Partially Cooperative Multi-Agent Periodic Indivisible Resource Allocation *

Extended Abstract

Yuval Gabai Shlosberg
Ben Gurion University of the Negev
Beer Sheva
gabaiyuv@post.bgu.ac.il

Roie Zivan
Ben Gurion University of the Negev
Beer Sheva
zivanr@bgu.ac.il

ABSTRACT

Asymmetric distributed constraint optimization problems (ADCOPs) in which agents are partially cooperative, is a model for representing multi-agent optimization problems in which agents, are willing to cooperate in order to achieve a global goal, as long as some minimal threshold on their personal utility is satisfied.

We contribute by: 1) extending the ADCOP model to represent resource allocation problems in which indivisible resources are periodically allocated, e.g., meeting rooms, operating rooms, etc. 2) adjusting partially cooperative local search algorithms to solve problems represented by the extended model. 3) presenting an implementation of a realistic problem that is represented by the proposed model, and empirical evidence of the compatibility of partially cooperative algorithms for this scenario.

KEYWORDS

Distributed Constraint Optimization; Distributed Local Search; Partial Cooperation

ACM Reference Format:

Yuval Gabai Shlosberg and Roie Zivan. 2021. Partially Cooperative Multi-Agent Periodic Indivisible Resource Allocation : Extended Abstract. In *Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), Online, May 3–7, 2021*, IFAAMAS, 3 pages.

1 INTRODUCTION

In many multi-agent applications, indivisible resources need to be allocated periodically among the agents. Some examples are, the allocation of class-rooms to lectures in a university, the allocation of meeting rooms in a working environment and the allocation of operating rooms in a hospital.

A partially cooperative model was proposed and studied in [1–3]. The model represents agents that act cooperatively – motivated by a desire to increase global (group) utility or by altruistic incentives – as long as a minimum condition on their personal utility is satisfied.

In distributed resource allocation scenarios, as described above, intentions to cooperate are implied by the willingness of agents to exchange or give away a resource that was allocated to them. Each agent has lower and upper bounds on the amount of resources allocated to it, which is determined by its requirements.

In this paper we advance the research on partial cooperative models and algorithms for distributed constraint optimization by: **1)** Proposing an extension of the socially motivated partial cooperative model (proposed in [2]), specific for periodic indivisible resource allocation. **2)** Proposing adjustments of socially motivated distributed local search algorithms for periodic resource allocation, and **3)** Presenting a realistic implementation of the proposed model and presenting experimental results that demonstrate the compatibility of the proposed algorithms with this realistic scenario.

Our empirical results demonstrate the importance of shared preferences, in scenarios where agents are partially cooperative. Ignorance may lead to altruistic decisions, which hurt the altruist agent more than they benefit their neighbors. On the other hand, exchanged indications regarding the preferences of agents on their neighbors' actions, trigger high quality solutions.

2 PROBLEM FORMALIZATION

A periodic indivisible resource allocation problem is composed of: A set of n agents $A = \{A_1, A_2, ..., A_n\}$ and a set of m indivisible resources $R = \{r_1, r_2, ..., r_m\}$. The atomic time unit in which a resource can be allocated is denoted by t and the time horizon H is finite. Each resource r_j is assigned at each time unit t_k solely to one of the agents $A_i \in A$. Thus, an allocation of a resource to an agent in some time unit is a triplet $\langle A_i, r_j, t_k \rangle$. A *complete allocation* CA is a set of exactly $m \cdot H$ allocation triplets, such that, each of the resources r_j $(1 \leq j \leq m)$ is included exactly once in triplets with each of the time units t_k $(1 \leq k \leq H)$ and every time unit t_k is included exactly once in a triplet with each resource r_j .

Each agent A_i has a cardinal constraint CC_i that defines the utility she derives with respect to the number of resources she received in the specified time interval, and two bounds. A lower bound defines the minimal amount of resources required in the time interval (LB_i), and an upper bound that defines the maximal number of resources the agent can use in the time interval (UB_i). These bounds define a different utility/cost scheme. An allocation that does not satisfy the lower bound incurs a high cost. It can be a fixed cost or related to the amount of resources allocated.The upper bound (UB_i) defines the number of resources allocated to agent A_i such that if she is allocated an additional resource, there is no increment to her utility.

The utility that an agent A_i derives from a complete allocation CA is denoted by $U_i(CA)$. The global utility of CA is the sum of the personal utilities of the agents, $U(CA) = \sum_{i=1}^{n} U_i(CA)$.

*This research is partially supported by the Israeli Innovation Authority as part of the knowledge directing track.

Proc. of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2021), U. Endriss, A. Nowé, F. Dignum, A. Lomuscio (eds.), May 3–7, 2021, Online.

PIRA as an ADCOP: In order to represent a PIRA as an asymmetric DCOP, we define the possible allocations of resources to agents in terms of variables and domains of values that can be assigned to them. Furthermore, the utility calculation needs to be decomposed into asymmetric constraints that agents can compute and aggregate. Agent A_i holds variables $v_{i_1}, v_{i_2}, ...v_{i_k}$, where k is the maximal number of resources that she may be allocated. The domains include all the relevant ordered pairs $\langle r, t \rangle$.

The utility that an agent derives from an allocation, is defined by her personal constraints. We denote by C_i the set of constraints of agent A_i. A constraint $c \in C_i$ includes a set of q assignments, $q \geq 1$ and the utility the agent derives from this constraint, i.e., $c = [\langle A_{i_1}, r_{j_1}, t_{k_1} \rangle, ..., \langle A_{i_q}, r_{j_q}, t_{k_q} \rangle, u_i]$ where at least one of the allocations is to agent A_i. Personal preferences are represented by unary constraints. *Cardinal constraints* are also unary constraints, which include all the resources allocated to a single agent. The utility that agent A_i derives from an allocation, U_i, is the sum of the utilities she derives from all the constraints she is involved in.

Hospital operating room scheduling implementation: has agents representing hospital wards and variables representing room allocations. Variables' domains include all possible allocation pairs of room and date that the agent can be assigned. Unary constraints define the utility derived from an allocation of a room at some date to a ward. Binary hard constraints exclude allocations of the same room to different wards on the same day. Cardinal constraints specify whether the allocation satisfies the ward's requirements and the utility they derive of the amount of rooms that were allocated to them in the specified period of time. They also implement the upper and lower bounds as described above.

3 PIRA ALGORITHMS

In order to solve PIRA_ADCOPs we adjusted partial cooperative local search algorithms (including socially motivated partial cooperative algorithms) such that they will be compatible with PIRA problems [1, 2]. The main difference between the existing general partial cooperative algorithms and the algorithms adjusted for PIRA, is that the actions in PIRA algorithms are specific requests for the release or exchange of resources in certain periods. The expected benefits that agents exchange are either the utility that they are expected to derive from the resources that are released for their use in specific periods or the increment in utility as a result of an exchange.

4 EXPERIMENTAL EVALUATION

Our experiments included different versions of socially motivated local search algorithms, solving a hospital operating room scheduling problem. The problem parameters were set as similar as possible to the realistic scenario in the hospital at our home town (apart from the sensitive data that we did not get permission to share). They included 10 wards and 15 operating rooms allocated periodically every day. The allocation was for a five working day week.

The versions of the partial cooperative local search algorithms we compared included (corresponding notations in brackets): AGC with $\lambda = 0.1$ (*AGC*_0.1), AGC with $\lambda = 0.7$ (*AGC*_0.7), SM_AGC with $\lambda = 0.1$ (*SM*_0.1), SM_AGC with $\lambda = 0.7$ (*SM*_0.7), cooperative SM_AGC with $\lambda = 1$ (*SM*_*c*), SM_AGC with bounds (agents

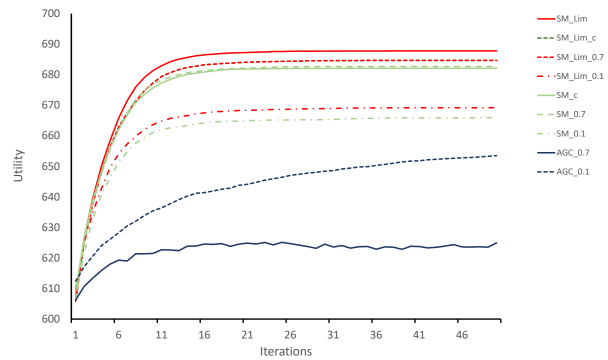

Figure 1: Average social welfare for the origin problem set. Initial allocation based on a real allocation.

reject any request that may cause a reduction beneath their lower bound and do not require a trade for allocations they give beyond their upper bound) (*SM_LIM*), SM_AGC with bounds and $\lambda = 0.1$ (*SM_LIM*_0.1), SM_AGC with bounds and $\lambda = 0.7$ (*SM_LIM*_0.7), cooperative SM_AGC with bounds and $\lambda = 1$ (*SM_LIM_c*).

Figure 1 presents the global utility (social welfare) derived from the allocations generated by the versions of the algorithms listed above, as a function of the number of iterations preformed. Consistent with the results presented in [2], the results depicted demonstrate the clear advantage of the socially motivated versions over standard AGC. Moreover, they demonstrate that intentions for cooperation (represented by λ) must be combined with preference sharing among agents, in order to increase social welfare. Thus, in all socially motivated versions the $\lambda = 0.7$ versions outperform the $\lambda = 0.1$ versions. On the other hand, The $lambda = 0.1$ version is more successful in AGC. Among the socially motivated versions of the algorithm, the ones using bounds are most successful. In fact, the best results were achieved by the version that uses bounds and does not use the cooperation threshold λ.

5 CONCLUSIONS

Periodic resource allocation scenarios, where agents cooperate to increase the benefit of their organization, as long as their minimal personal utility is preserved, fit well the Partial Cooperative paradigm. In these scenarios agents trade and exchange allocations in order to increase both personal and global gain. Each agent performs autonomously, according to her own personal constraints and possible benefits, however, she is willing to reduce her personal utility in order to help her peers increase their own, as long as her allocation does not drop into an unacceptable state.

Modelling this problem as an ADCOP allows the adjustment and use of socially motivated partially cooperative algorithms, which make efficient use of the intentions for cooperation of the agents in order to maximize global utility while keeping all agents above their cooperation threshold. The socially motivated partial cooperative algorithms adjusted for solving periodic resource allocation are unique, since agents need to perform trades in order to improve the current allocation, i.e. they perform an action that involves more than one agent, and affects their utility.

The proposed model and algorithms make DCOP applicable for a new class of real applications, as we demonstrated for the hospital operation room allocation problem.

REFERENCES

[1] Alon Grubshtein, Roie Zivan, and Amnon Meisels. 2012. Partial Cooperation in Multi-agent Local Search. In *ECAI 2012 - 20th European Conference on Artificial Intelligence. Including Prestigious Applications of Artificial Intelligence (PAIS-2012) System Demonstrations Track, Montpellier, France, August 27-31 , 2012.* 378–383.

[2] Tal Ze'evi, Roie Zivan, and Omer Lev. 2018. Socially Motivated Partial Cooperation in Multi-agent Local Search. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018.* 2150–2152.

[3] Roie Zivan, Alon Grubshtein, Michal Friedman, and Amnon Meisels. 2012. Partial cooperation in multi-agent search. In *International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2012, Valencia, Spain, June 4-8, 2012 (3 Volumes).* 1267–1268.