Monte Carlo Tree Search for Multi-Robot Task Allocation

Bilal Kartal, Ernesto Nunes, Julio Godoy, and Maria Gini
Department of Computer Science and Engineering
University of Minnesota
(bilal,enunes,godoy,gini)@cs.umn.edu

Abstract

Multi-robot teams are useful in a variety of task allocation domains such as warehouse automation and surveillance. Robots in such domains perform tasks at given locations and specific times, and are allocated tasks to optimize given team objectives. We propose an efficient, satisficing and centralized Monte Carlo Tree Search based algorithm exploiting branch and bound paradigm to solve the multi-robot task allocation problem with spatial, temporal and other side constraints. Unlike previous heuristics proposed for this problem, our approach offers theoretical guarantees and finds optimal solutions for some non-trivial data sets.

Introduction

Deploying robot teams to perform surveillance and delivery tasks requires that robots are allocated tasks with temporal constraints. We study the multi-robot task allocation problem with time-window constraints (henceforth TW-MRTA) in centralized settings. While centralized methods are regarded as less robust than distributed ones, they appear to be a natural choice in environments where robots work in tight spaces, communication is unconstrained, and high quality solutions and performance guarantees are important.

Many decentralized methods have been proposed for TW-MRTA (Koenig, Keskinocak, and Tovey 2010; Godoy and Gini 2012; Nunes and Gini 2015). However, these methods lack theoretical guarantees. In this work, we propose a novel centralized approach that offers high quality solutions; it also finds optimal allocations for many non-trivial data sets with in less time than existing optimal methods.

Our approach uses Monte Carlo Tree Search (MCTS), an anytime sampling based technique employing Upper Confidence Bounds (UCB) (Auer, Cesa-Bianchi, and Fischer 2002) to balance exploration vs. exploitation. MCTS with UCB is proven to converge to an optimal solution for finite horizon problems. MCTS has been shown to perform well for problems with high branching factor such as multi-agent story generation (Kartal, Koenig, and Guy 2014) and multi-robot patrolling (Kartal et al. 2015). We improve upon MCTS by pruning nodes in a branch and bound fashion, and propose an evaluation function that balances between minimizing distance and maximizing task completion rate. We evaluate our approach using the Solomon data set (Solomon 1987) for the vehicle routing problem with time-windows.

Problem Formulation

Let n and m denote the number of robots and tasks, respectively. We model each task with a x-y location, a capacity cost, a service time duration and a time-window determining when the task can be performed. Assume that there is a depot where robots are initially located, and to which they must return after completing their tasks. We use a graph representation, $G = (V, E)$, where V consists of the tasks and depot locations, and E includes all feasible edges obtained from pairwise vertices in V. Edge $e_a^b \in E$ iff task b can be completed after task a without violating the time-window constraints.

Let $\pi_i = \{r_i, t_i^1\}, \{r_i, t_i^2\}, ..., \{r_i, t_z^{|\pi_i|}\}$ denote the *individual task allocation policy* of robot i where t_i^j corresponds to the j-th allocated task in its policy and t_z corresponds to the depot. Let $\hat{\pi} = \{\pi_1 \cup \pi_2 \cup .... \cup \pi_n\}$ denote the *global task allocation policy* for the entire set of robots where each task can be allocated to a single robot. Let $D(\hat{\pi})$ represent the total distance traveled by the robots while following the policy $\hat{\pi}$. A complete policy is one where all tasks are allocated, i.e., $|\hat{\pi}| = m + n$, while in an incomplete policy some tasks remain unallocated, i.e., $|\hat{\pi}| < m + n$. Our objective is to find a *complete* policy $\hat{\pi}$ such that $D(\hat{\pi})$ is minimized.

Application of MCTS

Our adaptation of MCTS to TW-MRTA guarantees the utilization of all robots. At each level of the tree, a single robot (which did not return to the depot yet) is assigned one of the remaining tasks or returns to the depot. We employ robots one by one from the active fleet at each tree level. During the search, UCB is used to choose which task to allocate. Each parent node p chooses its child c with the largest $UCB(c)$ value according to Eq. 1. Here, $w(.)$ denotes the average evaluation score obtained by Eq. 2, $\hat{\pi}_c$ is the parent's policy so far updated to include child node c, p_v is visit count of parent node p, and c_v is visit count of child node c.

$$UCB(c) = w(\hat{\pi}_c) + \sqrt{\frac{2 \ln p_v}{c_v}} \qquad (1)$$

We apply branch and bound based on the best solution with $m' = m$ to prune nodes. We keep the best found solu-

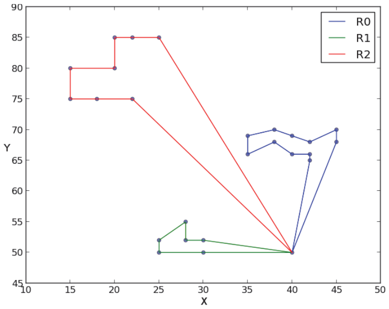

Figure 1: An example of an optimal policy found by our approach on Solomon's (C101) data set. $R_i, i \in \{0, 1, 2\}$ are robots' indices.

Scenario	MCTS	Optimal Distance	Time (in sec.)
C101	**191.3**	191.3	3
C102	**190.3**	190.3	80
C103	270.2	190.3	191
C104	235.7	186.9	293
C105	**191.3**	191.3	89
C106	**191.3**	191.3	6
C107	**191.3**	191.3	244
C108	242.4	191.3	256
C109	247.5	191.3	159

Table 1: Results on Solomon Benchmark for 25 tasks and three robots with a maximum search time of 5 minutes.

tion during the search and halt the search with a threshold by exploiting the anytime property of MCTS. Once the search is over, the solution $\hat{\pi}$ is decomposed into n individual robot policies to be executed simultaneously.

Policy Evaluation Function

We propose a policy evaluation function that guarantees that with more planning time, MCTS will find better solutions. Let α denote a loose upper bound on the total traveled distance $D(\hat{\pi})$, and $\hat{E}$ be a sorted version of E (in descending order of their distances). Then, $\alpha = 2 \times \sum_{i=1}^{m+n} e_i$ where $e_i \in \hat{E}$. To discourage incomplete policies, we define a negative reward parameter, ψ, which is computed as follows: $\psi = 2 \times \sum_{i=1}^{m-m'} e_i, e_i \in E'$, where m' denotes the number of completed tasks, and $E' \subset E$ is the set of edges directed from completed to uncompleted tasks and from uncompleted to the depot, sorted in descending order of their distances. Based on these definitions, we propose an evaluation function $f(\hat{\pi})$, to assess policy $\hat{\pi}$, as follows:

$$f(\hat{\pi}) = \frac{\alpha - (D(\hat{\pi}) + \psi)}{\alpha} \times \frac{m'}{m} \tag{2}$$

Our evaluation function is monotone: for every optimally allocated task, the actual traveled distance increases by some $c \geq 0$, while ψ decreases by at least c; this guarantees that the evaluation score of the policy never decreases. We can easily prove this by decomposing all traveled edges, however we omit the proof due to space constraints. Also, both $D(\hat{\pi}) \leq \alpha/2$ and $\psi < \alpha$, and consequently, $0.5 \leq f(\hat{\pi}) \leq 1$ holds for any complete policy $\hat{\pi}$.

Preliminary Results

We present preliminary results (see Table 1) on a Solomon data set with 25 tasks and three robots. For these experiments the UCB is not tuned and each experiment is run once. Our approach quickly finds solutions completing all the tasks, improves the distance quality, and although it takes more time on less constrained scenarios (e.g larger time-windows), it is orders of magnitude faster than exact methods. The optimal solution found by our approach is shown in Figure 1, where each robot is allocated to a task cluster.

Conclusions and Future Work

Preliminary results show that our approach performs well for the TW-MRTA problem, and in many instances it finds optimal allocations. For future, we plan to perform experiments on larger problem instances, and employ search parallelization. For example, we can create multiple trees with varied UCB exploration parameters and broadcast the best solution across all trees to expedite the pruning process. Lastly, we plan to compare our approach with cooperative auction based approaches (Koenig, Keskinocak, and Tovey 2010).

References

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine Learning* 47(2-3):235–256.

Godoy, J., and Gini, M. 2012. Task allocation for spatially and temporally distributed tasks. In *Proc. of the Int'l Conf. on Intelligent Autonomous Systems*, 603–612.

Kartal, B.; Godoy, J.; Karamouzas, I.; and Guy, S. J. 2015. Stochastic tree search with useful cycles for patrolling problems. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, 1289–1294.

Kartal, B.; Koenig, J.; and Guy, S. J. 2014. User-driven narrative variation in large story domains using monte carlo tree search. In *Int'l Conf. on Autonomous Agents and Multi-Agent Systems*, 69–76.

Koenig, S.; Keskinocak, P.; and Tovey, C. A. 2010. Progress on agent coordination with cooperative auctions. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 1713–1717.

Nunes, E., and Gini, M. 2015. Multi-robot auctions for allocation of tasks with temporal constraints. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2110 –2116.

Solomon, M. M. 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research* 35(2):254–265.