

MODELE DE SIMULATION RECURSIVE ET OPTIMISATION

CABARBAYE André ^{1&2}, LAULHERET Roland ¹

¹ *Centre National d'Etudes Spatiales (CNES), 18 av. Edouard Belin, 31401, Toulouse, France
05 61 28 27 41 / 05 61 27 47 19, andre.cabarbaye@cnes.fr / roland.laulheret@cnes.fr*

² *CAB INNOVATION, 3 rue de la Coquille, 31500, Toulouse, France
05 61 54 68 08, andre.cabarbaye@cabinnovation.fr*

Résumé : Outil de base de l'ingénieur trop souvent méconnu, la simulation de Monte-carlo présente un domaine d'application beaucoup plus vaste que celui d'autres méthodes de calcul (analytique, traitement markovien, etc.), mais souffre de trois limitations :

- l'imprécision des résultats obtenus sous la forme d'un intervalle de confiance que les techniques de réduction de variance permettent cependant de resserrer,
- la durée de calcul nécessaire à une évaluation qui rend difficile son couplage à des techniques d'optimisation malgré les performances atteintes par les ordinateurs d'aujourd'hui,
- la difficulté à modéliser un système d'une certaine complexité sans utilisation d'une interface de représentation telle que les réseaux de Petri par exemple pour les systèmes à états discrets.

Cherchant à étendre l'usage de la simulation de Monte-carlo, cette communication propose deux techniques originales :

- un couplage efficace à des outils d'optimisation (algorithmes génétiques, évolution différentielle et simplexe non linéaire) afin de ne plus rendre rédhibitoires les temps de calcul,
- une alternative à l'emploi des réseaux de Petri stochastiques. En effet, bien qu'ils aient rencontré un indéniable succès sur le plan académique, les réseaux de Petri stochastiques présentent parfois des limitations en termes de complexité et de validation quand ils sont confrontés à des problématiques industrielles. Aussi, les modèles de simulation récursive constituent-ils une autre technique de modélisation des systèmes à états discrets qu'ils soient de type hybride (mélangeant des variables aléatoires et continues), markoviens ou non-markoviens (avec ou sans conservation de la mémoire des changements d'états successifs).

Summary: Basic tool for engineers too often poorly known, the Monte-Carlo simulation has a scope much larger than other calculation methods (Analytical, Markovian, etc.), but suffers from three limitations:

- The lack of results accuracy obtained in the form of a confidence interval that the techniques for variance reducing, however, can improve,
- The computing time necessary for an evaluation. That induces difficulties for coupling to optimization techniques despite the performances of today's computers,
- The difficulty to model a complex system without using a representation interface such as Petri nets.

Seeking to expand the use of Monte Carlo simulations, this paper proposes two original techniques:

- An effective coupling with optimization tools (genetic algorithms, differential evolution and nonlinear simplex) in order not to make prohibitive the computing time.
- An alternative to stochastic Petri nets. Indeed, although they have met with undeniable success in the academic networks, stochastic Petri sometimes have limitations in terms of complexity and validation when they are confronted with industrial problems. The recursive simulation modelling constitutes an alternative technique for modelling discrete states systems of hybrid type (mixing continuous and random variables), Markovian or non-Markovian (with or without memory of the successive states transitions).

Mots-clés : Modèles graphiques - Statistique mathématique

1- Introduction

Outil de base de l'ingénieur trop souvent méconnu, la simulation de Monte-carlo présente un domaine d'application beaucoup plus vaste que celui d'autres méthodes de calcul (analytique, traitement markovien, etc.).

Elle souffre toutefois de trois limitations :

- l'imprécision des résultats obtenus sous la forme d'un intervalle de confiance que les techniques de réduction de variance permettent cependant de resserrer,
- la durée de calcul nécessaire à une évaluation qui rend difficile son couplage à des techniques d'optimisation malgré les performances atteintes par les ordinateurs d'aujourd'hui,
- la difficulté à modéliser un système d'une certaine complexité sans utilisation d'une interface de représentation telle que les réseaux de Petri par exemple pour les systèmes à états discrets.

Cherchant à étendre l'usage de la simulation de Monte-carlo, cette communication propose deux techniques originales :

- un couplage efficace à des outils d'optimisation (algorithmes génétiques, évolution différentielle et simplexe non linéaire), afin de ne plus rendre rédhibitoires les temps de calcul,
- une alternative à l'emploi des réseaux de Petri stochastiques. En effet, bien qu'ils aient rencontré un indéniable succès sur le plan académique comme l'atteste le foisonnement des publications dont ils font l'objet, les réseaux de Petri stochastiques présentent parfois des limitations en termes de complexité et de validation quand ils sont confrontés à des problématiques industrielles. Aussi, les modèles de simulation récursive constituent-ils une autre technique de modélisation des systèmes à états discrets qu'ils soient de type hybride (mélangeant des variables aléatoires et continues), markoviens ou non-markoviens (avec ou sans conservation de la mémoire des changements d'états successifs).

2 - Modèle de simulation récursive

Mise en œuvre par l'outil SIMCAB de la société CAB INNOVATION, cette technique de modélisation repose sur la récursivité. Elle ne cherche pas à représenter le comportement d'un système durant toute sa mission, mais se contente de décrire le comportement de celui-ci entre deux instants courants au moyen d'une transition générique entre états (figure 1). Ce servant d'un tableur, elle offre à l'analyste une large capacité et souplesse de représentation.

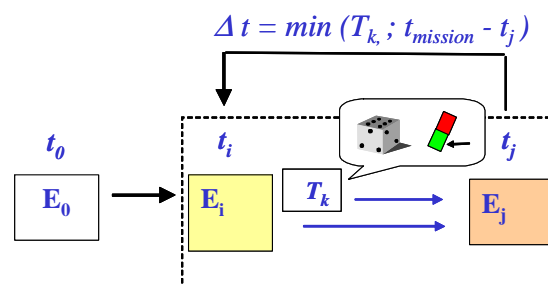


Figure 1. Modèle de simulation récursive

Le modèle décrit la transition générique entre les états E_i (à t_i) et E_j (à t_j). La simulation effectuée par l'outil consiste à recopier un certain nombre de fois l'état E_j , défini dans une plage de cellules du tableur, dans l'état E_i , défini dans une plage similaire, pendant toute la mission, en partant d'un état initial E_0 (à t_0), défini par ailleurs. L'incrément de temps considéré Δt entre t_i et t_j , correspondant à la durée séparant l'occurrence de deux événements successifs, est la plus petite valeur calculée, à l'instant courant, parmi divers incréments de temps T_k , définis dans une autre plage de cellules. Ces durées avant changement d'état sont définies au moyen d'expressions logiques, de calcul ou de fonctions de simulation de variables aléatoires selon diverses lois de probabilité. L'état E_j est également défini au moyen des fonctions du tableur à partir de l'état E_i en considérant la transition réalisée, selon la valeur Δt , parmi les différentes transitions possibles. La simulation peut être

réalisée en mode pas à pas pour valider les modèles, ou durant une mission complète qui est rejouée un grand nombre de fois selon la précision attendue des résultats.

A titre d'exemple, un système hybride constitué d'une simple pompe utilisée pour vider un réservoir dans lequel s'écoule un liquide, est représenté en figure 2. Ce modèle comprend 3 cellules du tableur correspondant à l'état de la pompe (en panne = 0 ou opérationnelle = 1), à l'état du réservoir (débordement = 0 ou normal = 1) et à la hauteur du liquide dans le réservoir, aux instants T_0 , T_i et T_j . Les durées avant défaillance (TTF : Time To Failure) ou de remise en état (TTR : Time To Repair) sont définies par des expressions dans lesquelles la fonction $L_Exp()$ exécute un tirage aléatoire de la loi exponentielle. L'état du système à T_j dépend de son état à T_i et de l'événement survenu à $T_i + \Delta T$ (la plus petite valeur parmi le TTF, le TTR et la durée avant le prochain débordement). Le nombre de débordement et la moyenne du volume de liquide répandu sont calculés par itération sur toute la durée de la mission (10 000 heures). Pour en faciliter la validation, l'ergonomie du modèle peut être améliorée en utilisant certaines fonctionnalités du tableur. Ainsi les formats conditionnels permettent de colorer les cellules en fonction de leur contenu (couleur dépendant de l'état des composants par exemple) et de véritables représentations animées du système peuvent être éventuellement couplées au modèle tel que le niveau de liquide qui est lié à la valeur du paramètre correspondant dans cet exemple.

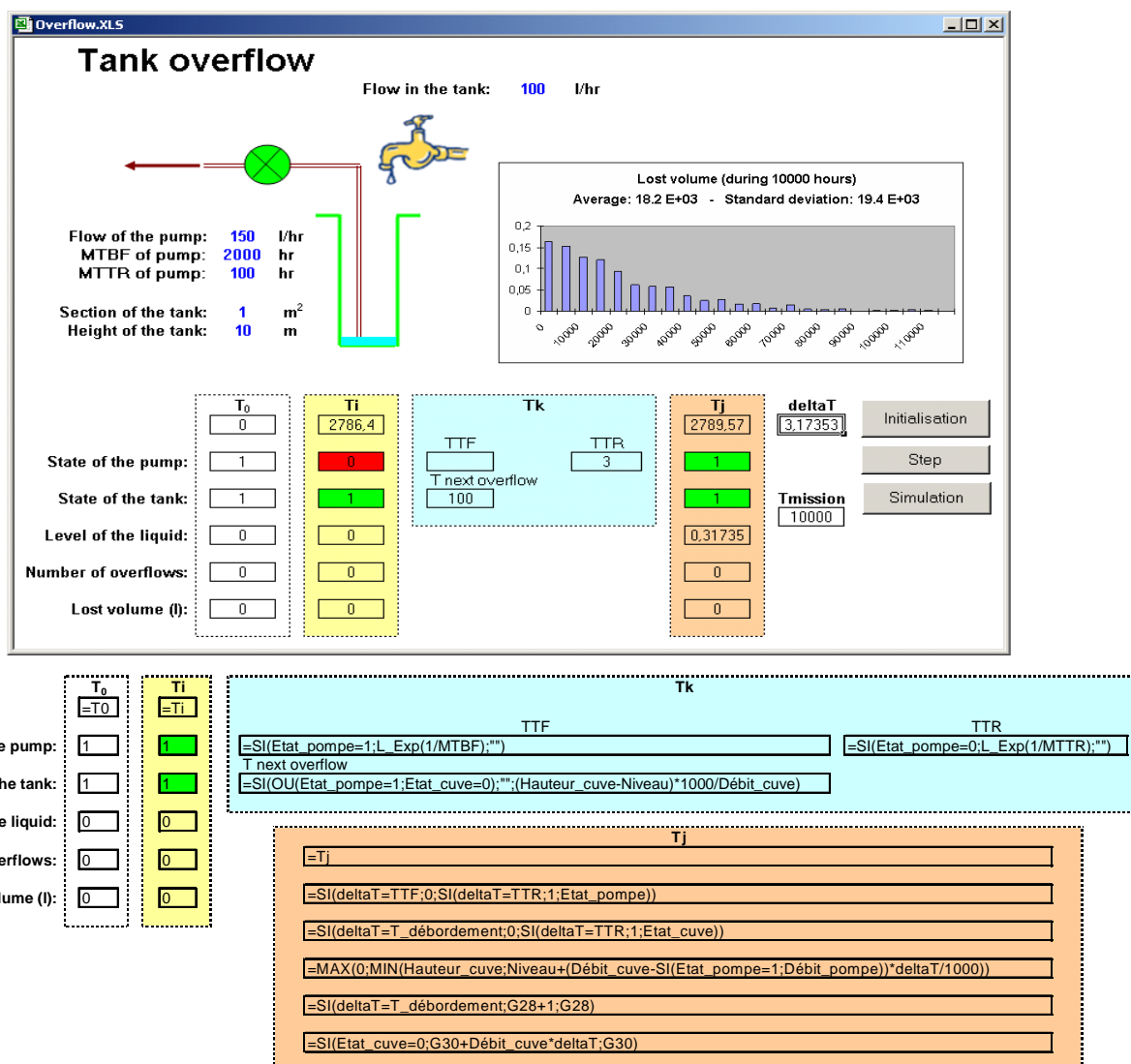


Figure 2. Débordement d'une cuve

Ne faisant appel qu'à des fonctions logiques ou de calcul, le modèle de simulation récursive peut conserver la topologie du système comme le montre l'exemple de la figure 3 correspondant à un réseau de distribution électrique alimenté par 4 centrales. Comprenant de multiples lignes et nœuds

de raccordement, ce modèle permet d'évaluer la disponibilité du service offert à un abonné situé en n'importe quel point X du réseau ; une fonction matricielle de propagation des états énergétiques faisant passer de l'état des constituants du réseau (à T_i) à leur niveau énergétique (en haut de la figure). Des réseaux de télécommunication ou de transport (ferroviaire, aéronautique, routier...) peuvent être traités de manière similaire.

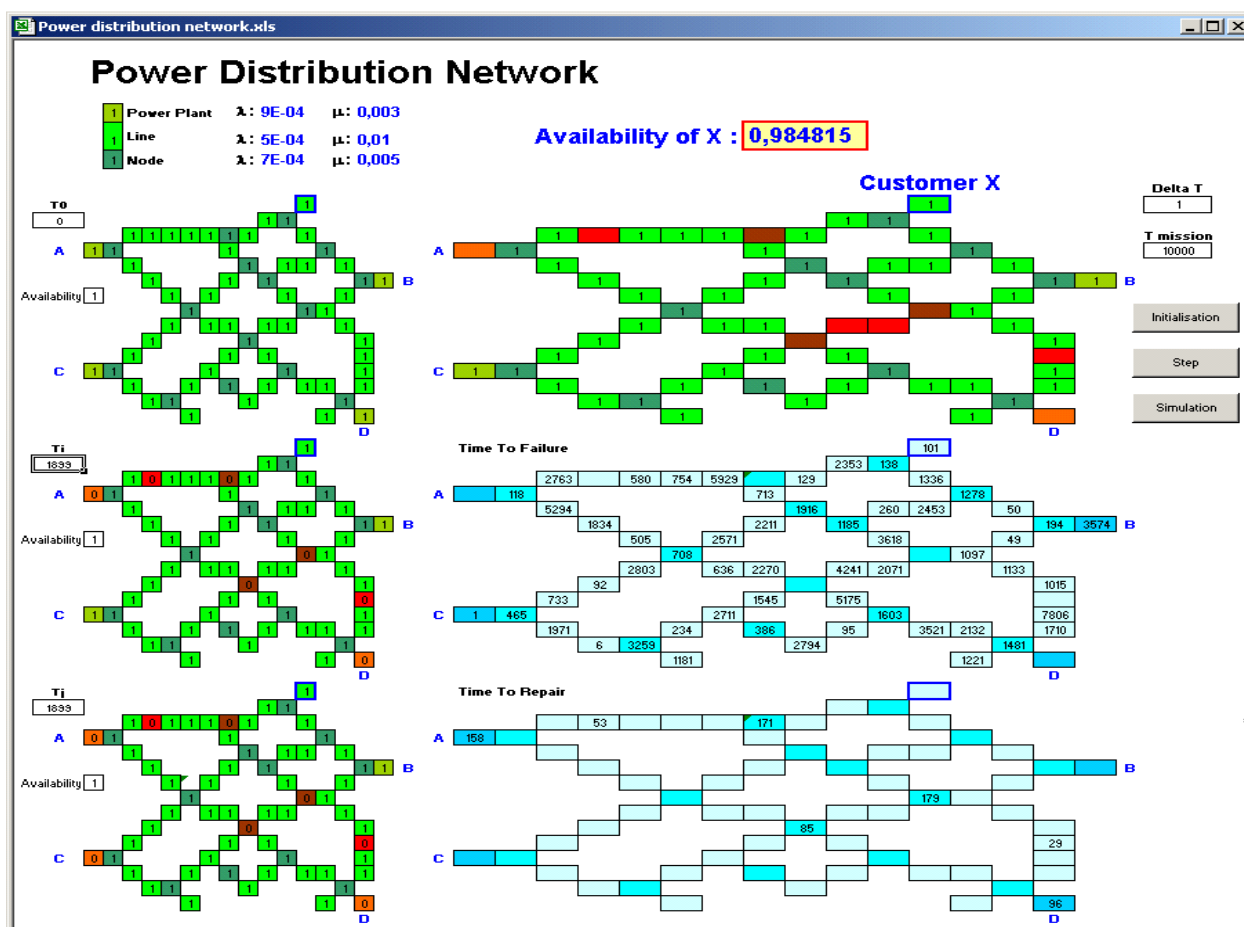


Figure 3. Réseau de distribution électrique

3 - Couplage entre optimisation et simulation

Malgré son intérêt évident dans de nombreux domaines d'ingénierie, le couplage entre optimisation et simulation stochastique, qui consiste à rechercher une configuration optimale de paramètres d'un système à partir des résultats d'une fonction d'évaluation traitée par simulation de Monte-Carlo, s'avère encore confidentiel, car très pénalisant en terme de durée de traitement. En première approximation, le nombre de cas de simulation à réaliser est, en effet, égal au nombre d'évaluations nécessaires à l'optimisation pour assurer la convergence, multiplié par le nombre de cas de simulation requis par la précision recherchée pour chacune des évaluations (soit environ $2000 * 2000 = 4$ millions pour une application typique). Toutefois cette durée peut être sensiblement diminuée par le choix d'une stratégie consistant à faire varier la précision de l'évaluation de chacune des solutions candidates (de 100 à 2000 simulations par exemple), en fonction des résultats d'une évaluation grossière menée préalablement (à 100 simulations par exemple).

Afin que chacune des solutions évaluées soit soumise au même risque de rejet inopportun, la condition de la formule 1 doit être respectée entre les valeurs N_i et N_j du nombre de simulations à réaliser pour évaluer deux candidats i et j , en fonction de la moyenne et de la variance des résultats obtenus à l'issue de l'évaluation grossière menée à N_0 simulations. Cette condition résulte directement de l'application du théorème central limite.

$$N_i/N_j = [(M-m_{j0}) * \sigma_{i0} / (M-m_{i0}) * \sigma_{j0}]^2 \quad [1]$$

Une technique de couplage dite optimale [1], l'algorithme OCBA (Optimal Computing Budget Allocation), applique cette condition pour rechercher une valeur optimale parmi un nombre fini p de candidats. A chaque itération k , celui-ci autorise n nouvelles simulations distribuées selon les ratios indiqués en figure 4, avec \hat{i} la meilleure solution courante trouvée durant l'itération $k-1$; J_i (moyenne) et σ_i (écart type) résultant de l'évaluation de i .

$$\frac{N_{\hat{i}}}{N_i} = \sigma_{\hat{i}} \sqrt{\sum_{j=1, j \neq \hat{i}}^p \frac{1}{\sigma_j^2} \rho_{ij}^2} \quad i \neq \hat{i}$$

$$\rho_{ij} = \left(\frac{\sigma_j / \Delta_j}{\sigma_i / \Delta_i} \right)^2 \quad i, j \in 1, 2, \dots, p \quad i, j \neq \hat{i} \quad \Delta_i = J_{\hat{i}} - J_i$$

Figure 4. Algorithme OCBA

Ce même principe a pu être adapté aux Algorithmes Génétiques, à l'Evolution Différentielle [2] (somme d'un gène de chromosome avec la différence entre les mêmes gènes de deux autres chromosomes) et au Simplexe sur l'outil GENCAB [3] de la société CAB INNOVATION, comme l'illustre la figure 5 :

- Le nombre de simulations réalisées au cours de l'évaluation grossière N_0 et celui nécessaire à la précision requise N étant définis a priori par l'utilisateur, la population initiale de chromosomes (solutions potentielles) est d'abord évaluée à N_0 , puis la meilleure solution parmi celle-ci (en valeur moyenne) est réévaluée à N (par ajout de $N-N_0$ simulations).
- Au cours des différentes boucles de traitement, chaque candidat i résultant d'une mutation, d'une évolution différentielle, d'un croisement ou d'une recherche locale (simplexe) est évalué à N_0 puis réévalué à la valeur N_i obtenue par application de l'algorithme OCBA limité à la valeur N (la réévaluation n'est effective que si $N_i > N_0$).
- La sommation utilisée par l'algorithme OCBA dans le calcul des ratios est mise à jour à chaque évaluation, afin de ne pas devoir réévaluer ultérieurement les anciens candidats (non tous mémorisés), et est réinitialisée à l'émergence de toute meilleure solution qui devient alors la solution optimale courante.

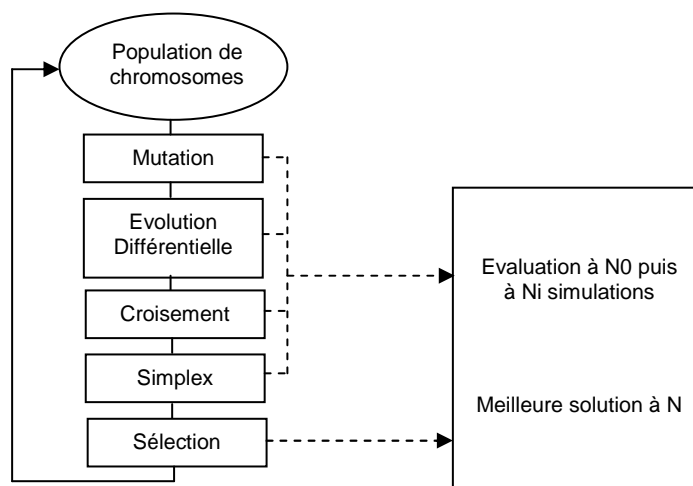


Figure 5. Couplage optimisation / simulation

Testée sur différents cas test, le couplage se révèle d'autant plus efficace que la précision requise est élevée. Il permet ainsi de diminuer les temps de calcul dans un rapport 30 environ dans le cas de 2000 simulations par évaluation fine et 100 simulations par évaluation grossière.

4 - Modèle de simulation récursive et optimisation

L'optimisation peut porter sur un modèle de simulation récursive comme l'illustre la figure 6 qui représente un petit système électrique dont on cherche à optimiser la maintenance.

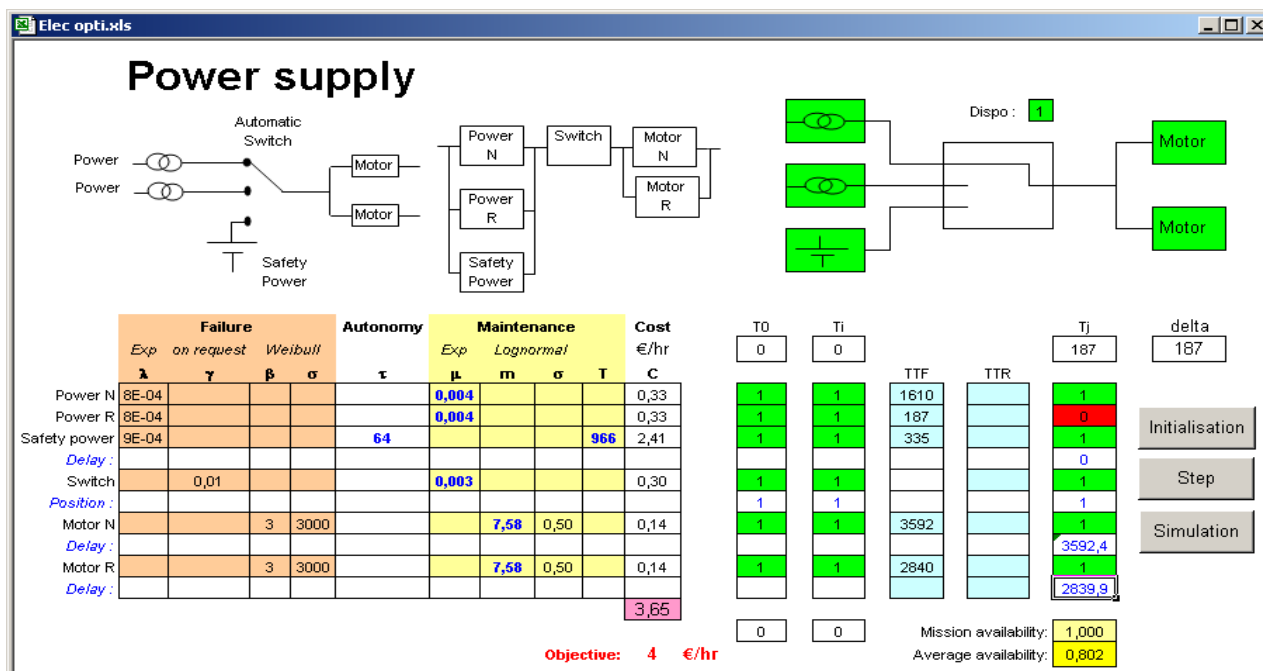


Figure 6. Système électrique

Celui-ci est constitué de deux alimentations en redondance secourues par une batterie d'accumulateurs, d'un commutateur automatique et de deux moteurs en redondance. Il faut noter que ce modèle doit mémoriser certains délais avant transition car le système n'est que partiellement markovien. L'optimisation consiste à rechercher la configuration de 5 paramètres, μ_1 (taux de réparation des alimentations), μ_2 (taux de réparation du commutateur), τ (durée d'autonomie batterie), T (période de maintenance de la batterie) et m (logarithme du MTTR des moteurs), qui maximise la disponibilité du système pendant une durée de 10000 heures, tout en satisfaisant une contrainte de coût moyen horaire (≤ 4 €/heure). Cette optimisation est réalisée en une douzaine d'heures environ au lieu d'une quinzaine de jours sans utiliser la technique de couplage amélioré.

6 – Conclusion

Se présentant comme une alternative à d'autres techniques de modélisation, les modèles de simulation récursive laisse à l'utilisateur une grande liberté dans le choix de la représentation de son système en s'appuyant sur les fonctionnalités d'un tableur largement diffusé. Cette souplesse permet de développer et de valider des modèles de simulation divers qui peuvent se coupler directement à des outils d'optimisation sans conduire à des durées de traitement rédhibitoires. Ils sont utilisés au CNES pour résoudre des problématiques complexes du domaine spatial, telles que l'optimisation de la stratégie de déploiement et de renouvellement d'une constellation de satellites ou celle de l'architecture et des conditions d'exploitation de moyens au sol (stations de réception, centres de contrôle ou d'exploitation, etc.).

Bibliographie

- [1] C. H. Chen, J. Lin, E. Yucesan, and S. E. Chick. (2000) Simulation budget allocation for further enhancing the efficiency of ordinal optimization. Journal of Discrete Event Dynamic Systems : Theory and Application, 10(3) :251-270.
- [2] V. Feoktistov, S. Janaqi (2004) - Evolution différentielle - Une vue d'ensemble, MOSIM 04, NANTES.

[3] A. Cabarbaye (2003), Outil générique d'optimisation dans le domaine discret et/ou continu éventuellement stochastique - ROADEF'03 – Avignon, 2003.