Dynamic Multi-Agent Task Allocation with Spatial and Temporal Constraints

(Extended Abstract)

Sofia Amador, Steven Okamoto, and Roie Zivan
Industrial Engineering and Management
Ben-Gurion University of the Negev
Beer-Sheva, Israel
{amador, okamotos, zivanr}@bgu.ac.il

ABSTRACT

Realistic multi-agent team applications often feature dynamic environments with soft deadlines that penalize late execution of tasks. This puts a premium on quickly allocating tasks to agents, but finding the optimal allocation is NP-hard because tasks must be executed sequentially by agents. We propose a novel task allocation algorithm that finds allocations that are fair (envy-free), balancing the load and sharing important tasks between agents, and efficient (Pareto optimal) by using a Fisher market based on a simplified problem model. Such allocations can be easily sequenced to yield high quality solutions, as shown empirically on problems inspired by real police logs.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: [Multiagent systems]

Keywords

Task Allocation, Market Equilibrium

1. INTRODUCTION

We formulate a novel task allocation problem in which agents move to perform dynamically-arising, spatially-situated tasks under soft deadlines that decrease utility the longer tasks are unperformed. Agents can share tasks to reduce workloads and speed up task execution. Inspired by law enforcement applications, we term this the *Law Enforcement Problem* (LEP). We propose a novel task allocation algorithm, FMC_TA, that first solves a simplified form of LEP using a Fisher market to obtain an initial allocation that is efficient and balanced over agents, then schedules the allocation for each agent according to a greedy heuristic.

2. LAW ENFORCEMENT PROBLEM

The dynamic Law Enforcement Problem (LEP) is modeled as a sequence of static problems instantiated when a new task arrives. In each static problem, cooperative agents (police units) $a_1, \ldots, a_n$ must be allocated tasks $v_1, \ldots, v_m$. Travel time between a_i's current location and v_j, or between two tasks $v_j, v_{j'}$, is denoted by $\rho(a_i, v_j)$ and $\rho(v_j, v_{j'})$, respectively.

There are two kinds of tasks: neighborhood *patrols* and *events* that require a police response. Every task v_j has arrival time $\alpha(v_j)$ and *importance* $I(v_j) > 0$; patrols are generally less important than events. Events also have a *workload* $w(v_j)$ specifying the amount of work (in time units) required to complete the task. Patrols have no workload but agents derive utility at a constant rate while patrolling. We assume that every task can be performed by a single agent but that multiple agents can also share a single task with additive contributions; shared events also divide the workload.

An allocation is an $n \times m$ matrix X where entry x_{ij} is the fraction of task v_j assigned to a_i. Agents can only perform a single task at a time, so tasks allocated to agent a_i must be ordered into a schedule $\sigma_i = (v_{s_1}, t_1, t'_1), \ldots, (v_{s_{M_i}}, t_{M_i}, t'_{M_i})$ of M_i triples of the form (v_{s_k}, t_k, t'_k), where v_{s_k} is the task performed from time t_k to t'_k. The time spent on each task must equal a_i's assigned share of the workload so $t'_k - t_k = x_{is_k} w(v_{s_k})$ for $1 \leq k \leq M_i$. Agents can only perform tasks at their current location, so $t_{k+1} - t'_k \geq \rho(v_{s_k}, v_{s_{k+1}})$ for $1 \leq k < M_i$ and $t_1 \geq t + \rho(a_i, v_{s_1})$ for allocation at time t.

Interrupting the current task CT_i of a_i incurs a penalty that depends on the total amount of work remaining, Δw. If $v_j \neq CT_i$, this penalty is $\pi(v_j, \Delta w) = \max\{I(v_j)c^{\Delta w}, \phi I(v_j)\}$, where $c \in [0, 1)$ and $\phi > 0$ are constants, with $\phi I(v_j)$ denoting the minimum penalty. There is no penalty if $v_j = CT_i$ or if CT_i is a patrol.

The utility of working on v_j decreases with the time t when task execution begins, according to the *soft deadline* function, $\delta(v_j, t) = \beta^{\gamma(t-\alpha(t))}$, where $\beta \in (0, 1]$ and $\gamma \geq 0$ are constants; $\gamma = 0$ for patrols as they have no deadline. Utility for a single agent a_i is

$$U(a_i) = \left(\sum_{k=1}^{M_i} x_{is_k} I(v_{s_k}) \delta(v_{s_k}, t_k) \right) - \pi(v_{s_1}, \Delta w)$$

and the total team utility is the sum of utilities for every agent.

3. FMC-BASED TASK ALLOCATION

We propose an innovative task allocation algorithm, FMC_TA, based on Fisher market clearing (FMC). FMC_TA first creates a Fisher market instance [5] with agents as buyers and tasks as goods; agents are given equal monetary endowments. The matrix R specifying buyer preferences for goods is created by considering agent utility in a simplified task allocation problem that ignores inter-task spatial and temporal constraints. Specifically,

$$r_{ij} = x_{ij} I(v_j) - \rho(a_i, v_j) - \pi(CT_i, \Delta w).$$

FMC_TA next solves the Fisher market to get an allocation X that is envy-free and Pareto optimal with respect to the simplified problem [5]. These properties ensure that we achieve an efficient allocation that is balanced over the agents. We demonstrate in Section 4 that this results in higher team utility in the full LEP than directly trying to maximize the utility represented by R. The FMC

Appears in: *Alessio Lomuscio, Paul Scerri, Ana Bazzan, and Michael Huhns (eds.), Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014), May 5-9, 2014, Paris, France.*

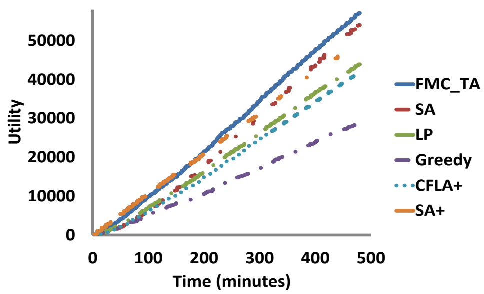

Figure 1: Accumulated team utility for shifts with 60 events.

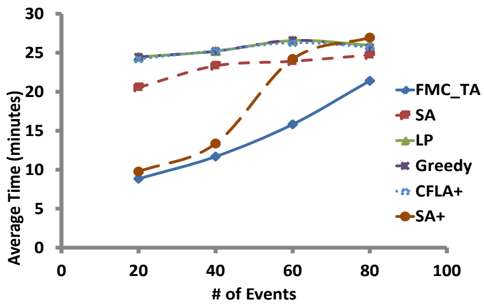

Figure 2: Response times compared to centralized approaches.

allocation can be computed in polynomial time in a centralized setting [2] or in pseudo-polynomial time in a distributed setting [6].

The final stage of FMC_TA schedules the fractions of tasks allocated to agents using a greedy heuristic that maximizes the utility that agents will derive from fulfilling a task. Each a_i schedules its tasks in decreasing order of $x_{ij}r_{ij}$. This can be done independently by each agent in both centralized and distributed settings.

4. EXPERIMENTAL EVALUATION

We compare FMC_TA to five centralized benchmark algorithms. Two are versions of simulated annealing: SA used a random starting point on every reallocation while SA+ started from the previous allocation. CFLA$^+$ is a version of CFLA [4] adapted to LEP by computing the maximum utility for pairs of tasks taking into account the soft deadlines. Greedy allocates tasks in decreasing order of importance to the agent that would derive highest utility. LP is identical to FMC_TA except that it finds X by using a linear program that directly maximizes team utility as represented by R.

We considered 20 random problems of an 8-hour shift with 9 agents in a city of 6×6 km divided into 9 neighborhoods. There were four types of events of decreasing importance from type 1 to type 4. Distribution of event types and workloads were based on police estimates, with 30%, 40%, 15%, 15% of events from type 1 to 4. Workloads were drawn from exponential distributions with means $58, 55, 45, 37$ for events of type 1 to 4. Locations were selected uniformly at random in the 6×6 planar region.

FMC_TA accumulates utility faster than the other approaches, as shown in Figure 1 for shifts of 60 events; this gap is larger when considering shifts with fewer events. LP outperforms CFLA$^+$ and Greedy, suggesting that the simplification in optimizing with respect to R is beneficial. However, the large gap between LP and FMC_TA indicates that direct optimization is less effective than one that seeks envy-freeness and Pareto optimality.

Rapid responses (short arrival times) are highly-valued by police departments, especially for more important events. FMC_TA outperforms the other approaches in this respect as well, as seen in

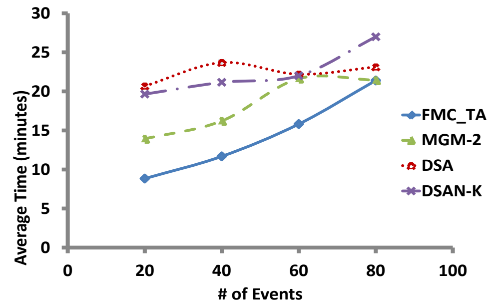

Figure 3: Response times compared to distributed approaches.

Figure 2. This is due to much faster responses to the more important type 1 and type 2 events, achieved by sharing more tasks than the other approaches (not shown for lack of space).

FMC_TA is similarly effective when compared to distributed approaches. Figure 3 shows that FMC_TA achieves faster response times than three leading DCOP algorithms: DSA [7], Distributed Simulated Annealing (DSAN) [1], and MGM-2 [3]. Again, this is especially true of more important tasks, resulting in FMC_TA achieving higher team utility (omitted for lack of space).

5. CONCLUSIONS

In this paper we proposed a new approach for dynamic task allocation that uses a simplified problem model to generate fair (envy-free) and efficient (Pareto optimal) allocations. We hypothesized that this combination of properties results in high quality solutions for task allocation problems in which we want all agents to contribute efficiently in order to achieve the group goal. Our experiments support this hypothesis, demonstrating the advantages of FMC_TA over competing centralized and distributed algorithms.

In future work we intend to investigate the use of non-linear utility functions. This will allow us to represent cooperative synergies where the utility derived by a group of agents is greater than the sum of utilities if each agent acted on its own.

6. REFERENCES

[1] M. Arshad and M. C. Silaghi. Distributed simulated annealing. *Distributed Constraint Problem Solving and Reasoning in Multi-Agent Systems, Frontiers in Artificial Intelligence and Applications series*, 112, November 2004.

[2] N. R. Devanur, C. H. Papadimitriou, A. Saberi, and V. V. Vazirani. Market equilibrium via a primal-dual-type algorithm. In *FOCS '02*, Washington, DC, USA, 2002.

[3] R. T. Maheswaran, M. Tambe, E. Bowring, J. P. Pearce, and P. Varakantham. Distributed algorithms for DCOP: A graphical-game-based approach. In *PDCS '04*, San Francisco, CA, USA, 2004.

[4] S. D. Ramchurn, M. Polukarov, A. Farinelli, C. Truong, and N. R. Jennings. Coalition formation with spatial and temporal constraints. In *AAMAS'10*, Richland, SC, 2010.

[5] J. H. Reijnierse and J. A. M. Potters. On finding an envy-free Pareto-optimal division. *Mathematical Programming*, 83:291–311, 1998.

[6] L. Zhang. Proportional response dynamics in the Fisher market. In *ICALP '09*, Berlin, Heidelberg, 2009.

[7] W. Zhang, Z. Xing, G. Wang, and L. Wittenburg. Distributed stochastic search and distributed breakout: properties, comparison and applications to constraints optimization problems in sensor networks. *Artificial Intelligence*, 161:1-2:55–88, January 2005.