DEALING WITH COALITION FORMATION IN THE RoboCup RESCUE
An Heuristic Approach

Daniel Epstein and Ana L. C. Bazzan
PPGC, UFRGS, Caixa Postal 15064, CEP 91501-970, Porto Alegre, RS, Brazil

Keywords: Robocup rescue, Coalitions, Multiagent systems.

Abstract: Finding an optimal coalition structure to divide agents in groups is equivalent to the set-partitioning problem. Several algorithms have been proposed. However, even to find a sub-optimal value they have to search within an exponential number of coalition structures. Therefore, in this paper, we use one of the proposed algorithms, which is anytime and hence suits environments such as the RoboCup Rescue, where a response is needed in a short time frame. Moreover, we propose to combine this algorithm with heuristics to reduce and constraint the number of agents and tasks that are allowed to participate in a coalition. In this paper we discuss the application of such an approach in a complex task allocation scenario, the RoboCup Rescue.

1 INTRODUCTION

In multiagent encounters, a coalition can be defined as a group of agents which decide to cooperate in order to achieve a common goal. A coalitional game (CG) is a model of interacting decision-makers that focuses on the behavior of *groups* of individuals. An outcome of a CG is a partition of the set of all players into coalitions, together with an action for each coalition. Unfortunately, this problem is equivalent to the set partitioning one.

Generating the coalition structure and finding the optimal one is a hard problem. The first algorithm to establish a bound within a minimal amount of search was given in (Sandholm et al., 1999). However, despite the fact that the bound can be established in linear time in the size of the input, the bad news remain that 2^{a-1} (where a is the number of agents) nodes of the coalition structure graph have to be searched in order to guarantee the worst case bound from optimum. This prevents the use of that kind of algorithm in multi-agent systems with a high number of agents. Sandholm and colleagues themselves showed that it is possible to lower the bound with further and/or smarter search (Sandholm et al., 1999). For example, Rahwan and colleagues (Rahwan et al., 2007) have proposed a near-optimal anytime algorithm for coalition structure generation, which partitions the space in terms of coalitions of particular sizes.

Here, we approach the problem taking advantage of the fact that particular CG's have coalitions and components that are meaningless in a given scenario. With this we hope to decrease the amount of search.

2 BACKGROUND

Due to space limitation we cannot fully describe the RoboCup Rescue simulator. Interested readers are referred to (Kitano et al., 1999; Skinner and Barley, 2006) for more details.

Currently the simulator tries to reproduce conditions that arise after the occurrence of an earthquake in an urban area, such as the collapsing of buildings, road blockages, fire spreading, buried and/or injured civilians. In the RoboCup Rescue simulator the main agents are fire brigades, police forces, and ambulance teams. These have limited perception of their surroundings; can communicate, but are limited on the number and size of messages they can exchange.

Information available to the agents contains attributes related to buildings, civilians, and blockages for instance. Regarding the civilians these attributes are the location, health degree (called HP), damage (how much the HP of those civilians dwindles every time step), and buriedness (a measure of how difficult it is to rescue those particular civilians). The attributes related to buildings are location, whether or not it is burning, how damaged it is, and the type of mate-

Epstein D. and L. C. Bazzan A..
DEALING WITH COALITION FORMATION IN THE ROBOCUP RESCUE - An Heuristic Approach.
DOI: 10.5220/0003294607170720
In *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence* (ICAART-2011), pages 717-720
ISBN: 978-989-8425-40-9

rial used in the construction. Finally the attributes of blockages are location in the street, cost to unblock, and their size.

To measure the performance of the agents, the rescue simulator defines a score which is computed, at the end of the simulation, as shown in Equation 1, where P is the quantity of civilians alive, H is a measure of their health condition, H_{init} is a measure of the health condition of all civilians at the beginning of the simulation, B is the building area left undamaged, and B_{max} is the initial building area.

$$Score = \left(P + \frac{H}{H_{inicial}}\right) * \sqrt{\frac{B}{B_{inicial}}} \qquad (1)$$

Regarding related work, the reader is referred to (Ferreira et al., 2010) where an extensive revision is made. Here we only mention works that are used to compare our results, as well as previous works for generating coalition structures. In the particular scenario of the RoboCup Rescue, agents may form coalitions to solve tasks posed by the environment. We start with the latter.

The generation of all possible coalition structures is exponential in the number of agents. This is an important issue because it was demonstrated that finding the optimal coalition structure is NP-complete (Sandholm et al., 1999).

Another form of task allocation in the RoboCup Rescue, i.e. non coalition-based, is by means of task assignment. The generalized assignment problem (GAP) is one possible model used to formalize a task allocation problem. It deals with the assignment of tasks to agents, constrained to agents' available resources, and aims at maximizing the total reward.

The GAP model was extended by Scerri *et al* (Scerri et al., 2005) to incorporate two features: scenario dynamics and inter-task constraints. This extended model was called extended generalized assignment problem (E-GAP).

An E-GAP can be converted to a distributed constraint optimization problem (DCOP) taking agents as variables and tasks as the domain of values. However, this modeling leads to dense constraint graphs. In order to deal with these issues, Scerri *et al.* present an approximate algorithm called Low-communication Approximate DCOP (LA-DCOP) (Scerri et al., 2005) that uses a token-based protocol; agents perceive a task in the environment and create a token to represent it, or they receive a token from another agent. An agent decides whether or not to perform a task based both on its capability and on a threshold.

If an agent in LA-DCOP is able to perform more than one task, then it must select those that maximize its capability given its resources. This maximization problem can be reduced to a binary knapsack problem (BKP), which suggests that the complexity of LA-DCOP depends on the complexity of the function implemented to deal with the BKP.

Swarm-GAP (Ferreira et al., 2010) resembles LA-DCOP in the sense that it is also GAP-based, is approximate, uses tokens for communication, and deals with extreme teams. An agent in Swarm-GAP decides whether or not to perform a task based on the model of division of labor used by social insects colonies, which has low communication and computation effort. This avoids the complexity of the maximization function used in LA-DCOP. A key parameter in the approach is the stimulus s agents have towards tasks, as it is responsible for the selectiveness of the agent regarding perceived tasks.

In order to deal with inter-task constraints, agents in Swarm-GAP just increase the tendency to perform related tasks by a factor called execution coefficient (details in (Ferreira et al., 2010)).

3 HEURISTICS FOR COALITION FORMATION IN THE RESCUE

In the RoboCup Rescue the number of coalition structures is affected not only by the number of agents but also by the number of tasks. Low-priority tasks are not considered in the set of coalition structures. From the point of view of the agent, some tasks may also be discarded because the agent either has no resources to perform it, or is located far away from it. This way agents that cannot be allocated to important tasks are removed from the set of agents to form coalitions, further reducing the number of possible coalition structures. After we find a reduced number of agents and tasks, the search of the actual optimal coalition structure is performed by the anytime algorithm proposed by (Rahwan et al., 2007). In what follows we analyze the characteristics of tasks and agents as well as how they influence the quality of the simulation.

In order to rank tasks that are related to fire spots, one must examine how the final score (Eq. 1) is affected by such tasks. Buildings that have larger areas are more important than those with small ones. Further, the type of building (regarding construction material) is an important factor. Finally, the area of the neighboring buildings must equally be considered in order to try to prevent the propagation of fire.

To appropriately rank tasks related to rescuing civilians, one must consider their HP, their location, and how difficult is the rescue. The heuristics that are considered here are as follows. Civilians with higher priority are those that have a high HP but that can still be saved i.e. the time to arrive at its location and rescue it (considering how buried it is) is enough. Coalitions are important here because the presence of more than one ambulance is key to rescue a civilian faster (one civilian can be saved in 4 time units by one ambulance or in 2 time units by 2 ambulances), up to a limit. As mentioned, an ambulance must be able to arrive to the location of a civilian within a time frame that is compatible with its rescue. Ambulances that do not fulfill such constraints must be excluded from the corresponding coalitions, thus reducing the number of coalition structures.

Contrarily to the two other kinds of tasks, removing blockages does not have a direct impact in the score. However, blockages hinder the traffic of the ambulances and firefighters thus they have a great indirect impact. The attributes of this kind of task are its type (arterial, secondary, and so on), number of blocked lanes, and how difficult is the removal. Regarding the former, priority will be given to arterials and/or roads with many blocked lanes. Roads that are only partially blocked have low priority because traffic is still possible there. Regarding the last attribute, priority is given to roads with less blockages so that the number of roads where traffic is at least partially possible is maximized.

The main heuristic to consider a police force or not in a coalition is whether or not it is located very close to the blockage. If it is not within three time units from location of a task, a police is not considered to deal with this task. This number was experimentally selected.

For ambulances one has to consider the trip time between its current location and the location of a civilian. Civilians that are not reachable in a time frame that is compatible with the time to unbury and rescue them are not considered. This time is called here c and is the sum of the time to arrive at the civilian location plus the remaining life time of the civilian, computed based on its HP and level of buriedness. Teams of ambulances are not considered here for sake of including or not an ambulance in a coalition but of course the value of coalitions with different number of ambulances is different as mentioned before.

Fire fighters also have distance as an important attribute i.e. the agent must be able to reach the task location in time to be valuable. However, another one is equally important: the level of water each has. Only tasks for which the agent has a good ratio water to distance will be considered. Therefore agents with low level of water must consider only nearby tasks. Since on average a building takes 5 time steps to lose half of its value, no agent that is more than 5 time steps away of the building will be considered.

We are now in position to formalize the heuristics related to the agents regarding the tasks.

Heuristic for Firefighters

Let A_j be the area of building j, n_j the number of adjacent buildings to j, A_j^k the area of the k-th building around j, and F_j the influence of A_j in the score (Eq. 1) i.e. how destroyed is building j. The value of a task (building) j (irrespective of any agent i) is given by $V_j = A_j \times F_j + \frac{\sum_{k=1}^{n_j}(A_j^k \times F_j^k)}{A_j}$. If we also include D_{ij}, the distance between i and j in the value of task j, now for each agent i, this equation turns: $V_j^i = \left(A_j \times F_j + \frac{\sum_{k=1}^{n_j}(A_j^k \times F_j^k)}{A_j}\right) \times \frac{5}{2 \times D_{ij}}$.

Heuristic for Police Forces

Similarly to firefighters, here we quantify the importance of tasks, this time for police forces. As mentioned, the main idea is to free the highest possible number of roads, even if partially. Relevant attributes here are: the number of lanes in a road, P_t; the number of free lanes, P_l; the cost to unblock the lane, C_b. Thus, the value of task (removal of blockage j) V_j^i, already considering the distance D_{ij} to agent i is: $V_j^i = \left(P_{t_j} - 2 \times P_{l_j} - C_{b_j} + \frac{1}{P_{t_j}}\right) \times \frac{3}{2 \times D_{ij}}$.

Heuristic for Ambulances

To compute the value of rescue tasks, we must compute the life expectancy of each civilian, E_j. Let HP_j be the hit points of a civilian j (roughly a measure of how *alive* it is), and B_j a measure of how buried j is. Given that B_j is reduced by 200 at each time step, we can compute E_j: $E_j = HP_j - B_j \times 200$ The value of j for agent i, E_j^i is: $E_j^i = HP_j - B_j \times 200 - D_{ij} \times 200$.

4 EXPERIMENTS AND RESULTS

We have run our experiments in two maps that are largely used by the community around RoboCup Rescue, namely *Kobe* and *Kobe*4.Due to limitation of space we restrict the discussion of results to the first map. We have used version 0.49.9 of the simulator. We remark that the version 0.50 has several bugs. Also, there is a completely new version of the simulator, which was not used by us so far.

Table 1: Scores for non-coalition-based approaches.

LA-DCOP	Swarm-Gap	Greedy
49.69 ± 6.31	44.97 ± 1.76	43.78 ± 7.19

Table 2: Scores for the coalition-based approach, for various values of ρ.

$\rho = 0.3$	$\rho = 0.5$	$\rho = 0.7$
70.28 ± 6.94	67.63 ± 9.68	65.34 ± 7.22

In this map there are 6 ambulance teams, 10 fire brigades, and 8 police forces. There are also 72 civilians, 734 buildings, and 820 roads. The dynamics of the rescue scenario means that the number and type of tasks change, which is a problem for the grouping of the agents and consequent re-computation of the possible coalitions. These must be re-grouped from time to time or event-based as also done in (Santos and Bazzan, 2010). We have tried both approaches but the former does not perform well because different tasks have different execution times. Therefore we only discuss the latter. For the event-based re-computation, such an event is the reaching of a certain rate ρ of ungrouped agents i.e. agents that have finished performing their previous assigned tasks and that are selecting tasks in a greedy way or not at all. Tested values were $\rho \in \{0.3, 0.5, 0.7\}$ i.e. new coalitions are formed when 30%, 50%, or 70% of the agents of a given type are no longer in coalitions (because their assigned tasks are over).

In order to compare the results with other approaches that are not based on coalition formation, we use LA-DCOP, Swarm-GAP, and a greedy strategy. The latter is equivalente to the so-called sample agents. We have performed 20 repetitions of each simulation. For LA-DCOP, the threshold used was $T = 0.2$, while Swarm-GAP was tested with stimulus $s = 0.1$. These values were selected after calibration in (Ferreira et al., 2010). Results appear in Table 1 where we give the scores (Eq. 1) at the end of the simulation. The same setting was then used to test the coalition-based approach, for various values of ρ. Results appear in Table 2.

It is possible to see that the use of coalitions represents an increase in performance. The best scores are achieved if the re-grouping of agents (i.e. the re-evaluation of the coalition formation) is done when at most 30% of the agents have finished their tasks.

REFERENCES

Ferreira, Jr., P. R., dos Santos, F., Bazzan, A. L. C., Epstein, D., and Waskow, S. J. (2010). Robocup rescue as multiagent task allocation among teams: experiments with task interdependencies. *Journal of Autonomous Agents and Multiagent Systems*, 20(3):421–443.

Kitano, H., Tadokoro, S., Noda, I., Matsubara, H., Takahashi, T., Shinjou, A., and Shimada, S. (1999). Robocup rescue: search and rescue in large-scale disasters as adomain for autonomous agents research. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, volume 6, pages 739–743, Tokyo, Japan. IEEE.

Rahwan, T., Ramchurn, S. D., Dang, V. D., and Jennings, N. R. (2007). Near-optimal anytime coalition structure generation. In *Proc. of the Int. Joint Conf. on Art. Intelligence (IJCAI 07)*, pages 2365–2371. available at http://ijcai.org/proceedings07.php.

Sandholm, T., Larson, K., Andersson, M., Shehory, O., and Tohmé, F. (1999). Coalition structure generation with worst case guarantees. *Artificial Intelligence*, 111(1–2):209–238.

Santos, D. S. d. and Bazzan, A. L. C. (2010). Distributed clustering for group formation and task allocation. In van der Hoek, W., Kaminka, G. A., Lespérance, Y., Luck, M., and Sen, S., editors, *Proceedings of the Ninth International Conference on Autonomous Agents and Multi-Agent Systems*, pages 1429–1430, Toronto. IFAAMAS: Internatioal Foundation for Autonomous Agents and Multiagent Systems.

Scerri, P., Farinelli, A., Okamoto, S., and Tambe, M. (2005). Allocating tasks in extreme teams. In Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M. P., and Wooldridge, M., editors, *Proc. of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 727–734, New York, USA. ACM Press.

Skinner, C. and Barley, M. (2006). Robocup rescue simulation competition: Status report. In Bredenfeld, A., Jacoff, A., Noda, I., and Takahashi, Y., editors, *RoboCup 2005: Robot Soccer World Cup IX*, volume 4020 of *Lecture Notes in Computer Science*, pages 632–639. Springer-Verlag, Berlin.