

OPTIMISATION ET SIMULATION STOCHASTIQUE

A. Cabarbaye, J. Faure, R. Lautheret
Centre National d'Etudes Spatiales (CNES)
18, avenue Edouard Belin 31401 Toulouse
prenom.nom@cnes.fr

A. Cabarbaye
CAB INNOVATION
3, rue de la Coquille - 31500 Toulouse
andre.cabarbaye@cabinnovation.fr

Résumé

En s'appuyant sur de récentes innovations algorithmiques et sur un cas d'application réel du domaine spatial, cette communication a pour objet de démontrer qu'il est aujourd'hui possible d'optimiser des systèmes complexes, en jouant simultanément sur tous les paramètres influents. Elle propose une technique efficace de couplage entre optimisation et simulation de type Monte-Carlo qui permet de diminuer très significativement la durée globale des traitements (dans un rapport 30 environ).

Summary

By presenting recent algorithmic innovations and a real application of space field, this communication shows that it is today possible to optimize complex systems, while exploiting all the influential parameters simultaneously. It proposes an effective technique of coupling between optimization and Monte-Carlo simulation tools which allows to decrease very significantly the total duration of the computation process (in a ratio 30 approximately).

Introduction

Ayant au départ pour objet la simple vérification de la conformité à des demandes contractuelles, l'évaluation de Sécurité de Fonctionnement est de plus en plus souvent réalisée à des fins d'optimisation pour diminuer le coût global de possession et d'exploitation des systèmes tout en garantissant à l'utilisateur une certaine qualité de service. Mais dans les faits, cette optimisation reste partielle et se limite le plus souvent à l'étude de sensibilité de quelques paramètres. Outre de possibles résistances organisationnelles nuisant au rapprochement de la fiabilité et de la gestion des coûts dans l'entreprise, cette optimisation rencontre quelques difficultés sur le plan méthodologique.

Bien que les performances des ordinateurs aujourd'hui disponibles rendent accessibles, à tous, les techniques de simulation de Monte-Carlo et d'optimisation stochastique, toutes deux fondées sur la répétition d'un grand nombre de calculs, 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 évaluation traitée par simulation de Monte-Carlo, s'avère difficile voire rédhibitoire, 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).

Cherchant à optimiser les systèmes spatiaux dans leur globalité, et non plus au niveau de leurs seuls composants (satellites, stations de contrôle, stations de réception, centres d'exploitation...), le Centre National d'Etudes Spatiales a été confronté à cette difficulté qui apparaît aujourd'hui surmontée sur le plan méthodologique. En s'appuyant sur un cas d'application réel concernant la définition et la maintenance d'une constellation de satellites, cette communication a pour objet de démontrer qu'il est aujourd'hui possible d'optimiser des systèmes quelconques, dans un contexte industriel et non plus de recherche fondamentale, en jouant simultanément sur tous les paramètres influents.

Méthode

La durée de traitement, induit par le couplage entre optimisation et simulation stochastique, peut être significativement réduite en faisant varier la précision de l'évaluation de chacune des solutions candidates, en fonction des résultats d'une pré-évaluation grossière. Une telle stratégie a été implantée en interface de deux outils génériques d'optimisation et de simulation, fonctionnant sous Excel, de la société CAB INNOVATION.

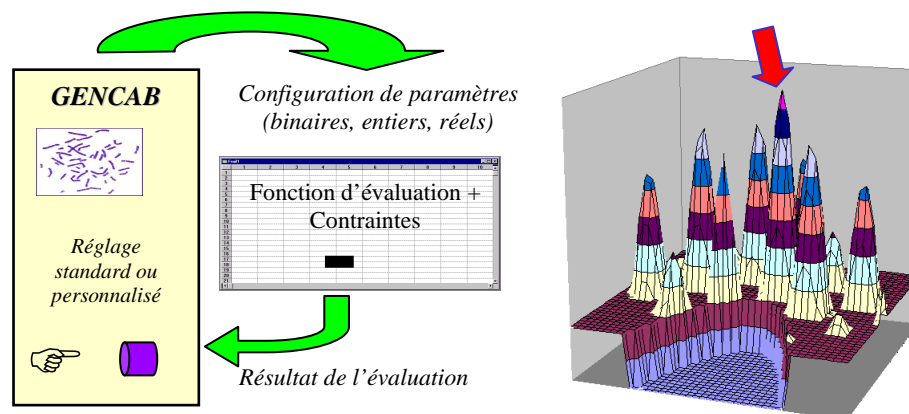


Figure 1. Principe de l'outil d'optimisation GEN CAB

Optimisation et couplage

Basé sur une méthode hybride associant Algorithmes Génétiques [1] & [2] et Simplexe non linéaire (algorithme de Nelder Mead), l'outil d'optimisation GEN CAB [4], dont le principe est illustré en figure 1, s'est récemment enrichi d'un opérateur d'Evolution Différentielle [6]. Constitués de divers paramètres (gènes) de type

réel, entier ou binaire, les chromosomes sont aléatoirement soumis à des mutations, croisements et évolutions différentielles (somme d'un gène de chromosome avec la différence entre les mêmes gènes de deux autres chromosomes) puis, après sélection, les meilleurs éléments de la population bénéficient d'une amélioration au niveau local par un certain nombre de pas de Simplexe. Cette hybridation de différentes techniques de

recherche globale et locale, qui peuvent être éventuellement paramétrées par l'utilisateur, a pour finalité de rendre l'outil robuste à la diversité des problématiques rencontrées. Ainsi l'Evolution Différentielle sera généralement plus efficace que les Algorithmes Génétiques pour traiter des fonctions convexes mais présentera l'inconvénient, pour d'autres, de jouer simultanément sur l'ensemble des gènes.

Illustré par la figure 2, le principe de base du couplage entre optimisation et simulation stochastique consiste à faire varier la précision de l'évaluation de chacune des solutions candidates (de 50 à 2000 simulations de la mission par exemple), en fonction des résultats d'une évaluation grossière menée préalablement (à 50 simulations par exemple).

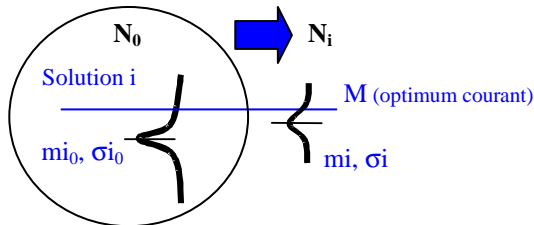


Figure 2. Principe de couplage

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 Ni et Nj 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 N0 simulations. Cette condition résulte directement de l'application du théorème central limite.

$$N_i/N_j = [(M-m_{j_0}) * \sigma_{i_0} / (M-m_{i_0}) * \sigma_{j_0}]^2 \quad [1]$$

Trouvée dans la littérature scientifique [3], une technique de couplage dite optimale, l'algorithme OCBA (Optimal Computing Budget Allocation), applique ce même principe 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 3, avec i la meilleure solution courante trouvée durant l'itération k-1 ; Ji (moyenne) et sigma_i (écart type) résultant de l'évaluation de i.

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

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

$$i, j \neq \hat{i} \quad \Delta_i = J_{\hat{i}} - J_i$$

Figure 3. Algorithme OCBA

Comme l'illustre la figure 4, ce même principe a pu être appliqué aux Algorithmes Génétiques, à l'Evolution Différentielle et au Simplexe après certaines adaptations :

- Le nombre de simulations réalisées au cours de l'évaluation grossière N0 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 N0, puis la meilleure solution parmi celle-ci (en valeur moyenne) est réévaluée à N (par ajout de N-N0 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 N0 puis réévalué à la valeur Ni obtenue par application de l'algorithme OCBA limité à la valeur N (la réévaluation n'est effective que si Ni > N0).
- 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.

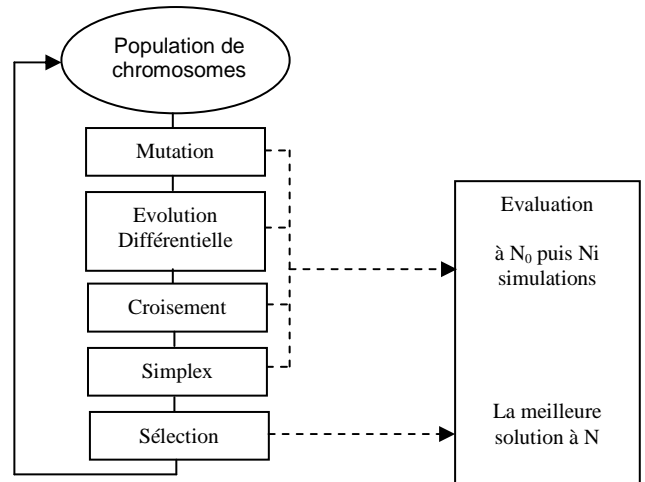


Figure 4. Couplage

Par ailleurs, il est apparu judicieux de ne pas requérir la précision maximale dès les premiers calculs mais de faire croître la précision demandée tout au long du traitement parallèlement à l'amélioration progressive de la population. Aussi, un profil d'évolution du nombre de simulations, allant de N0 à N de la première à la dernière boucle, a également été implanté.

Modèle de simulation récurrente

L'outil de simulation SIMCAB présente une fonctionnalité originale de modélisation récurrente de systèmes à états discrets [6] qui est utilisée pour traiter le cas d'application.

Comme l'illustre la figure 5, cette technique de modélisation consiste à définir une transition générique entre deux états i et j. Chaque simulation effectuée par l'outil consiste à recopier un certain nombre de fois l'état Ej (à tj), défini dans une plage de cellules du tableur, dans l'état Ei (à ti), défini dans une plage similaire, pendant toute la durée d'observation (ou mission), en partant d'un état initial E0 (à t0), défini par ailleurs. L'incrément de temps considéré Delta t entre ti et tj, 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 Tk, correspondant à des changements aléatoires d'état du système ou au franchissement de seuils par des variables continues. Les systèmes considérés peuvent être markoviens ou non-markoviens (avec ou sans conservation de la mémoire des changements d'états successifs, c'est-à-dire de l'occurrence de tous les événements tirés) et éventuellement de type hybride, caractérisés par des dépendances entre variables continues et stochastiques [5].

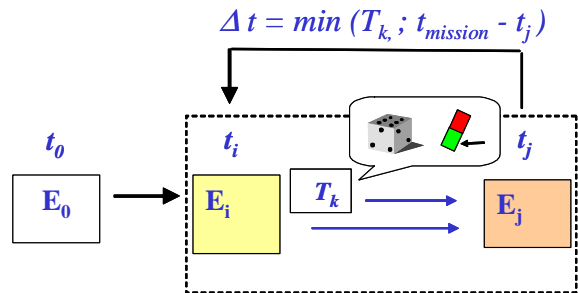


Figure 5. Modèle de simulation récurrente

Cette technique particulière de modélisation fait l'objet d'un article spécifique au lambda mu 15 [8].

Cas d'application

L'application choisie est suffisamment riche pour aboutir à des résultats probants mais cependant pas trop complexe pour ne pas se tromper de problématique en s'approchant des limites propres aux techniques d'optimisation (on ne cherche pas ici à tester les Algorithmes Génétiques).

Concernant la définition d'un nouveau système de satellites d'observation de la terre, l'application a fait l'objet du modèle de simulation récursive, présenté en figure 6, pour évaluer la disponibilité du service offert et les coûts associés, relatifs au déploiement et à la maintenance de la constellation sur une très longue durée (30 ans). Menée selon un critère de coût avec satisfaction d'une contrainte de qualité de service, l'optimisation a porté sur une dizaine de paramètres influents de diverses natures (discret, réel, binaire) concernant tant les caractéristiques des constituants du système que leurs modes d'exploitation.

Le modèle permet de simuler le déploiement et le renouvellement d'une constellation de satellites susceptibles de tomber en panne ($OK \neq 1$) de manière aléatoire, selon une loi exponentielle dans

cette application (λ constant), ou de manière déterministe au terme d'une durée de vie limitée par la capacité en ergol (DV). Constituée nominale d'un certain nombre (7) de satellites simultanément opérationnels ($OP = 1$), la constellation est maintenue par des lancements simples ou multiples de satellites pouvant éventuellement séjourner un certain temps en orbite en tant que rechange. Leur activation est alors déclenchée par le sol mais n'est effective qu'après un certain délai de reconfiguration (1 mois). Un lancement est décidé dès que le nombre de satellites fonctionnant en orbite est inférieur à un nombre minimum requis, en anticipant les fins de vie des satellites pour éviter des interruptions de service prévisibles. Le lancement effectif n'a lieu qu'après une durée de réservation du lanceur et de fabrication des satellites, si des rechanges ne sont pas disponibles au sol, et le tir peut éventuellement conduire à l'échec ($OK \neq \text{vrai}$). L'évaluation porte sur la disponibilité du service rendu par la constellation durant une mission opérationnelle de 30 ans, en termes de nombre moyen de satellites simultanément opérationnels, et sur les coûts associés ramenés à la date du premier lancement par un taux d'intérêt (3 %).

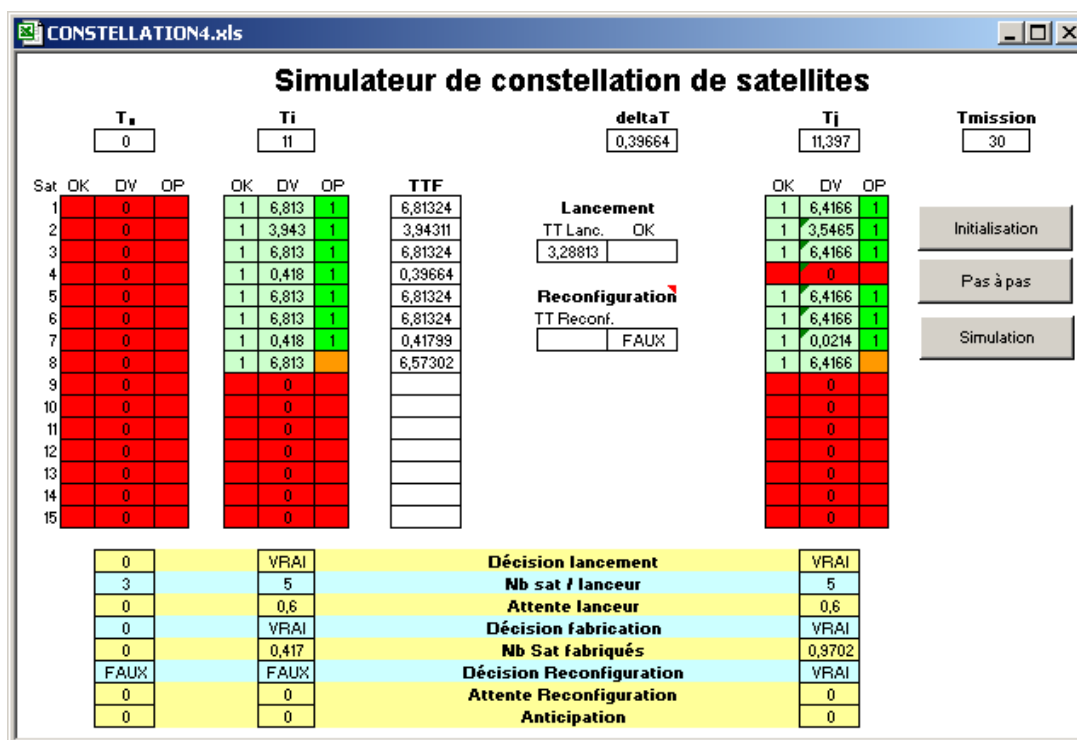


Figure 6. Simulateur de déploiement et de renouvellement de constellation de satellites

Les hypothèses considérées dans ce cas d'application sont détaillées ci-après (bien que le problème soit réel, les données présentées sont fictives pour des raisons de confidentialité).

- La constellation engendre des revenus à l'opérateur qui varie avec le nombre de satellites opérationnels :

$Revenu = (5000 * \text{durée de fonctionnement avec 7 satellites opérationnels} + 2000 * \text{durée de fonctionnement avec 6 satellites opérationnels}) / 30 \text{ ans}$

- Le coût des satellites récurrents dépend de leur durée de vie, de leur fiabilité en fin de vie ($\lambda = -\ln[\text{fiabilité}] / \text{durée de vie}$) et de leur durée de fabrication, selon l'expression suivante :

$Coût \text{ satellite} = 50 + 3 * (\text{Durée de vie} - 5 \text{ ans})^2 + 100 * (\text{Fiabilité} - 0,3) + 10 * (2 \text{ ans} - \text{Durée de fabrication})$

- Un type de lanceur est choisi pour le premier tir et un autre pour l'ensemble des tirs suivants ; chacun ayant ses caractéristiques propres en termes de taux de réussite, de délai de réservation, de capacité maximum et de coût en fonction du nombre de satellites embarqués, comme l'indique la table de la figure 7.

| Type de lanceur | 1 | 2 | 3 | 4 | 5 |
|-------------------------|------|------|------|-----|------|
| Taux de réussite | 0,95 | 0,97 | 0,98 | 0,9 | 0,93 |
| Durée de réservation | 0,6 | 0,5 | 0,3 | 1 | 0,8 |
| Capacité max | 6 | 3 | 2 | 1 | 5 |
| Coût pour : 1 satellite | 20 | 15 | 22 | 12 | 18 |
| 2 satellites | 36 | 20 | 40 | | 34 |
| 3 satellites | 50 | 25 | | | 48 |
| 4 satellites | 60 | | | | 59 |
| 5 satellites | 66 | | | | 65 |
| 6 satellites | 70 | | | | |

Figure 7. Caractéristiques des lanceurs

L'optimisation porte sur la maximisation de l'espérance des gains sur 30 ans (revenus - coût) en jouant sur les paramètres suivants :

- Type (1 à 5) et capacité (1 à 6) du premier lanceur utilisé
- Type (1 à 5) et capacité (1 à 6) des lanceurs suivants
- Durée de vie des satellites (5 à 10 ans)
- Fiabilité des satellites en fin de vie (0,3 à 0,8)
- Durée de fabrication des satellites (0,5 à 2 ans)
- Nombre minimum de satellites en bon fonctionnement requis en orbite (7 à 11)

- Fabrication de rechange au sol après chaque lancement: (vrai ou faux)
- Durée de renouvellement de la constellation : (20 à 30 ans)

Ce cas d'application est évalué en 1 minutes environ pour 100 simulations de la mission de 30 ans avec un Pentium 4. La recherche d'une configuration optimale nécessitant plus de 2000 évaluations (nombre nécessaire pour résoudre des problématiques de complexité similaire par calcul markovien), la durée globale du traitement, sans amélioration du couplage, devrait avoisiner une semaine pour 500 simulations par évaluation et un mois pour 2000 simulations par évaluation, soit des durées non réellement compatibles avec une activité industrielle, d'autant que l'application choisie est de complexité moyenne.

Résultats

Afin de tester l'apport du couplage étudié, l'optimisation a été lancée simultanément sur deux ordinateurs, dont l'un seulement en bénéficiait. Le paramétrage d'usine de l'outil d'optimisation ayant été conservé (population de 50 chromosomes, amélioration à chaque itération du meilleur chromosome par 50 pas de simplexe, etc.), 50 boucles de calcul ont été effectuées par chacun d'eux ; le premier avec un nombre de simulations par évaluation compris entre 50 et une valeur maximale croissant linéairement de 50 à 2000 en fonction du numéro de la boucle en cours et à 500 simulations par évaluation pour le second.

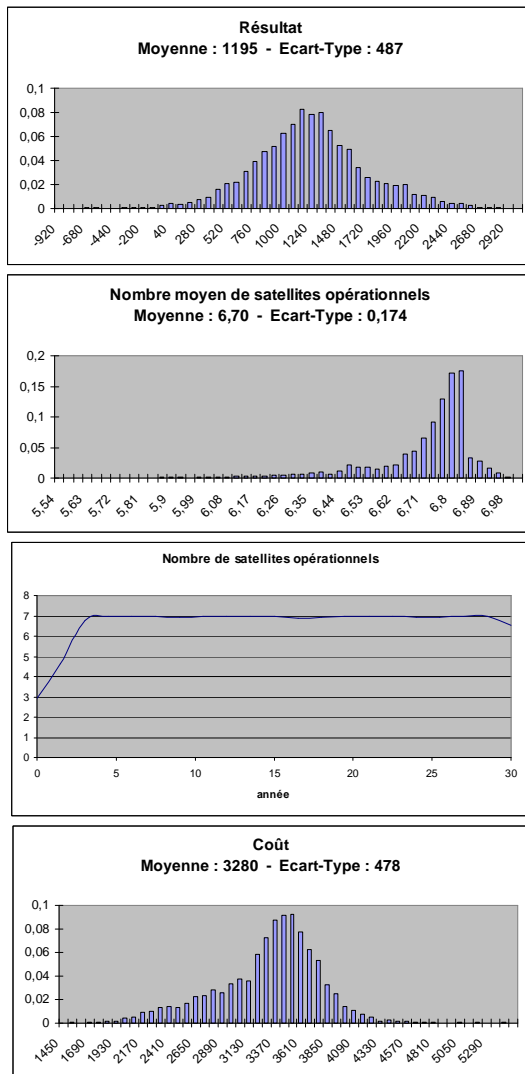


Figure 8. Résultats obtenus par l'ordinateur bénéficiant du couplage amélioré

Après moins de 16 heures de fonctionnement (une nuit), le premier ordinateur a trouvé la solution suivante évaluée à 2000 simulations (figure 8) :

- Lanceur de type 2 avec 3 satellites pour le premier lanceur utilisé
- Lanceur de type 2 avec 1 satellite pour les suivants
- Durée de vie des satellites : 7,79 ans
- Fiabilité des satellites en fin de vie : 0,78
- Durée de fabrication des satellites : 0,51 ans
- Nombre minimum de satellites en bon fonctionnement requis en orbite : 7
- Fabrication de rechange au sol après chaque lancement (mais pas de rechange en orbite)
- Durée de renouvellement de la constellation : 29,57 ans

Le nombre moyen de simulations réalisées pour chaque évaluation étant de 64 (calculé par l'outil tout au long du traitement) au lieu de 2000 (nombre atteint uniquement pour départager les meilleurs candidats), le couplage a permis dans cet exemple de diminuer le temps de calcul dans un rapport de 31,25.

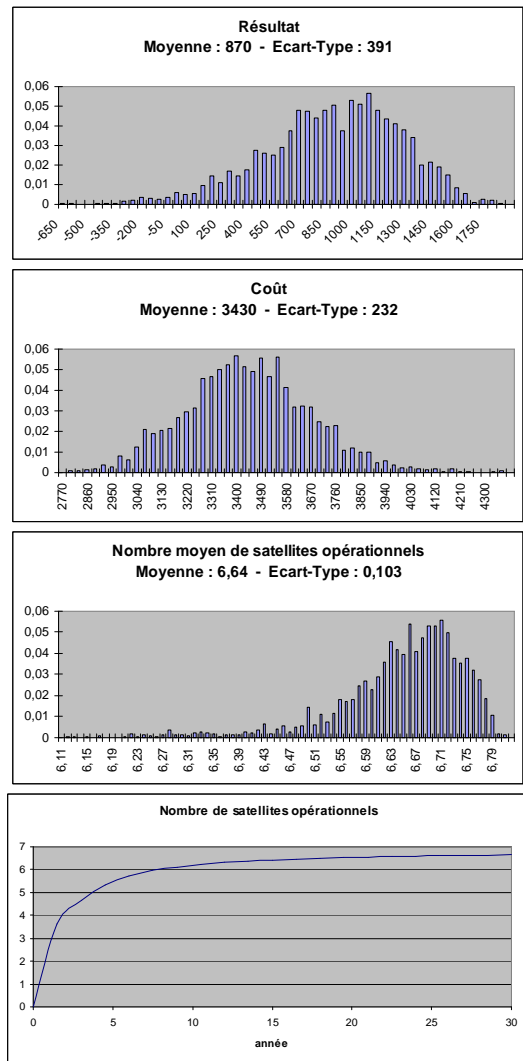


Figure 9. Résultats obtenus par l'ordinateur ne bénéficiant pas du couplage amélioré

Après environ une semaine de calcul le second ordinateur a trouvé une solution voisine de la première bien que moins performante (figure 9). L'évaluation de chaque solution ayant été volontairement limitée à 500 simulations au lieu de 2000, en raison de contraintes matérielles (disponibilité de machine).

- Lanceur de type 3 avec 2 satellites pour le premier
- Lanceur de type 2 avec 1 satellites pour les suivants
- Durée de vie des satellites : 7,82 ans
- Fiabilité des satellites en fin de vie : 0,60
- Durée de fabrication des satellites : 0,55 ans

- Nombre minimum de satellites en bon fonctionnement requis en orbite : 7
- Fabrication de rechange au sol après chaque lancement (mais pas de rechange en orbite)
- Durée de renouvellement de la constellation : 29,91 ans

Il est à noter que le nombre de paramètres considérés (10 dans le cas présenté) n'est pas réellement dimensionnant. En effet, la durée de traitement a été multipliée par 2 environ pour traiter une problématique similaire de 50 paramètres.

Ces tests ont par ailleurs mis en évidence une difficulté propre au couplage entre optimisation et simulation stochastique relative au traitement d'éventuelles contraintes. En effet, outre la maximisation d'une espérance de revenu, les problématiques industrielles se présentent souvent en termes de minimisation d'un coût avec satisfaction d'une contrainte de performance (qualité de service) ou de maximisation d'une performance dans une enveloppe de coût. Le coût et la performance étant souvent antagonistes, l'optimum est généralement situé en bordure de contrainte. Or en raison de la variance des résultats obtenus par simulation, plusieurs évaluations d'une même solution à proximité des bornes peuvent fortement varier au-delà et en deçà de celles-ci. Aussi, la pénalité associée à l'amplitude du dépassement de la contrainte devient un paramètre délicat à régler.

Conclusion

Les résultats obtenus sur un cas d'application réel montrent que l'apport de l'amélioration du couplage entre optimisation et simulation stochastique, présentée dans cet article, peut être très significatif en temps de calcul (durée divisée par 30 environ).

Cet apport, qui a été également testé sur des problématiques fort différentes telles que celle de l'ordonnancement (9), est d'autant plus significatif que la solution optimale courante est meilleure que les candidats évalués. Ainsi une fonction d'évaluation présentant de nombreux pics sera traitée plus efficacement qu'une fonction aux sommets plus arrondis. De même, cet apport est d'autant plus

important que le nombre de simulations requis par la précision recherchée est élevé.

Outre le domaine spatial, ce résultat devrait ouvrir de réelles perspectives pour ce type de traitement dans d'autres domaines d'ingénierie, sans attendre la mise sur le marché d'ordinateurs beaucoup plus performants.

Références

- [1] David E. Goldberg, Algorithmes Génétiques, Exploration optimisation et apprentissage automatique, Addison-Wesley, 1994
- [2] J-M. Renders, Algorithmes génétiques et réseaux de neurone, Hermes, 1995.
- [3] C. H. Chen, J. Lin, E. Yucesan, and S. E. Chick. Simulation budget allocation for further enhancing the efficiency of ordinal optimization. *Journal of Discrete Event Dynamic Systems : Theory and Application*, 2000, 10(3) :251-270.
- [4] A. Cabarbaye, Outil générique d'optimisation dans le domaine discret et/ou continu éventuellement stochastique - ROADEF'03 – Avignon, 2003.
- [5] Pierre Etienne Labeau, Evolution de la modélisation en fiabilité dynamique - Journée « Fiabilité dynamique et simulation hybride », ENSAM, PARIS, 2003.
- [6] V. Feoktistov, S. Janaqi - Evolution différentielle - Une vue d'ensemble, MOSIM 04, NANTES, 2004.
- [7] Cabarbaye A., Laulheret R., Evaluation de la sûreté de fonctionnement des systèmes dynamiques par modélisation récursive, QUALITA 2005, Bordeaux, 2005.
- [8] A. Cabarbaye, Julien Faure, R.Laulheret, Apport des modèles de simulation récursive à l'évaluation des systèmes, $\lambda\mu 15$, Lille, 2006.
- [9] A. Cabarbaye, Julien Faure, R.Laulheret, Risques projet et décision dans l'incertain, $\lambda\mu 15$, Lille, 2006.