

Ordonnancement robuste et prise de décision dans l'incertain

A. Cabarbaye^{1&2} and J. Faure² and R. Laulheret²

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

2. Centre National d'Etudes Spatiales (CNES)
18, avenue Edouard Belin - 31401 Toulouse
andre.cabarbaye@cnes.fr
julien.faure@cnes.fr
roland.laulheret@cnes.f

Mots-clefs : Ordonnancement robuste, aide à la décision, Optimisation Stochastique

1 Introduction

Dans un environnement généralement moins déterministe que celui considéré dans les modèles classiques de résolution [1], l'ordonnancement robuste devient une problématique prégnante [2][3][4][5]. Cette robustesse peut être acquise par la régénération en ligne de l'ordonnancement quand survient l'aléa, même si celle-ci est alors souvent contrainte par des aspects contractuels entre partenaires multiples, ou recherchée a priori par une approche dite « proactive » consistant à élaborer un ordonnancement intrinsèquement robuste à l'ensemble des aléas préalablement identifiés au cours d'analyses de « risques projets » [6]. Mais quand l'incertitude est généralisée, l'ordonnancement n'a plus grand sens et se limite alors à de la prise de décision dans l'incertain, concernant la réalisation d'un nombre restreint de tâches censées nous rapprocher au mieux d'un objectif recherché.

L'objet de cette communication porte sur une technique d'ordonnancement adaptée à la prise en compte de l'aléa. Basée sur une méthode d'optimisation stochastique, elle permet de générer des ordonnancements robustes et cherche à couvrir la problématique de la prise de décision dans l'incertain.

2 Présentation succincte du formalisme et de l'outil utilisé

Cette technique originale a été implantée dans un outil d'ordonnancement (Cabplan), développé par la société CAB INNOVATION, qui est basé sur une méthode d'optimisation hybride associant Algorithmes Génétiques [7][8][9], Evolution Différentielle [10], et Simplexe non linéaire (algorithme de Nelder Mead). Cette association de différentes techniques de recherche globale et locale, qui peuvent être paramétrées par l'utilisateur, a pour finalité de rendre la recherche plus efficace et d'apporter à l'outil une certaine robustesse face à la diversité des problèmes rencontrés.

Défini sous la forme d'une feuille de tableur Microsoft Excel, comme celle présentée en figure 1, l'ordonnancement peut être généré selon différents critères tels que la maximisation des revenus à échéance (gains moins coûts associés à l'ordonnancement ramené à T0 par un taux d'intérêt) tout en respectant des contraintes de précédence entre tâches, de ressources partagées (capables de mener un nombre limité de tâches simultanément) ou de dates de fin de tâche au plus tard. Ainsi dans l'exemple de la figure 1, la tâche 2 ne peut débuter qu'après la réalisation complète des tâches 1 et 3 et ne peut pas être concomitante avec plus de 2 tâches parmi les tâches 1, 5, 20, 12, 15 et 9 pour lesquelles certaines ressources sont partagées. Ces ressources n'ont pas besoin d'être explicitées et chaque condition peut n'être exprimée qu'une fois (si la tâche i ne peut pas se dérouler pendant les tâches j, la tâche j ne peut pas se dérouler pendant la tâche i). Cette formulation relativement simple permet de définir la plupart des contraintes résultant de l'utilisation de plusieurs ressources affectées à une même tâche (la tâche 20 partage une ressource avec la tâche 1 et une autre avec la tâche 5). Dans les cas où cela ne s'avère pas possible, quand par exemple trois ressources ne sont utilisées que par deux tâches, une dichotomie des tâches peut être effectuée suivant les ressources utilisées. Enfin dans cet exemple, la tâche 3 doit être impérativement achevée à la date $t = 200$ jours après T0, date du commencement de la première tâche de l'ordonnancement. Le début de chacune des tâches peut être choisi au plus tôt ou au plus tard, à date fixe, ou laissé à l'appréciation de l'outil, en tant que paramètre de l'optimisation. Le nombre de tâches de l'ordonnancement n'est pas limité (26 dans l'exemple), mais celui

des tâches laissées à l'appréciation de l'outil (17 dans l'exemple) ne peut augmenter indéfiniment en raison des limites mêmes des techniques d'optimisation employées (limité à 150 par l'outil).

Tâches		Antériorité				Ressources communes				Achèvement au plus tard		Durée		Début à date fixe		Au plus tôt		Au plus tard		Début		Fin		Coût en début de		Coût / durée		Coût total à T0		Gain en fin de		Gain total à T0	
N°	Nom	N° Tache	+	C	N° Tache	+	Nb max	C	Date(jr)	C	(jr)	marge	Date(jr)		Date(jr)	Date(jr)	Date(jr)	Date(jr)	Date(jr)	Date(jr)	Date(jr)	Date(jr)	(K Euro)	(K Euro)	(K Euro)	(K Euro)	(K Euro)	(K Euro)	(K Euro)	(K Euro)			
1	Tâche 1	10																															
2	Tâche 2	1	3																														
3	Tâche 3																																
4	Tâche 4	3	6																														
5	Tâche 5																																
6	Tâche 6	5																															
7	Tâche 7																																
8	Tâche 8	7																															
9	Tâche 9																																
10	Tâche 10																																
11	Tâche 11																																
12	Tâche 12																																
13	Tâche 13	9																															
14	Tâche 14																																
15	Tâche 15	13																															
16	Tâche 16	2	5																														
17	Tâche 17	2	1																														
18	Tâche 18	16																															
19	Tâche 19	18	15																														
20	Tâche 20																																
21	Tâche 21	3	6																														
22	Tâche 22	3																															
23	Tâche 23	22	21																														
24	Tâche 24	23																															
25	Tâche 25	24																															
26	Tâche 26	25																															
+	Fin																																

Fin de la dernière tâche : 1245 Coût : 24770 Gain : 30776

Revenu global à T0 : 6006 Taux d'intérêt : 5,50%

Figure 1. Formalisme utilisé par l'outil CABPLAN

Outre la satisfaction des contraintes, l'optimisation peut porter sur la durée globale de l'ordonnancement ou sur un critère plus pertinent tel que la maximisation du revenu ramené à T0 par un taux d'intérêt. Aussi peut-on affecter à chaque tâche un coût initial (un échéancier de coûts d'approvisionnement pouvant être ramené à un coût unique au démarrage) ainsi qu'un coût proportionnel à la durée. De même, l'achèvement d'une tâche ou de l'ordonnancement complet peut apporter un gain ; les gains et les coûts étant ramenés à T0 par application du taux d'intérêt. Traitées par l'outil, les contraintes non satisfaites de la table disparaissent progressivement et l'ordonnancement évolue selon le critère choisi. Une fonction de dessin permet de générer le diagramme PERT correspondant à la table afin d'en faciliter la lecture des résultats (figure 2).

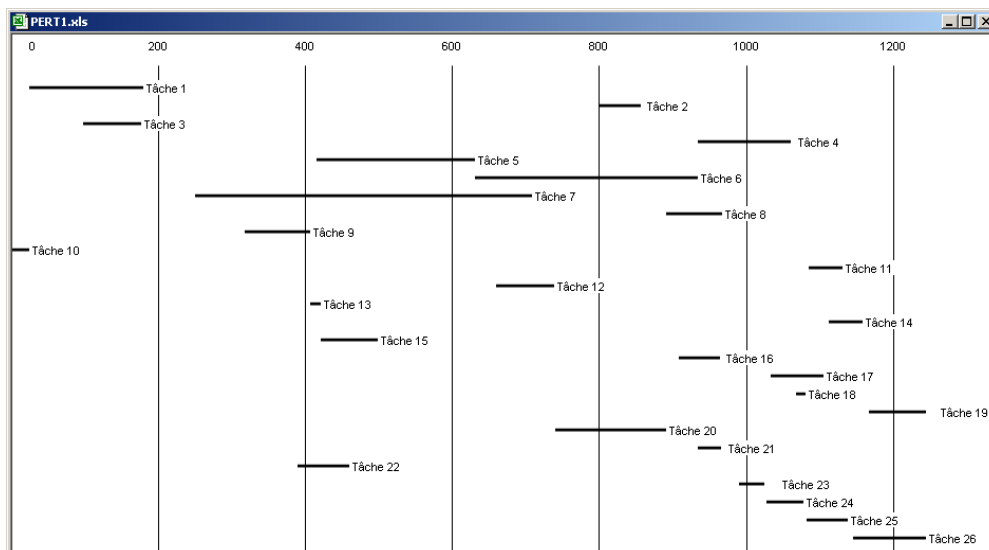


Figure 2. Diagramme PERT correspondant à la table

La durée de traitement de l'exemple présenté est de l'ordre de 15 minutes avec un Pentium 4, soit 2000 boucles de calcul dans la configuration de réglage d'usine des algorithmes de l'outil ; une durée de convergence de l'ordre de 30 minutes étant nécessaire pour traiter un exemple similaire mais plus conséquent de 60 tâches dont 52 à optimiser. Il est à noter que la recherche de l'optimum s'avère plus efficace avec le critère de maximisation du revenu qu'avec celui de la durée globale de l'ordonnancement. En effet, chaque variation de date de début de tâche influence le résultat final avec ce premier critère, ce qui n'est pas le cas avec le second (limité aux seules tâches placées momentanément sur le chemin critique). Le taux d'intérêt à cependant des effets antagonistes sur les tâches qui ont tendance à reculer en raison de leurs coûts tout en étant amenées à avancer par les gains qu'elles génèrent collectivement.

A partir du diagramme « PERT » obtenu, l'impact d'une dérive unitaire sur la durée globale du projet est évalué par l'outil, en remplaçant les dates de début de tâches par un majorant entre les dates prévues et celles résultant de la satisfaction des contraintes de précédence. La modification de la durée ou d'un coût relatif à une tâche se traduit alors par une modification du PERT et des coûts associés, avec l'apparition d'éventuelles contraintes de ressources ou de date au plus tard non satisfaites. Ainsi peut-on connaître l'impact d'un aléa, en termes de coût et délai sur la totalité du développement, et juger de l'opportunité d'éventuelles actions de sécurisation (séduisantes au niveau élémentaire, certaines d'entre elles pouvant s'avérer globalement contre productives). On peut également régénérer le PERT en ligne, quand l'aléa survient, pour tenter de minimiser ses conséquences par un réajustement des tâches non encore réalisées. Une simulation de type Monte-Carlo du diagramme PERT peut enfin être réalisée pour obtenir des résultats prévisionnels, financiers ou calendaires, sous la forme de distributions statistiques (figure 3). Identifiés au cours d'analyses de risques, les aléas, relatifs à des durées ou coûts de tâches élémentaires, sont alors caractérisés par des lois de probabilité simples (uniforme, triangulaire, normale), ou plus complexes si nécessaire.

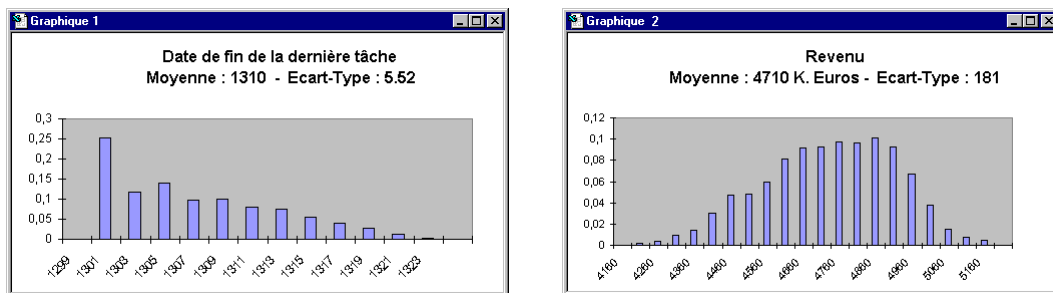


Figure 3. Distribution statistique de résultats

3 Ordonnancement robuste

Outre la régénération ou la simulation d'un diagramme PERT soumis à aléa, un ordonnancement robuste peut résulter d'une optimisation réalisée à partir de résultats de simulation de type Monte-Carlo dans laquelle les risques identifiés au cours des analyses de risques « projets », relatifs à des durées ou coûts de tâches élémentaires, sont caractérisés par des lois de probabilité. Différent de celui obtenu hors aléa et naturellement plus margé, l'ordonnancement devient ainsi robuste car il maximise, en moyenne, le critère choisi pour l'ensemble des cas aléatoires simulés.

La durée du traitement est alors beaucoup plus longue car, en première approximation, multipliée par le nombre de simulations nécessaires à l'évaluation de chacune des solutions. C'est pourquoi une technique originale de couplage entre optimisation et évaluation stochastique a été introduite pour diminuer la durée globale des traitements. Décrite dans un autre article du ROADEF [12], cette technique relativement efficace (diminution des temps de calcul dans un rapport supérieur à 10 selon la problématique) consiste à faire varier la précision de l'évaluation de chacune des solutions candidates, en fonction des résultats d'une évaluation grossière menée préalablement.

A titre de démonstration, l'ordonnancement présenté précédemment a été modifié en remplaçant les paramètres indiqués en figure 4 par des variables aléatoires centrées sur les anciennes valeurs de ces

paramètres. Les résultats de cet ordonnancement, optimisé dans le cas déterministe, sont alors obtenus sous forme de distribution par simulation de Monte-Carlo (2000 cas dans l'exemple).

- Durée tâche 1 : Loi Normale ($m = 150, \sigma = 30$)*
- Durée tâche 4 : Loi Normale ($m = 125, \sigma = 10$)*
- Durée tâche 5 : Loi Normale ($m = 215, \sigma = 40$)*
- Durée tâche 6 : Loi Triangulaire (probable = 302, min = 252, max = 352)*
- Durée tâche 7 : Loi Normale ($m = 459, \sigma = 50$)*
- Durée tâche 10 : Loi uniforme (min = 10, max = 40,*
- Durée tâche 11 : Loi Normale ($m = 46, \sigma = 5$)*
- Durée tâche 16 : Loi Normale ($m = 54, \sigma = 15$)*
- Durée tâche 18 : Loi uniforme (min = 2, max = 22)*
- Durée tâche 21 : Loi Normale ($m = 38.4, \sigma = 8$)*
- Durée tâche 23 : Loi Normale ($m = 31, \sigma = 8$)*
- Durée tâche 24 : Loi Normale ($m = 49, \sigma = 10$)*
- Coût début tâche 15 : Loi Normale ($m = 89, \sigma = 15$)*
- Coût / durée tâche 5 : Loi uniforme (min = 20, max = 30)*
- Coût / durée tâche 18 : Loi Normale ($m = 12, \sigma = 3$)*

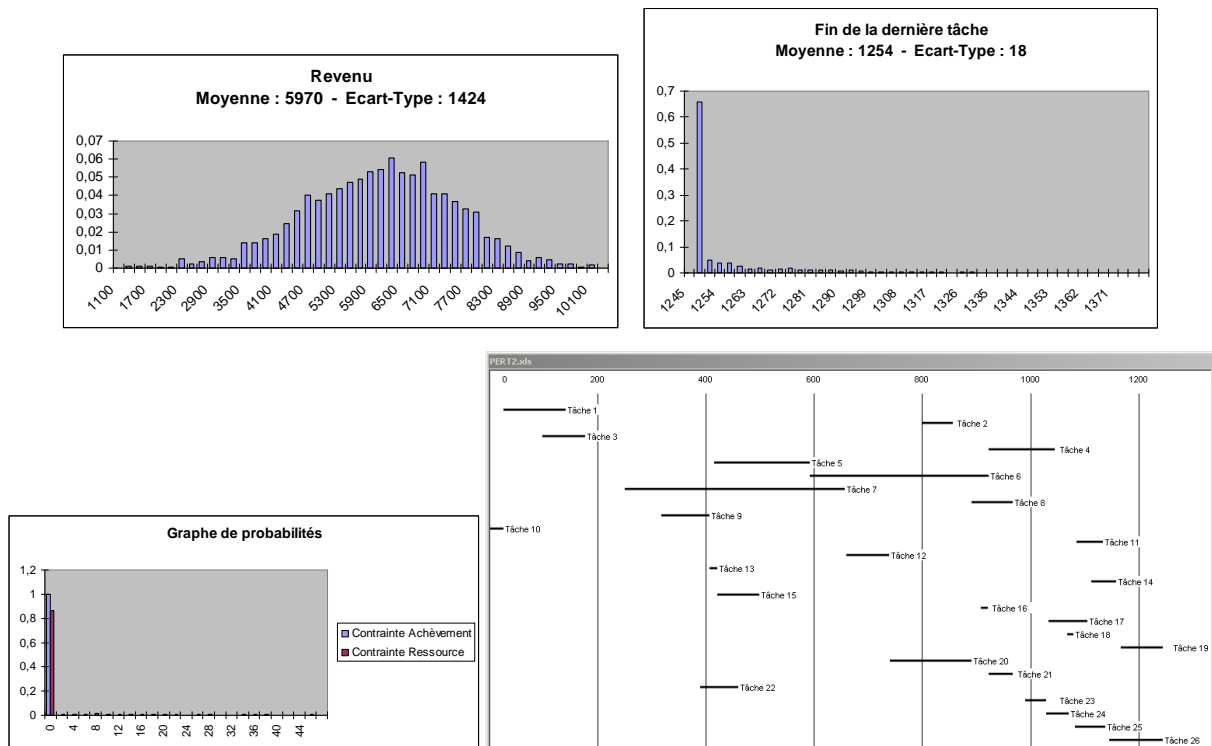


Figure 4. Réponse à des variables aléatoires de l'ordonnancement préalablement obtenu

Une nouvelle optimisation, dont les résultats sont présentés en figure 5, a été réalisée à partir de résultats de simulation de Monte-Carlo en cherchant à maximiser, outre la satisfaction de l'ensemble des contraintes, la moyenne des revenus obtenus pour chacune des configurations. En termes de calcul, les optimisations ont été menées à partir des résultats de 50 simulations pour les évaluations grossières et jusqu'à 2000 simulations pour les évaluations fines, selon la qualité des candidats. Chaque solution ayant fait l'objet de 74 simulations en moyenne (résultat fourni par l'outil), la durée globale du traitement (18 heures environ) a donc été divisée par 27 par rapport à celle d'une optimisation menée uniformément à 2000 simulations.

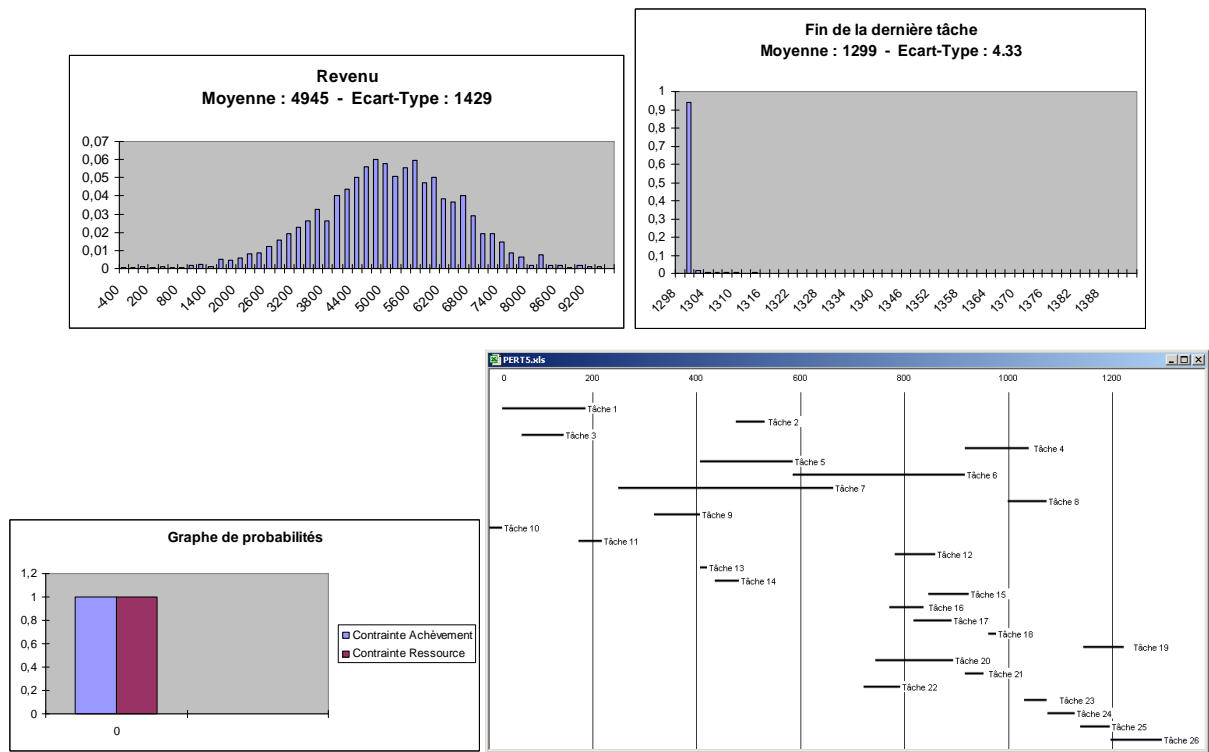


Figure 5. Ordonnancement obtenu par maximisation de l'espérance des revenus

Maximiser l'espérance de revenu n'est pas forcément le critère le plus pertinent pour un responsable de projet, car le revenu peut être caractérisé par une très grande variance comme le montre l'exemple choisi. Aussi nous est-il paru pertinent de chercher à garantir un certain niveau de revenu en choisissant un critère, qui ne remette pas en cause la technique de couplage entre optimisation et simulation. Nous avons choisi comme nouveau critère de minimiser l'écart entre le revenu et le revenu moyen espéré quand ce dernier n'est pas atteint, soit l'expression $\text{Min} [\text{revenu} - 5000 ; 0]$ dans l'exemple présenté en figure 6.

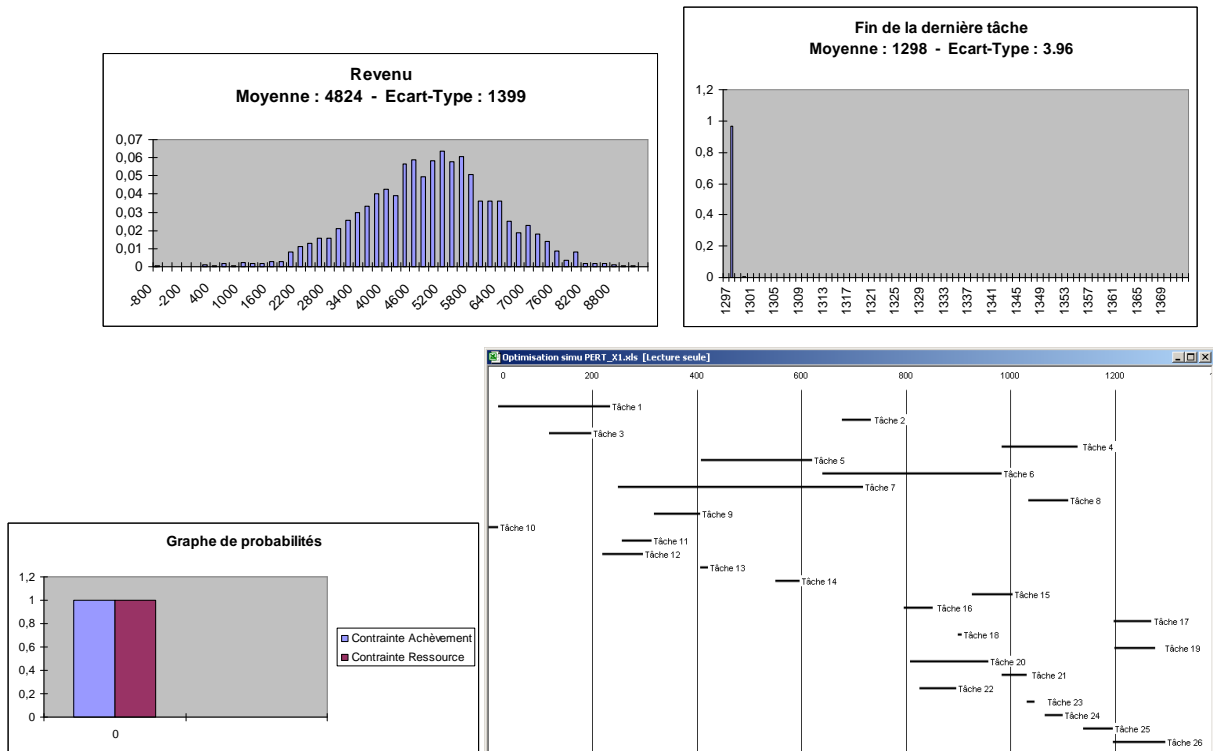


Figure 6. Ordonnancement cherchant à garantir un certain niveau de revenu

En première analyse, l'exemple choisi était apparemment trop simple (limité à 17 paramètres) pour être réellement probant, car l'espérance de revenu du premier ordonnancement, bâti à partir d'hypothèses déterministes, apparaît sensiblement la plus forte. Ce dernier se révèle toutefois moins robuste en termes de variance des dates de fin de dernière tâche, et surtout en termes de satisfaction des contraintes de ressources. En effet 15 % environ des cas simulés conduisent à des dépassements de capacité, alors que cette contrainte est toujours satisfaite dans les deux autres cas. Aussi, la comparaison des revenus réels est-elle particulièrement difficile. C'est pourquoi il est envisagé, dans une future version de l'outil, de transformer tout ou partie des contraintes de ressource en contraintes de précedence dans le fichier de simulation, afin de mieux en tenir compte dans l'estimation des revenus.

Par ailleurs, le critère cherchant à garantir un certain niveau de revenu n'apparaît pas beaucoup plus efficace, dans cet exemple, que la simple maximisation de l'espérance de revenu. Toutefois, il est nécessaire de rappeler ici que les techniques d'optimisation stochastique utilisées ne permettent pas de garantir l'optimalité des solutions trouvées, même si leur efficacité n'est plus à démontrer. On se gardera donc de généraliser un résultat à partir d'un seul exemple ou d'une seule optimisation ; une excellente solution pouvant toujours être trouvée par hasard et inversement la convergence ralentie par blocage momentané dans un optimum local.

4 Prise de décision dans l'incertain

Cette nouvelle problématique reste encore en phase de formalisation et de maquettage. L'idée sous-jacente est de pouvoir attribuer à chaque tâche un poids par rapport à un objectif recherché. Ces tâches sont toujours soumises à des contraintes de précedence ou de ressources partagées et leur achèvement peut conditionner des événements aléatoires, dont l'occurrence est elle même nécessaire au démarrage de nouvelles tâches. A titre d'exemple, la figure 7 illustre le cas d'une implantation d'usine. Plusieurs sites peuvent être envisagés faisant chacun l'objet d'études nécessaires à la délivrance du permis de construire. La construction de l'usine pourra effectivement démarrer dès la délivrance d'une autorisation administrative pour l'un des sites.

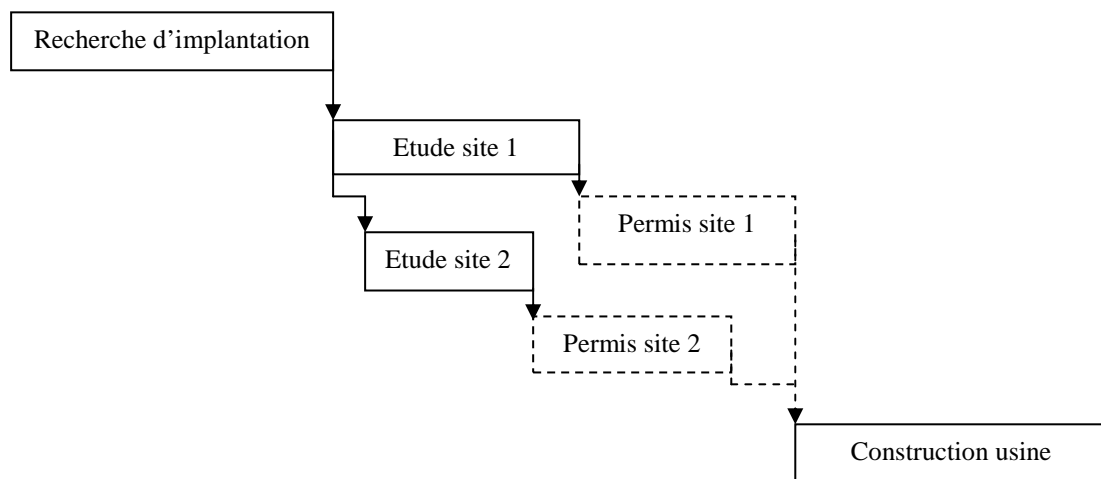


Figure 7. Exemple de problématique de prise de décision dans l'incertain

Ce type de problématique peut se formaliser, sur une feuille de tableur, de la même manière que celle proposée pour un ordonnancement classique. Les événements aléatoires (délivrance des permis de construire) sont considérés comme des tâches à durée éventuellement infinie pour traiter les cas d'absence d'occurrence (rejet du permis). Outre des conditions de précedence (constitution du dossier de permis de construire), les lois de probabilité caractérisant la durée de réalisation de ces événements peuvent être éventuellement influencées (dépendances stochastiques) par la réalisation de certaines tâches antérieures

annexes (étude d'impact...). Des conditions logiques (OU) sur des conditions de précédence doivent être également considérées (construction d'usine dès la délivrance de l'un des permis de construire).

Toutefois cette ébauche de formalisation de la problématique de prise de décision dans l'incertain, qui peut d'hors et déjà être traitée au cas par cas, devrait être validée auprès d'utilisateurs potentiels avant d'aboutir au développement d'une nouvelle fonctionnalité proposée par l'outil.

5 Conclusion

Même si le choix d'un cas test plus conséquent et l'emploi de variables aléatoires plus perturbantes pour l'ordonnancement déterministe généré initialement aurait sans doute conduit à une démonstration plus probante, les résultats obtenus montrent la capacité de la méthode présentée à générer, en ligne ou hors ligne, un ordonnancement robuste aux aléas préalablement identifiés. Elle bénéficie d'une technique de couplage très performante entre optimisation et simulation de Monte-carlo, qui rend la durée de ce type de traitement non rédhibitoire, ainsi que d'une grande souplesse d'adaptation aux problématiques rencontrées, formalisées sous la forme d'une simple feuille de tableur.

Ainsi la problématique plus générale de la prise de décision dans l'incertain pourrait être traitée de la même manière, dans la mesure où sa formalisation répond à un besoin largement partagé.

Références

- [1] Pinedo M.L. (1995), Scheduling : theory, algorithms and systems, Prentice Hall, Englewood Cliffs, New Jersey,
- [2] Lereno E., Morello B., Baptiste P. (2001), Système d'aide au paramétrage d'un logiciel l'ordonnancement, 3e Conférence Francophone de Modélisation et Simulation (MOSIM'01), Troyes
- [3] Briand C., Hoang Trung La, Erschler J. (2003), Une approche pour l'ordonnancement robuste de tâches sur une machine, 4e Conférence Francophone de Modélisation et Simulation - (MOSIM'03) Toulouse
- [4] Esswein C., Artigues C., Billaut J.-C. (2003). Maximiser la flexibilité sur une machine et dans un job shop : ordonnancements de groupes - ROADEF'03 - Avignon,
- [5] Vidal T., Bidot J., Beck J.-C., Laborie P. (2003). Gestion de projets sous incertitudes : un modèle de génération de plans flexibles en horizon glissant - ROADEF'03 - Avignon,
- [6] Cabarbaye A., Laulheret R. (2005). Ordonnancement robuste aux risques projets - QUALITA 2005 - Bordeaux
- [7] David E. Goldberg (1994), Algorithmes Génétiques, Exploration optimisation et apprentissage automatique, Addison-Wesley,
- [8] J-M. Renders (1995), Algorithmes génétiques et réseaux de neurone, Hermes,
- [9] A. Cabarbaye (2003). Outil générique d'optimisation dans le domaine discret et/ou continu éventuellement stochastique - ROADEF'03 - Avignon,
- [10] V. Feoktistov, S. Janaqi (2004) - Evolution différentielle - Une vue d'ensemble - MOSIM 04, NANTES
- [11] Blazewicz J. , Ecker K. , Schmidt G. , Weglarz J. , Scheluding in Computer and Manufacturing Systems, Springer Verlag, Berlin Heildelberg, 1993.
- [12] A. Cabarbaye, J.Faure, R. Laulheret (2006). Couplage entre optimisation et simulation stochastique - ROADEF'06 - Lille,