

Monte Carlo Tree Search appliqué à la gestion de stocks

Adrien Couëtoux^{1,2} Olivier Teytaud¹ Nicolas Bonnard² Nicolas Omont²
Olivier Ratier²

¹ TAO-INRIA, LRI, CNRS UMR 8623,
Université Paris-Sud, Orsay, France

² Artelys, 12 rue du Quatre Septembre Paris, France

Mots-clés : *Processus de décision markovien, Fouille d'arbres par Monte-Carlo, Programmation stochastique, Gestion de stocks*

Les perspectives couplées de l'épuisement des ressources d'énergie non renouvelables et de la hausse des températures liée à l'activité humaine ont placé les systèmes électriques sous les feux des projecteurs. On prévoit que l'utilisation massive des systèmes d'information pour gérer les systèmes électriques permettra simultanément de réaliser des économies d'énergie et d'intégrer des sources d'énergies à la fois renouvelables et à faible impact sur le climat telles que le vent et le soleil. Un des principaux enjeux posé par leur intégration est le caractère non contrôlé de la puissance produite. En effet, tandis que la plupart des centrales actuelles adaptent instantanément le niveau de production à celui de la consommation, cela n'est pas le cas avec le vent ou le soleil. Dans ce cadre, les moyens de stockages de l'électricité (à proprement parler de l'énergie sous une forme facilement convertible en électricité) prennent donc toute leur importance afin d'ajuster dans le temps la production et la consommation. Ceux-ci peuvent être divers : eau dans le réservoir amont d'une installation hydroélectrique, air comprimé, volant d'inertie, mais aussi capacité thermique des habitations, des congélateurs ou batteries des futurs véhicules électriques.

En l'état actuel, les meilleures techniques de gestion de stocks ne passent pas à l'échelle. Ainsi, en matière d'hydroélectricité, on peut considérer que l'application de la programmation dynamique duale stochastique à 5 stocks distincts représentant de manière agrégée les barrages des différentes régions climatiques brésiliennes [1] est proche de l'état de l'art. Il est donc nécessaire de développer de nouvelles méthodes afin de répondre aux enjeux en matière de stockage d'électricité. Les méthodes de fouille d'arbres par Monte-Carlo (Monte-Carlo Tree Search, MCTS) permettent d'optimiser des processus décisionnels markoviens, cadre assez général de modélisation de processus décisionnels allant du jeu de go aux stocks d'électricité. Elles se sont révélées particulièrement efficace dans ce premier domaine [3] en permettant de nets progrès, jusqu'à gagner des matchs face à des joueurs professionnels [4]. Ces méthodes craignent peu l'explosion combinatoire liée au nombre de stocks, d'autant qu'elles sont aisément parallélisables, et ne nécessitent aucune hypothèse de convexité.

Nous considérons un problème où le preneur de décision est placé dans un état initial avec un horizon temporel au delà duquel les évènements ne sont pas pris en compte. Il doit choisir une décision parmi un ensemble de décisions faisables donné. Le bon fonctionnement du MCTS demande également l'accès à une fonction de transition, qui simule la réaction de l'environnement à une action donnée (c'est-à-dire une fonction qui, prenant comme arguments un état et une décision, retourne l'état résultant et le coût associé). Le Monte Carlo Tree Search procède de la manière suivante : à partir d'un état initial, on évalue successivement plusieurs décisions possibles. On construit ainsi un arbre, dont les noeuds représentent les états, et les arcs les décisions. La boucle suivie par l'algorithme est illustrée par la figure (1). On commence cette boucle dans l'état initial (donc dans la racine de l'arbre). On doit à chaque fois choisir entre sélectionner une décision déjà explorée (et donc se déplacer dans le fils correspondant) et créer un nouveau noeud (i.e. tester une nouvelle décision). L'ajout d'un noeud à l'arbre met fin à la phase de sélection et est appelé phase d'expansion. Si le noeud ajouté n'atteint pas l'horizon temporel, on simule de manière aléatoire les évènements de ce noeud à

l'horizon. On calcule alors le coût associé, et on propage cette information sur la série d'états et de décisions qui ont été parcourus durant cette dernière boucle. On réitère ce processus jusqu'à épuiser le temps alloué à la prise de décision, et on sélectionne alors la décision qui a été simulée le plus grand nombre de fois. De récents travaux ont étendu cet algorithme aux espaces continus [5].

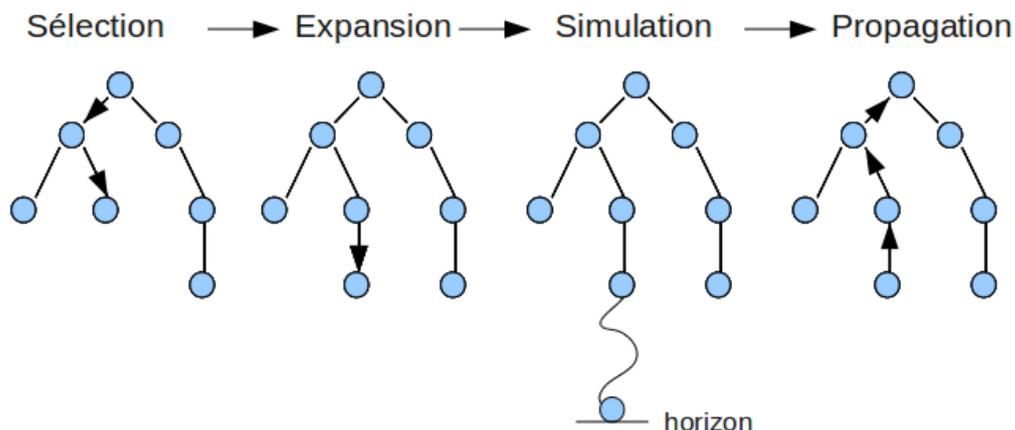


FIG. 1 – Monte Carlo Tree Search loop

Afin d'illustrer ces méthodes, nous les appliquons à un problème jouet constitués de plusieurs stocks d'eau enchaînés le long d'une vallée hydroélectrique. Dans chacun de ces stocks, le flux entrant est constitué du flux arrivant du stock amont et d'un flux externe qui suit une loi log-normale. L'exploitant doit satisfaire une demande donnée, à l'aide des stocks hydroélectriques et d'une centrale thermique dont les coûts de fonctionnement sont supposés linéaires par morceaux. On compare ces méthodes avec les méthodes classiques de recherche opérationnelle.

Références

- [1] M.E.P. Maceira, V.S. Duarte, D.D.J. Penna, L.A.M. Moraes, A.C.G. Melo. Ten years of application of stochastic dual dynamic programming in official and agent studies in Brazil - Description of the new wave program. *16th PSCC, Glasgow, Scotland*, July 14-18, 2008.
- [2] Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo Planning. *ECML, Number 4212 in LNCS* :282–293, 2006
- [3] S. Gelly and D. Silver. Combining Online and Offline Knowledge in UCT. *International Conference of Machine Learning*, 2007.
- [4] Billouet, Simon and Hoock, Jean-Baptiste and Lee, Chang-Shing and Teytaud, Olivier and Yen, Shi-Jim. 9x9 Go as black with Komi 7.5 : At last some games won against top players in the disadvantageous situation. *ICGA*, 2009.
- [5] P. Rolet, M. Sebag and O. Teytaud. Boosting Active Learning to Optimality : A Tractable Monte-Carlo, Billiard-Based Algorithm. *ECML*, 2009