

## Approches exacte et heuristique pour l'optimisation de la fiabilité dans le cadre d'une maintenance sélective

T. LUST, F. RIANE, O. ROUX

Facultés Universitaires Catholiques de Mons  
Centre de Recherche en Gestion Industrielle (CREGI)  
151, chaussée de Binche 7000 Mons  
riane@fucam.ac.be, roux@fucam.ac.be

P. DEHOMBREUX

Faculté Polytechnique de Mons  
Service de Génie Mécanique  
53, rue du Joncquois 7000 Mons  
pierre.dehombreux@fpms.ac.be

**RÉSUMÉ :** Nous présentons dans cet article deux nouvelles méthodes de résolution du problème de maintenance sélective. Ce problème consiste à trouver le meilleur choix d'actions de maintenance à réaliser sur un système multicomposant, de manière à maximiser sa fiabilité, sachant que la durée de l'arrêt de maintenance est limitée. Lorsque le nombre de composants du système est important, ce problème combinatoire n'est pas simple à résoudre, notamment à cause de la fonction objective non linéaire modélisant la fiabilité du système. Ce problème est peu étudié en littérature. Dès lors, rares sont les méthodes efficaces de résolution qui s'offrent à l'utilisateur. Nous avons donc développé une heuristique et une méthode exacte basée sur une procédure par séparation et évaluation, que nous appliquons à différentes configurations du système. Nous comparons les résultats obtenus, et nous évaluons la meilleure méthode à employer dans différentes situations.

**MOTS-CLÉS :** Maintenance sélective, fiabilité, optimisation combinatoire, heuristique, procédure par séparation et évaluation.

### 1. INTRODUCTION

Dans cet article, nous nous intéressons à la maintenance préventive (Pierskalla and Voelker 1976, Sherif and Smith 1981, Cho and Parlar 1991, Kadi 1999) et plus particulièrement au problème de maintenance sélective, qui consiste à trouver le meilleur choix d'actions de maintenance à réaliser sur un système, sachant que la durée de l'arrêt de maintenance est limitée et qu'il est donc impossible d'effectuer toutes les actions de maintenance. Les applications du problème de maintenance sélective sont multiples : équipements de production dont les actions de maintenance se réalisent le week-end, véhicules entretenus entre deux livraisons, équipements militaires en maintenance entre deux missions, etc. Ce problème a été introduit par Rice et al. (Rice *et al.* 1998) où ils considèrent des systèmes présentant une architecture particulière, des probabilités de panne des composants constantes et un seul type d'action de maintenance (réparer un composant). Cassidy et al. (Cassidy *et al.* 2001) ont étendu ce modèle en considérant des probabilités de panne des composants dépendantes de leur âge, et différents types d'actions de maintenance. Ils résolvent ce problème par une simple énu-

mération de toutes les solutions admissibles, et sur des systèmes de configuration spécifique. Cette énumération ne peut d'ailleurs donner rapidement la solution du problème uniquement sur des systèmes de petite taille. Nous proposons donc deux nouvelles approches de résolution : une heuristique, qui permet de trouver très rapidement une bonne solution, sans garantie d'optimalité, et une méthode exacte, basée sur une procédure par séparation et évaluation.

L'article est organisé de la façon suivante : nous présentons dans un premier temps le problème de maintenance sélective pris en compte et sa modélisation. Nous décrivons ensuite les méthodes de résolution développées et les résultats numériques de ces méthodes sur différents systèmes.

### 2. PRÉSENTATION DU PROBLÈME DE MAINTENANCE SÉLECTIVE

Nous définissons dans cette section le système que nous proposons d'étudier, les différents types d'actions de maintenance considérées, et leurs effets sur le système. Nous présentons également le calcul de la fiabilité d'un système série et/ou parallèle.

Les actions de maintenance sont réalisées sur un système devant exercer une fonction donnée, et défini par un ensemble de composants reliés entre eux en série et/ou en parallèle. Un exemple d'un tel système est présenté à la figure 1. Dans cette représentation, les blocs représentent des composants. La panne d'un des composants 3, 4, 5 ou 10 placés en série entraîne la panne du système. Les composants 6, 7, 8 ou 9 étant placés en parallèle, le système n'est en panne que si les deux branches du sous-système parallèle ne fonctionnent plus. Cela arrive lorsque la panne d'un des composants 6 ou 7 est en combinaison avec la panne d'un des composants 8 ou 9.

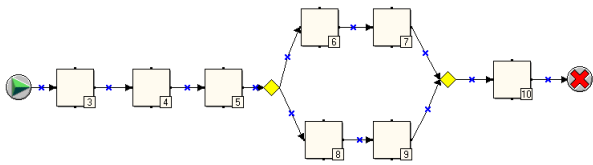


FIG. 1 – Exemple de système multicomposant.

Le système doit effectuer des missions avec des arrêts d'une certaine durée entre chaque mission pendant lesquels sont accomplis les actions de maintenance. Pendant chaque mission, des composants peuvent tomber en panne. Ainsi, à la fin d'une mission, les composants peuvent être dans deux états : en état de fonctionnement ou en panne. Lors de la période consacrée à la maintenance, deux types d'actions sont possibles :

- remplacer un composant en panne ou en état de fonctionnement par du neuf (ou effectuer une opération de maintenance qui remet le composant à neuf).
- effectuer une réparation minimale qui remet en état de fonctionnement un composant en panne.

Chaque action de maintenance consomme du temps mais permet d'augmenter la fiabilité du système, définie par la probabilité qu'il accomplisse la mission. Le problème considéré est de trouver les actions qui vont maximiser la fiabilité d'un système, sous contrainte de temps. On dispose par exemple de maximum deux jours pour effectuer les actions de maintenance.

Nous considérons que les probabilités de fonctionnement des composants pour une mission donnée sont dépendantes de leur âge, c'est-à-dire que plus l'âge d'un composant est élevé, plus sa probabilité de tomber en panne est grande. On modélise les instants de panne d'un composant à travers une loi de fiabilité (Lyonnet 2000, Patton 2004), qui est en fait une loi de probabilité. La plus courante en maintenance est la distribution de Weibull. Grâce aux probabilités

de fonctionnement des composants déterminées grâce à leurs lois de fiabilité, on peut déterminer la probabilité de fonctionnement d'un système pour une mission d'une durée donnée. Celle-ci dépend des probabilités de fonctionnement des composants et de l'architecture du système. Nous considérons ici n'importe quel système série et/ou parallèle. Nous présentons ci-dessous comment calculer la probabilité de fonctionnement d'un composant grâce à sa loi de fiabilité de type Weibull, et la probabilité de fonctionnement d'un système grâce aux probabilités de fonctionnement des composants qui le compose.

### 2.1. Probabilité de fonctionnement d'un composant

La loi de fiabilité (Lyonnet 2000, Patton 2004) donne la probabilité que le composant accomplisse sans panne une mission d'une durée  $t$ . Dans le cas de la distribution de Weibull, la probabilité  $R(t)$  est donnée par la relation :

$$R(t) = e^{-\left(\frac{t}{\eta}\right)^\beta}$$

où  $\beta$  et  $\eta$  sont les paramètres de forme et d'échelle de la distribution de Weibull. Ce sont des réels supérieurs à zéro.

Nous avons représenté la loi de fiabilité  $R(t)$  de type Weibull avec  $\beta = 4$  et  $\eta = 150$  à la figure 2.

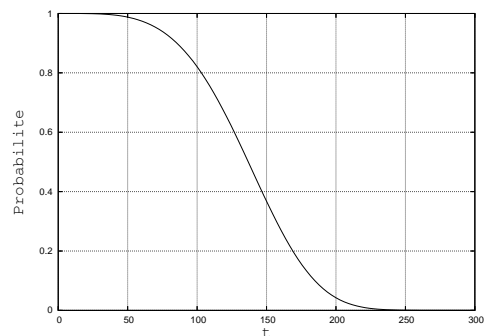


FIG. 2 – Loi de fiabilité de type Weibull( $\beta = 4, \eta = 150$ ).

Si le composant a déjà effectué une mission d'une durée  $T$  et que le composant fonctionne à la fin de cette mission, on utilise la fiabilité conditionnelle pour déterminer la probabilité que le composant accomplisse avec succès une nouvelle mission d'une durée  $t$ , définie par la relation :

$$R(T, t) = \frac{R(T+t)}{R(T)} = \frac{e^{-\left(\frac{T+t}{\eta}\right)^\beta}}{e^{-\left(\frac{T}{\eta}\right)^\beta}}$$

Nous avons représenté à la figure 3 la comparaison de la loi de fiabilité d'un composant neuf et d'un composant âgé de 30 unités de temps, toujours avec la loi de

Weibull( $\beta = 4, \eta = 150$ ). On remarque que les probabilités de fonctionnement pour le composant âgé sont inférieures à celles du composant neuf.

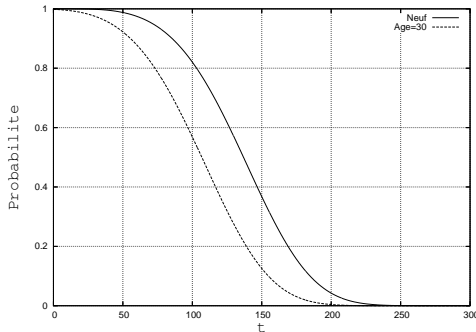


FIG. 3 – Comparaison des lois de fiabilité de type Weibull( $\beta = 4, \eta = 150$ ) pour un composant neuf et un composant âgé de 30 u.t.

Ainsi, à partir de l'âge du composant et de la durée d'une mission, il est facile, dans le cas d'une loi de fiabilité de type Weibull, de déterminer la probabilité que le composant accomplisse la mission.

## 2.2. Probabilité de fonctionnement d'un système multicomposant

On peut aisément décomposer un système multicomposant en un ensemble de sous-systèmes entièrement en série ou en parallèle. La décomposition pour l'exemple de la figure 1 est donnée à la figure 4. On a le sous-système 1 en série formé des composants 6 et 7 (SS1), le sous-système 2 en série formé des composants 8 et 9 (SS2), le sous-système 3 en parallèle formé des sous-systèmes 1 et 2 (SS3). Le système multicomposant global (SG) en série est ainsi formé des composants 3, 4, 5, 10 et du sous-système 3. En utilisant cette décomposition, il est uniquement nécessaire de déterminer les probabilités de fonctionnement de simples systèmes entièrement en série ou en parallèle.

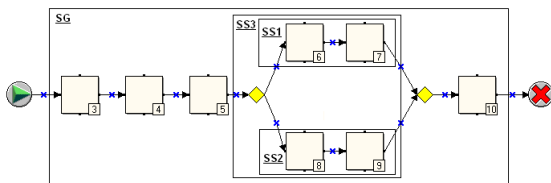


FIG. 4 – Exemple de système multicomposant décomposé.

### 2.2.1 Système en série

Pour qu'un système en série fonctionne après une mission d'une durée  $t$ , il faut que tous les composants  $i$  du système fonctionnent. La probabilité  $RS(t)$  que le système fonctionne après une mission de durée  $t$  est donc égale au produit des probabilités de fonctionnement des  $n$  composants qui le compose :

$$RS(t) = \prod_i^n R_i(t)$$

### 2.2.2 Système en parallèle

Pour qu'un système en parallèle fonctionne après une mission d'une durée  $t$ , il faut qu'au moins un composant du système fonctionne, ou que tous les composants ne tombent pas en panne. Comme la probabilité qu'un composant  $i$  tombe en panne est égale à  $(1 - R_i(t))$ , la probabilité  $RP(t)$  que le système fonctionne après une mission de durée  $t$  est égale au complément à 1 du produit des probabilités de panne des  $n$  composants qui le compose :

$$RP(t) = 1 - \prod_i^n (1 - R_i(t))$$

## 3. MODELISATION DU PROBLEME

La modélisation du problème de maintenance sélective a déjà été abordée par Cassady et al. (Cassady et al. 2001). Nous reprenons ici en partie cette modélisation, en l'étendant à tout système série et/ou parallèle.

Nous désignons respectivement par  $tmr_i$ ,  $tr_i$  et  $trf_i$  les temps requis pour effectuer une réparation minimale sur un composant  $i$  en panne, remplacer un composant  $i$  en panne et remplacer un composant  $i$  en état de fonctionnement. Nous supposons que  $tr_i \geq tmr_i$  et  $tr_i \geq trf_i$ , c'est-à-dire que le temps de remplacement d'un composant en panne est supérieur ou égal aux temps des deux autres actions.

Le système est formé de  $n$  composants, en état de fonctionnement ou en panne à la fin d'une mission  $k$ , selon l'état de la variable binaire  $Y_i(k)$  :

$$Y_i(k) = \begin{cases} 1 & \text{si le composant } i \text{ fonctionne à la fin} \\ & \text{de la mission } k, \\ 0 & \text{autrement.} \end{cases}$$

L'action de réparation minimale sur un composant  $i$  en panne à la fin d'une mission  $k$  est symbolisée par la variable binaire  $W_i(k)$ , de sorte que :

$$W_i(k) = \begin{cases} 1 & \text{si une réparation minimale est réalisée} \\ & \text{entre les missions } k \text{ et } k + 1, \\ 0 & \text{autrement.} \end{cases}$$

De la même façon, l'action de remplacer un composant  $i$  à la fin d'une mission  $k$  est définie par la variable binaire  $V_i(k)$  :

$$V_i(k) = \begin{cases} 1 & \text{si un remplacement est réalisé entre} \\ & \text{les missions } k \text{ et } k + 1, \\ 0 & \text{autrement.} \end{cases}$$

Le temps total  $TMR(k)$  nécessaire à la réalisation des réparations minimales est donné par :

$$TMR(k) = \sum_{i=1}^n tmr_i \cdot W_i(k)$$

Le temps total  $TRF(k)$  consacré aux remplacements des composants en état de fonctionnement est exprimé par :

$$TRF(k) = \sum_{i=1}^n trf_i \cdot V_i(k) \cdot Y_i(k)$$

Le temps total  $TR(k)$  utilisé pour le remplacement des composants en panne est donné par :

$$TR(k) = \sum_{i=1}^n tr_i \cdot V_i(k) \cdot (1 - Y_i(k))$$

Le temps total  $T(k)$  nécessaire à la réalisation des actions de maintenance  $W_i(k)$  et  $V_i(k)$  à la fin d'une mission  $k$  peut donc être exprimé par :

$$T(k) = TMR(k) + TRF(k) + TR(k)$$

Nous considérons également que la durée de la mission  $k + 1$  est égale à  $L(k + 1)$ , que le temps accordé à la maintenance après la mission  $k$  est de  $T_0(k)$  (qui est supposé suffisamment grand pour la remise en fonctionnement du système), que  $B_i(k)$  et  $A_i(k + 1)$  sont respectivement l'âge d'un composant  $i$  après la mission  $k$  et avant la mission  $k + 1$  et que  $X_i(k + 1)$  est l'état d'un composant avant la mission  $k + 1$ . Chaque composant suit une loi de fiabilité de type Weibull de paramètre de forme  $\beta_i$  et de paramètre d'échelle  $\eta_i$ . La fiabilité  $R(k + 1)$  du système avant la mission  $k + 1$  est une fonction  $F$  de l'architecture du système, qui reçoit en argument les probabilités que les composants  $i$  soient en état de fonctionnement à la fin de la mission  $k + 1$  de durée  $L(k + 1)$ . Ces probabilités  $R_i(k + 1)$  sont données pour chaque composant  $i$  par la relation suivante :

$$R_i(k + 1) = \frac{e^{-\left(\frac{L(k+1)+A_i(k+1)}{\eta_i}\right)^{\beta_i}}}{e^{-\left(\frac{A_i(k+1)}{\eta_i}\right)^{\beta_i}}}$$

La modélisation obtenue pour le problème de maintenance sélective, qui consiste à déterminer quels composants remplacer (décisions représentées par les

variables  $V_i(k)$ ) et quels composants réparer de façon minimale (décisions représentées par les variables  $W_i(k)$ ) à la fin d'une mission  $k$  est donnée ci-dessous. On peut remarquer qu'on obtient un problème d'optimisation combinatoire non linéaire.

$$\left[ \begin{array}{l} \max R(k + 1) = F(R_i(k + 1) \cdot X_i(k + 1)) \\ \text{sous} \\ T(k) \leq T_0(k) \\ W_i(k) + V_i(k) \leq 1 \quad \forall i \\ W_i(k) + Y_i(k) \leq 1 \quad \forall i \\ \text{avec} \\ A_i(k + 1) = B_i(k) - B_i(k) \cdot V_i(k) \quad \forall i \\ X_i(k + 1) = Y_i(k) + W_i(k) + \\ V_i(k) \cdot (1 - Y_i(k)) \quad \forall i \end{array} \right]$$

Le problème présente trois contraintes : la première symbolise le fait que le temps d'exécution des opérations de maintenance est limité, la deuxième qu'on ne peut effectuer qu'une des deux actions de maintenance et la troisième qu'on ne peut effectuer l'action de réparation minimale que si le composant  $i$  est en panne ( $Y_i = 0$ ). Les deux dernières égalités permettent uniquement de déterminer l'âge et l'état des composants après la réalisation des opérations de maintenance, ce qui est nécessaire à la détermination de la fiabilité du système.

#### Remarque :

Nous allons ici uniquement nous intéresser à optimiser la fiabilité à la fin d'une mission donnée. Nous nous n'intéressons pas au problème beaucoup plus complexe qui consiste à optimiser globalement les actions de maintenance de manière à maximiser la fiabilité sur un grand nombre de missions. En effet, rien ne dit qu'optimiser localement à la fin d'une mission va permettre d'optimiser la fiabilité sur un nombre important de missions.

## 4. METHODES DE RESOLUTION DU PROBLEME

Deux méthodes ont été développées pour résoudre le problème de maintenance sélective : une heuristique et une méthode exacte basée sur une procédure par séparation et évaluation (PSE). Nous présentons ci-après ces deux méthodes.

### 4.1. Heuristique

La méthode génère dans un premier temps une solution de départ, nécessaire à l'application de l'heuristique proprement dite, dans le cas où le système est en panne après la mission  $k$ . La solution de départ est obtenue par réparation minimale des composants en panne jusqu'à ce que le système soit en état de fonctionner, en considérant en premier lieu les composants les plus critiques, c'est-à-dire ceux situés dans le moins de sous-systèmes.

On effectue ensuite de manière itérative l'action de maintenance qui maximise le rapport défini par la

fiabilité du système après action moins la fiabilité du système avant action, le tout divisé par le temps de l'action. A chaque action réalisée, on actualise la fiabilité du système, qui ne fait que croître à chaque itération. La génération d'une solution de départ, dans le cas où le système est en panne, est justifiée par le fait que dans ce cas particulier, rien ne dit qu'au moins une action pourra augmenter la fiabilité du système. Le choix de l'action réalisée s'effectuerait donc totalement au hasard, ce qui réduirait considérablement les performances de l'heuristique.

Nous ne considérons pas cette heuristique comme une méthode gloutonne car on peut remettre en cause des décisions prises préalablement. En effet, pour les composants en panne, on a le choix de réaliser une action  $V$  ou une action  $W$ . Dans le cas où le choix s'est porté sur une action  $W$ , nous donnons toujours l'occasion de remettre en cause cette décision, et d'effectuer une action  $V$  à la place de l'action  $W$ . Le rapport de l'action  $V$  est alors calculé par la fiabilité du système avec l'action  $V$  à la place de l'action  $W$  moins la fiabilité avant action, le tout sur le temps de l'action  $V$  moins le temps de l'action  $W$ . Ce cas de figure n'arrive en général qu'en fin d'heuristique, lorsqu'il reste assez de temps pour effectuer une action  $V$  à la place d'une action  $W$ , ce qui ne peut faire qu'augmenter la fiabilité du système. Le cas contraire, effectuer une action  $W$  à la place d'une action  $V$ , n'est pas considéré, car la fiabilité du système ne pourrait que diminuer.

Le principe général de l'heuristique est donné ci-dessous :

#### **Initialisation**

Si le système est en panne :

Générer une solution initiale par réparation minimale des composants en panne les plus critiques jusqu'à ce que le système soit en état de fonctionner.

#### **Itération $i$**

Réalisation de l'action maximisant le rapport défini par : la fiabilité du système après action moins la fiabilité du système avant action, le tout divisé par le temps de l'action.

Actualisation de la fiabilité du système.

#### **Critère d'arrêt**

Plus aucune action possible.

## **4.2. Méthode exacte**

L'approche exacte est basée sur une procédure par séparation et évaluation (PSE), qui est une méthode arborescente (branch and bound method) qui procède

par une énumération intelligente de l'espace des solutions. L'énumération est réduite grâce à l'élimination de sous-ensembles de solutions par calcul de bornes sur leurs fonctions d'évaluation. Dans le but d'accélérer la PSE, il est intéressant de posséder une bonne solution initiale. Nous utilisons donc comme solution initiale de la PSE, dans notre cas, la solution trouvée par l'heuristique.

Nous présentons ci-dessous les éléments nécessaires au développement d'une PSE (Lacomme *et al.* 2003), c'est-à-dire :

- la règle de séparation des solutions : comment créer les sous-ensembles de solutions.
- la fonction d'évaluation : comment évaluer les sous-ensembles de solutions.
- la stratégie d'exploration : comment diriger la recherche dans l'arborescence.

### **4.2.1 Règle de séparation**

La règle de séparation est implicite : on choisit un composant  $i$  et on sépare en deux sous-ensembles si le composant est en état de fonctionnement ( $W_i = 0, V_i = 0$  et  $W_i = 0, V_i = 1$ ) et en trois sous-ensembles si le composant est en panne ( $W_i = 0, V_i = 0$ ;  $W_i = 0, V_i = 1$  et  $W_i = 1, V_i = 0$ ).

### **4.2.2 Fonction d'évaluation**

On évalue un sous-ensemble de solutions par relaxation de la contrainte de temps du problème de maintenance sélective : on effectue un remplacement des composants pour lesquels aucune décision n'a déjà été entreprise. De cette manière, on garantit l'obtention d'une borne supérieure pour l'évaluation d'un sous-ensemble de solutions.

### **4.2.3 Stratégie d'exploration**

Deux stratégies d'exploration différentes ont été considérées, ce qui conduit à deux variantes de la PSE :

- Recherche en profondeur d'abord, où on sépare en premier lieu le dernier nœud créé. Cette méthode, appelée procédure par séparation et évaluation séquentielle (PSES), a l'avantage d'être peu gourmande en mémoire.
- Recherche en largeur, où le nœud choisi est celui d'évaluation maximale. On appelle cette méthode procédure par séparation et évaluation progressive (PSEP). Elle présente le désavantage de consommer plus de mémoire que la PSES, mais permet en général d'améliorer assez rapidement la solution initiale.

Le composant sur lequel est réalisé la séparation est comme dans l'heuristique, celui qui présente le meilleur rapport fiabilité du système après action

moins la fiabilité du système avant action, le tout divisé par le temps de l'action. De cette manière, on espère rapidement améliorer la solution courante et éliminer un grand nombre de sous-ensembles de solutions.

## 5. RESULTATS NUMERIQUES

Nous avons appliqué l'heuristique, les deux versions de la PSE et également une méthode exacte procédant par simple énumération de toutes les solutions admissibles, sur des systèmes de dimension et configurations différentes. Le système élémentaire, noté E, qui a servi de base à la création des jeux de données est donné à la figure 5. Il est composé de quatre éléments, en série et en parallèle.

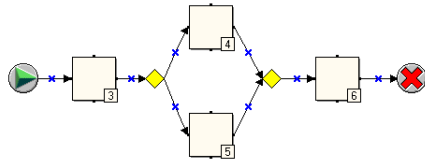


FIG. 5 – Système élémentaire E.

Les caractéristiques des composants de ce système sont données au tableau 1. On peut remarquer que deux des quatre composants sont en panne à la fin de la mission  $k$ . La panne de ces composants entraîne la panne du système, puisque le composant 6 en série est en panne.

Pour générer des systèmes plus complexes, nous reprenons le système élémentaire, qu'on multiplie en l'agencant en série et/ou en parallèle. La mise en série est représentée par le symbole « \* », et la mise en parallèle par le symbole « + ». Par exemple, le système «  $E*(E+E)$  » est composé de deux systèmes élémentaires mis en parallèle, le tout mis en série avec un autre système élémentaire. Ce système comprend donc 12 composants, dont 6 en panne. Nous avons au total généré des systèmes de dimension allant de  $n = 4$  à  $n = 28$ .

La durée de la mission  $L(k)$  est fixée à 40 jours pour tous les systèmes considérés. La période de maintenance  $T_0(k)$  augmente en fonction de la complexité du système (plus il y a de composants en panne, plus on accorde une durée de maintenance élevée).

Les résultats de l'application des deux méthodes exactes PSES et PSEP, de l'énumération et de l'heuristique, pour les différents systèmes générés à partir du système élémentaire sont donnés au tableau 2. Ils ont été obtenus sur un Pentium IV cadencé à 2.4GHz et disposant de 480Mo de mémoire.

On constate que les résultats de l'heuristique sont bons, puisque la méthode permet d'obtenir la solution optimale pour les systèmes à  $n = 4$  et  $n = 8$  composants. Pour les systèmes de plus grande dimension, l'écart maximum par rapport à la solution optimale est de 3.71%. Le temps d'exécution de l'heuristique est de plus très faible, ce qui permet d'obtenir la solution instantanément.

La version PSEP de la PSE est plus rapide que la PSES, pour des systèmes de dimension inférieure à  $n = 16$ . Cette différence au niveau du temps d'exécution est par contre peu significative. Au-delà de  $n = 16$ , la PSES devient plus rapide que la PSEP, ce qu'on explique par la grande place mémoire demandée par la PSEP, qui est nécessaire à la sauvegarde de tous les nœuds de l'arborescence en cours d'exploitation. Au delà de  $n = 24$ , la place mémoire sollicitée par la PSEP est telle que la méthode n'est plus applicable.

L'énumération complète des solutions admissibles devient vite inexploitable pour des systèmes de dimension supérieur à  $n = 20$ . En effet, pour  $n = 20$ , le temps d'exécution est d'environ 30min, et pour  $n = 24$ , ce temps devient égal à environ 15h.

Pour le système de dimension  $n = 28$ , le temps d'exécution de la PSES devient également assez élevé (de l'ordre de l'heure). Il est donc préférable d'utiliser l'heuristique pour des systèmes de grande taille, même si aucune garantie n'est donnée sur la qualité de la solution. Dans l'objectif d'améliorer les résultats, des méthodes approchées génériques, telles que les métaheuristiques, pourraient être développées. Nous retiendrons particulièrement la méthode GRASP (Greedy Randomized Adaptive Search Procedure) (Feo and Resende 1995) puisque son fonctionnement est basée sur une méthode heuristique, telle que celle développée dans cet article.

## 6. CONCLUSION

Nous avons donc proposé dans cet article deux nouvelles méthodes de résolution du problème de maintenance sélective, étendu à toute architecture de système série et/ou parallèle. Nous avons montré que l'heuristique donnait de bons résultats, et de plus rapidement. Les méthodes exactes développées, basées sur une procédure par séparation et évaluation, permettent de réduire considérablement le temps d'exécution d'une énumération complète.

Ces méthodes de résolution pourraient être à la base d'une nouvelle stratégie de maintenance de systèmes multicomposants. Cette stratégie prendrait comme paramètres la durée de la maintenance  $T_0$  et la période  $T$  à laquelle on arrête le système. Il s'avèrerait

donc nécessaire de déterminer la durée  $T_0$  optimale (si la durée est trop courte, peu d'actions de maintenance auront pu être entreprises et le système présentera peu de chances d'accomplir la prochaine mission et si la durée est trop longue, l'indisponibilité du système augmentera) et la période  $T$  optimale (si cette période est trop longue, le système risque de tomber en panne). Pour cela, un modèle de simulation devrait être développé (reproduisant la dynamique du système durant les missions), ainsi que des méthodes d'optimisation pour problèmes continus (pour déterminer les paramètres  $T$  et  $T_0$  continus). Les méthodes d'optimisation combinatoire développées dans cet article interviendraient au moment de l'arrêt du système, pour la détermination des actions de maintenance à entreprendre. D'autres actions, telles que des remplacements imparfaits, pourraient aussi être développées.

Un critère qui n'a pas été considéré dans cette étude, mais qui pourrait être ajouté, est le coût des actions de maintenance. Il pourrait être intégré comme une contrainte dans le problème de maintenance sélective (on dispose d'un budget maximum pour l'accomplissement des actions de maintenance) ou comme un nouveau critère, en plus de la fiabilité. Dans ce dernier cas, on obtiendrait un problème d'optimisation combinatoire multicritère, où il faudrait maximiser la fiabilité et minimiser le coût. Cela impliquerait l'intervention du décideur, puisqu'il devra choisir la solution correspondant le mieux à ses préférences.

## REMERCIEMENTS

Cette recherche est financée par la Direction Générale des Technologies, de la Recherche et de l'Energie de la Région Wallonne, dans le cadre du projet OPTIMAIN.

## Références

- Cassady, C. R., W. P. Murdock and E. A. Pohl (2001). Selective maintenance for support equipment involving multiple maintenance actions. *European Journal of Operational Research* **129**(1), 252–258.
- Cho, D.I. and M. Parlar (1991). A survey of maintenance models for multiunit systems, european journal of operational research. *European Journal of Operational Research* **51**, 1–23.
- Feo, T.A. and M.G.C. Resende (1995). Greedy randomized adaptive search procedures. *Journal of Global Optimization* **6**, 109–133.
- Kadi, D. A. (1999). Les stratégies de maintenance préventives pour la maximisation de la dispo-

nibilité des systèmes. In : *Proceedings 3th International Conference Industrial Automation*. Montréal. pp. pp. 1.13 – 1.16.

- Lacomme, P., C. Prins and M. Sevaux (2003). *Algorithmes de graphe*. Eyrolles.
- Lyonnet, P. (2000). *La maintenance, mathématique et méthodes, 4e Edition*. Edition TEC & DOC.
- Patton, J.D. Jr (2004). *Preventive maintenance 3rd Edition*. ISA - The Instrumentation, System, and Automation Society.
- Pierskalla, W.P. and J. A. Voelker (1976). A survey of maintenance models : The control and surveillance of deteriorating systems. *Naval Research Logistics Quarterly* **23**(3), 353–388.
- Rice, W. F., C. R. Cassady and J. A. Nachlas (1998). Optimal maintenance plans under limited maintenance time. In : *Proceedings of the Seventh Industrial Engineering Research Conference*.
- Sherif, Y.S. and M.L. Smith (1981). Optimal maintenance models for systems subject to failure : a review. *Naval research Logistics Quarterly* **28**, 47–74.

Composant	$\beta$	$\eta$ (jours)	tmr(h)	tr(h)	trf(h)	$B(k)$ (jours)	$Y(k)$
3	3	120	3	5	1	30	1
4	4	150	2	4	2	60	0
5	2.5	130	1	3	2	28	1
6	4	180	2	6	3	56	0

TAB. 1 – Caractéristiques des composants du système élémentaire.

n	Configuration	$T_0(k)$ (h)	Fiabilité optimale	Temps des méthodes exactes(S)			Heuristique		
				PSES	PSEP	Enumération	Fiabilité	Ecart(%)	Temps(S)
4	E	6	0.874	0	0	0	0.874	0	0
8	E*E	12	0.784	0.01	0	0.02	0.784	0	0
	E+E	12	0.987	0.01	0	0.02	0.987	0	0
12	E*(E+E)	18	0.918	0.05	0.05	0.83	0.908	1.09	0.01
	E+(E*E)	18	0.983	0.07	0.04	0.78	0.961	2.24	0
16	E*(E+(E*E))	24	0.925	1.51	0.88	37.92	0.923	0.22	0.01
	E+(E*(E+E))	24	0.994	0.56	0.49	37.62	0.984	1.01	0.01
20	E*(E+(E*(E+E)))	30	0.949	29.81	125.65	1893.09	0.924	2.63	0
	E+(E*(E+(E*E)))	30	0.995	18.22	68.47	1727.90	0.961	3.42	0.01
24	E*(E+(E*(E+(E*E))))	36	0.954	423.59	/	57360	0.924	3.14	0.01
	E+(E*(E+(E*(E+E))))	36	0.997	237.48	/	55920	0.961	3.61	0.01
28	E*(E+(E*(E+(E*(E+E))))	42	0.957	7765.17	/	/	0.924	3.45	0.01
	E+(E*(E+(E*(E+(E*E))))	42	0.998	3483.51	/	/	0.961	3.71	0.02

TAB. 2 – Résultats des différentes méthodes de résolution du problème de maintenance sélective (Pentium IV 2.4 GHz 480 Mo).