);
  ((quantity < n) and (situation = dock)) |- filling -> quantity := (quantity + 1);
  (situation = waiting) |- delay -> situation := sea;
  (situation = dock) |- docking -> situation := sea;
  (situation = sea) |- patrol -> situation := dock;
init
  quantity := n,
  situation := waiting;
extern
  observer <global Stock> = end_value(<term (quantity)>);
edon
  
```

Figure 5. Code AltaRica d'un rechange bord

Le stock à bord est optimisé vis-à-vis d'un objectif de disponibilité moyenne sur la durée d'une patrouille et grâce à l'observateur *Stock* (nombre d'éléments restant en fin de patrouille). Cet observateur est utilisé pour prioriser l'incrémentement des stocks. Une première simulation stochastique (Monte Carlo) est réalisée avec une quantité initiale n égale à 1 pour tous les éléments en stock. Puis, si l'objectif de disponibilité n'est pas respecté, d'autres simulations sont effectuées en faisant varier la valeur n de *quantity* (pour les éléments dont les rechanges présents en stock en fin de simulation sont les plus faibles) jusqu'à obtention d'une disponibilité moyenne satisfaisante.

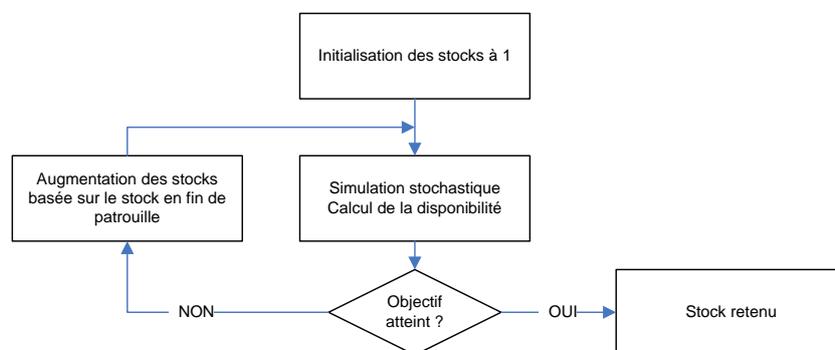


Figure 6. Processus d'optimisation du stock de rechanges bord

### 3. Prise en compte de la notion de flotte

D'autres instances du modèle du système sont ajoutées au modèle de la flotte. De la même façon que pour le stock présent à bord, un stock est mis en place au niveau de la base. Lors de chaque passage à quai, le stock de chaque navire est remis à niveau en puisant dans le stock de la base. Le stock présent à quai est alors partagé entre tous les navires.

### 4. Optimisation des rechanges à la base

Le stock à la base est optimisé pour répondre aux besoins de l'ensemble des systèmes embarqués vis-à-vis du nombre d'éléments restant en fin de simulation par le biais de l'observateur *stock*. Une première simulation stochastique est réalisée avec une quantité initiale d'éléments en stock *n* importante (dépendant de la fiabilité des composants et de la durée de la simulation). Puis, en fonction des résultats obtenus, d'autres simulations sont réalisées en faisant varier la valeur *n* de quantity (nombre d'éléments en stock) afin de converger à une valeur finale comprise entre 1 et 2 (supérieure à 1 pour garantir la non rupture du stock et inférieure à 2 pour limiter le stock au strict nécessaire).

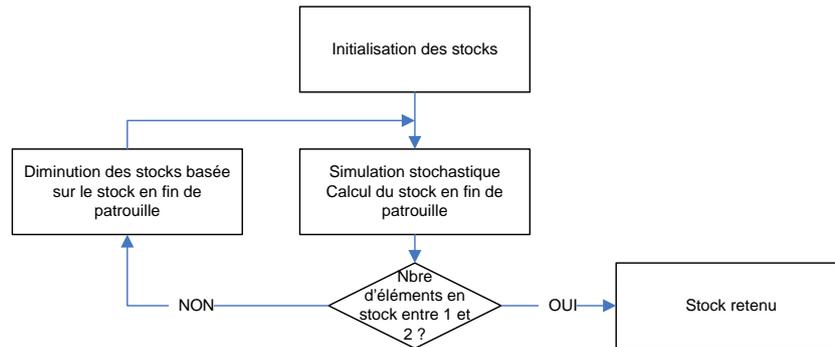


Figure 7. Processus d'optimisation du stock de rechanges base

## Résultats

Afin d'illustrer la méthode employée, un système de surveillance, similaire à certains systèmes présents sur les navires développés par DCNS, est traité à titre d'exemple (voir Figure 8). Son rôle est de récupérer des informations (au niveau du Hub1) et de les transmettre, via un réseau informatique, aux opérateurs sur leurs stations de travail (Workstation1 et Workstation2). Ce système comporte des redondances (double attachement réseau...) et il est considéré disponible si les informations sont transmises à au moins une station de travail.

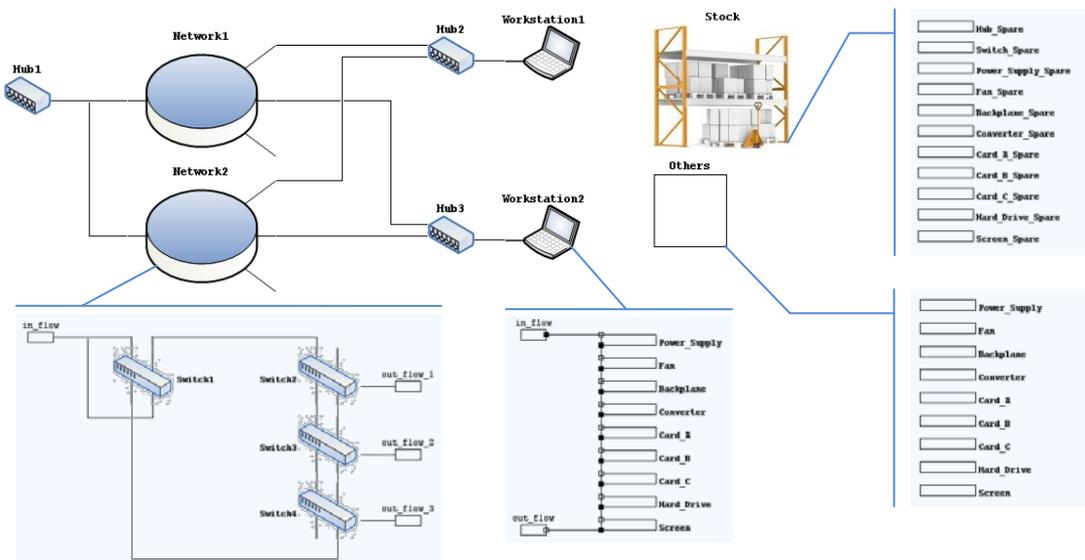
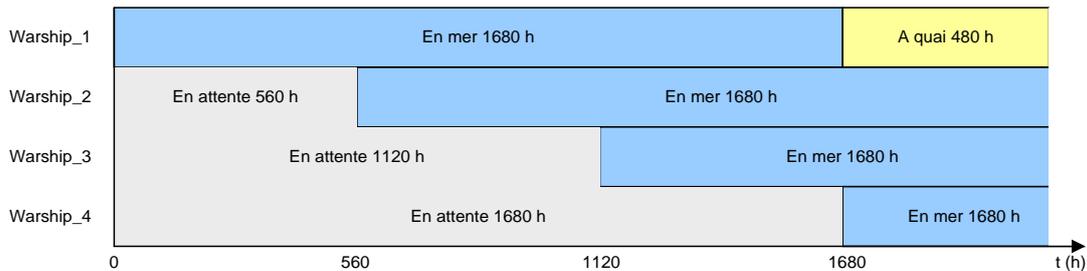


Figure 8. Système étudié

La flotte est constituée de 4 navires dont le profil de mission correspond à une alternance de périodes de 70 jours en mer et de 20 jours à quai. Un délai de 560 h est appliqué entre chaque premier départ à la mer des navires.


**Figure 9.** Profil de mission

Une disponibilité moyenne de 98% durant la première mission est spécifiée pour une chaîne fonctionnelle de transmission de données (c'est cet objectif qui va permettre d'optimiser le nombre de rechange à bord). Les données utilisées pour la quantification de la disponibilité sont synthétisées dans la Table 1.

Designation	$\lambda$	$\mu$	Nbre / navire
Hub	5,00E-06	2	3
Switch	1,00E-05	2	4
Backplane	5,00E-06	2	22
Card A	1,00E-05	2	22
Card B	1,00E-06	2	22
Card C	8,00E-06	2	22
Converter	1,00E-05	2	22
Fan	5,00E-05	2	22
Hard Drive	5,00E-06	2	22
Power Supply	1,00E-06	2	22
Screen	5,00E-05	2	22

**Table 1.** Données utilisées

Seule une chaîne fonctionnelle est étudiée, mais d'autres existent à bord. Celles-ci sont composées des mêmes éléments, partagent le même stock de rechanges affectés et contribuent donc à la consommation de ce stock ce qui impacte indirectement la disponibilité de la chaîne fonctionnelle étudiée. C'est pourquoi il est nécessaire d'ajouter ces éléments au modèle pour représenter les pannes associées aux éléments des autres chaînes et les prendre en compte dans la consommation des stocks.

Afin de simplifier le modèle, les  $n$  éléments supplémentaires associés à chaque type d'élément ont été représentés par un seul composant avec un taux de défaillance  $\lambda_n = n \times \lambda$ . Dans notre cas, cela permet de prendre en compte la demande générée par 720 composants en représentant seulement 36 nœuds dans le modèle, soit une économie de 684 nœuds et d'autant de synchronisations. Le modèle ainsi réalisé comporte 207 nœuds, 403 états et 364 transitions.

Le stock à la base a été évalué pour une période de 70 jours (durée correspondant à la première mission du navire n°1). Trois simulations successives ont été nécessaires afin d'optimiser le stock à bord et d'obtenir une disponibilité moyenne de 98,30% (supérieure à l'objectif de 98%). Les résultats de ces itérations sont présentés dans la Table 2.

Simulations	Nombre d'histoires	Durée du calcul	Moyenne	95% min	95% max
Simulation 1 : Stock initial (1 rechange pour chaque type de composants)	10 000	15,687 s	<b>92,74%</b>	92,38%	93,11%
Simulation 1 : Stock initial + 1 fan +1 screen	10 000	16,234 s	<b>96,66%</b>	96,41%	96,91%
Simulation 2 : Stock simulation 1 + 1 fan +1 screen	100 000	175,115 s	<b>98,30%</b>	98,25%	98,36%

**Table 2.** Résultats des simulations de disponibilité

Le stock à la base a été évalué pour une période de 18 mois et 70 jours (durée correspondant à 18 mois d'exploitation du navire n°4). Quatre simulations successives ont été nécessaires pour optimiser le stock à la base (avec une initialisation des stocks à 100). Chaque itération est réalisée avec 10 000 histoires et la dernière itération fait l'objet d'une simulation supplémentaire à 100 000 histoires.

A titre de comparaison, une évaluation plus classique de ces stocks a été réalisée à partir d'une loi de poisson avec un objectif de Probabilité de Non Rupture de Stock de 95%. La Figure 10 et la Figure 11 présentent les résultats obtenus avec les deux méthodes. Dans cet exemple, les quantités de rechanges préconisées par la simulation sont plus faibles que celles obtenues en utilisant la loi de Poisson permettant ainsi de faire quelques économies sur le coût de ces rechanges.

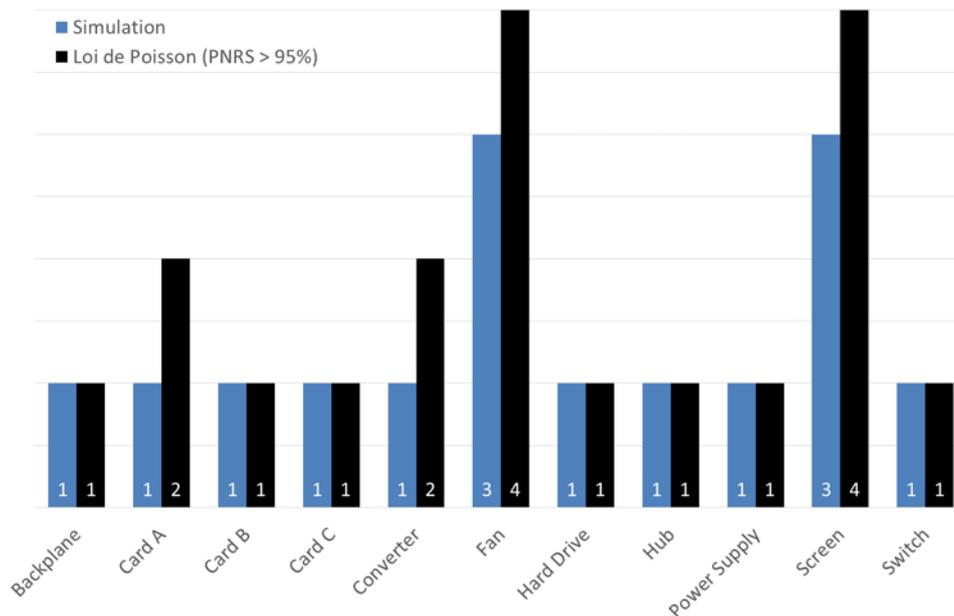


Figure 10. Stock préconisé (à bord)

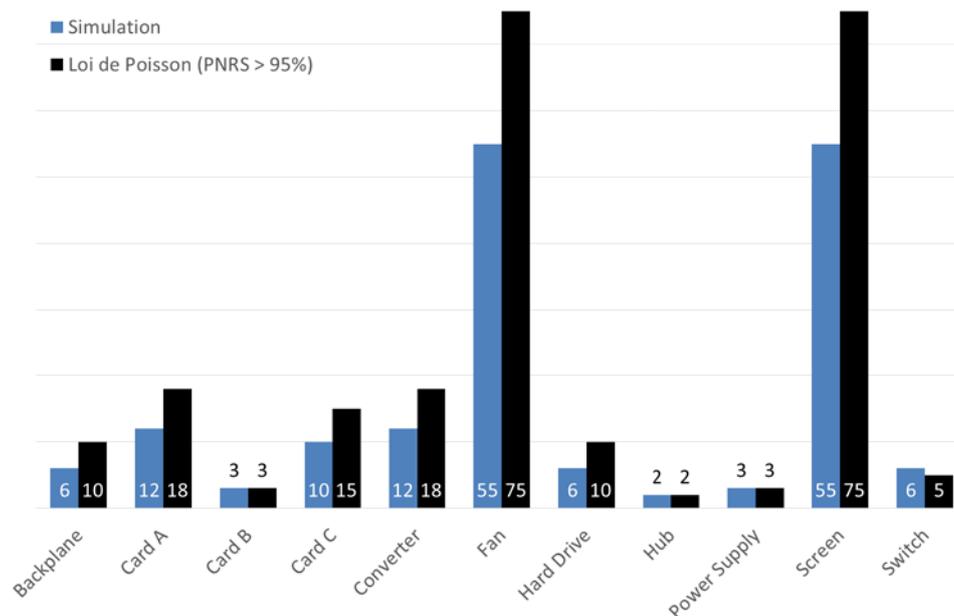


Figure 11. Stock préconisé (à la base)

### Conclusions

Les résultats obtenus permettent de confirmer qu'il est possible d'utiliser les modélisations dynamiques dans le but d'optimiser la maintenance d'une flotte de matériels. En effet, les principales contraintes de maintenance ont pu être intégrées à un modèle représentatif constitué d'une flotte de 4 navires.

Dans ce cas d'application, le stock préconisé est globalement plus restreint que celui obtenu avec une méthode plus classique (basée sur une loi de Poisson). Toutefois, il faut veiller à ne pas généraliser ce résultat car il est lié à la Probabilité de Non Rupture de Stock individuelle fixée.

Les avantages de cette méthode sont les suivants :

- Le stock de rechanges à bord est optimisé par rapport à une estimation réaliste de la disponibilité opérationnelle,
- Le stock de rechanges à la base est optimisé par rapport à une estimation réaliste de la consommation prenant en compte le profil de mission des différents navires,
- Pour ce type d'application, les valeurs recherchées sont assez fortes ce qui ne nécessite pas un nombre d'histoires important pour les simulations stochastiques.

Toutefois, les capacités de traitement des moteurs de calcul doivent être prises en compte lors de la constitution du modèle afin de rendre possible son exploitation. Par exemple, traiter l'ensemble des candidats à la maintenance d'un navire dans le même modèle semble totalement irréalisable, mais une approche par sous-ensemble fonctionnel (conduite le navire, manœuvrer, mettre en œuvre les armes offensives,...) reste possible.

Ces premiers travaux ainsi que les évolutions du langage AltaRica (Prosvirnova *et al.*, 2013) et des moteurs de calcul compatibles (Batteux *et al.*, 2013) ouvrent d'autres perspectives :

- Utilisation de la notion d'héritage implémentée dans le langage AltaRica v3,
- Possibilité de modéliser des politiques de maintenance différentes sur un même équipement selon le niveau technique d'intervention,
- Intégration d'autres paramètres liés au système de soutien (personnel dédié, temps d'approvisionnement, taux de rebut,...),
- Etude de la disponibilité opérationnelle sur de longues périodes,
- Evaluation du Coût Global de Possession,
- ...

### Références

- [1] M. BOITEAU, Y. DUTUIT, A. RAUZY, J.-P. SIGNORET, 2006 The AltaRica Data-Flow language in use: Assessment of Production Availability of a MultiStates System. *Reliability Engineering and System Safety*, 91:747-755.
- [2] T. PROSVIRNOVA, M. BATTEUX, P.-A. BRAMERET, A. CHERFI, T. FRIEDLHUBER, J.-M. ROUSSEL, A. RAUZY, 2013, The AltaRica 3.0 project for Model-Based Safety Assessment, in *Proceedings of 4th IFAC Workshop on Dependable Control of Discrete Systems, DCDS 2013, York (Great Britain)*.
- [3] M. BOUISSOU, 2008, Gestion de la complexité dans les études quantitatives de Sûreté de Fonctionnement de systèmes, pages 81 à 85, LAVOISIER.
- [4] D. RIERA, F. MILCENT, J. PARISOT, E. CLEMENT, 2012, Modélisation dynamique en Sûreté de Fonctionnement : Une avancée pour l'analyse de systèmes complexes, *Actes du Congrès Lambda-Mu 18*.
- [5] M. BATTEUX, A. RAUZY, 2013, Stochastic simulation of AltaRica 3.0 models, In *Proceedings of the European Safety and Reliability Conference, ESREL 2013. Amsterdam (The Netherlands)*.