A MCS-based methodology for computing coalitions in Multirobot Systems

Patrice Caire [1] and **Antonis Bikakis**[2]

Abstract. We introduce a new methodology to systematically compute all possible coalitions among the robots of a multirobot system using the model and algorithms of Multi-Context Systems (MCS).

1 Introduction

In multirobot systems, robots have goals to satisfy. Typically, a robot cannot reach its goals just by itself, but needs to cooperate with other robots, for example because it needs a resource or does not have the capability required to perform a task. The questions then are: Which robots to cooperate with? Which coalition to join? The problem of assembling a group of robots to cooperate in order for each of them to reach their own goals, or their common goals, is referred to as coalition formation. To our knowledge, there are no previous works on multirobot systems that have made use of MCS, however, MCS can bring new insights to the problem. Based on a parallelism that we draw between notions of MCS on the one hand such as *context*, *bridge rule* and *equilibria*, and notions of multirobot systems on the other such as *robot*, *dependency* and *coalition*, we can use MCS tools to solve problems in multirobot systems.

In this paper we address the question of how to find and evaluate coalitions among robots in a multirobot system. Our methodology is the following. First, we model the dependencies among the system robots, using dependence relations. Second, we model the system as a MCS: each robot is modeled as a context with a knowledge base, an underlying logic and a set of bridge rules. Third, we compute all possible coalitions among the robots using algorithms used for the computation of equilibria in MCS. Fourth, given a set of requirements, we show how to select the best solutions, and illustrate our research with an example using humanoid robots.

2 Running Example

In an office building, there are assistant robots to human workers. As office materials are often insufficient, they have to be shared. Workers can submit requests to the robots to deliver the needed materials for them, while they keep on working at their desks. We refer to a request submitted to the robots as a task. Workers and robots communicate via a simple web-based application, which transmits the workers' requests to the robots and keeps track of their status. The robots have limited computational resources. They only keep track of what they have done recently. Furthermore, not all of the robots know about the exact location of a material. Therefore, they rely on each other for information about the location of materials. In short, the last robot which dealt with a material is the one which knows where it is.

[1] SnT and CSC, University of Luxembourg, email:patrice.caire@uni.lu
[2] Department of Information Studies, UCL, email:a.bikakis@ucl.ac.uk

We assume that there is a set of 4 robots $R = \{r_1, r_2, r_3, r_4\}$ and four tasks: $T = \{t_1, t_2, t_3, t_4\}$, where: t_1 is to deliver a pen to desk A, t_2 is to deliver a piece of paper to desk A, t_3 is to deliver a tube of glue to desk B, and t_4 is to deliver a cutter to desk B. Due to their structures, robots can carry certain materials: r_1 can carry the pen or the glue, r_2 can only carry the paper, r_3 can carry the glue or the cutter, and r_4 can carry the pen or the cutter.

The accomplishment of a task requires getting information about the source and the destination of a material and carrying the material from the source to the destination. Each robot knows who has this information, but the actual coordinates are revealed only after an agreement on a coalition among the robots has been made. This involves interdependency among robots. Upon receiving information about the tasks, robots decide how to form coalitions to execute them. We refer to a coalition as a group of robots executing a task. For example to accomplish all tasks t_1, t_2, t_3, t_4, the following coalitions may be formed: $C_0 : \{(r_1, t_3), (r_2, t_2), (r_3, t_4), (r_4, t_1)\}$ and $C_1 : \{(r_1, t_1), (r_2, t_2), (r_3, t_3), (r_4, t_4)\}$ After forming coalitions, each robot has to generate its own plan to carry out the assigned tasks, e.g. plan the optimal route to get the material and carry it to its destination. Figure 1 presents robots' knowledge and current distances.

Robots' knowledge

Robot	r_1				r_2			
Task	t_1	t_2	t_3	t_4	t_1	t_2	t_3	t_4
Source		x			x			
Destination	x		x					x

Robot	r_3				r_4			
Task	t_1	t_2	t_3	t_4	t_1	t_2	t_3	t_4
Source				x			x	
Destination		x						

Distances among locations

Robot	Pen	Paper	Glue	Cutter
r_1	10	15	9	12
r_2	14	8	11	13
r_3	12	14	10	7
r_4	9	12	15	11

Destination	Pen	Paper	Glue	Cutter
Desk A	11	16	9	8
Desk B	14	7	12	9

Figure 1. Robots'knowledge (left); Distances (right)

3 Computing coalitions with MCS

One question that arises in scenarios such as the one that we present in Section 2 is how to compute the alternative coalitions that may be formed to achieve a set of given goals. Our proposed solution is based on the use of heterogeneous MCS [2]. Roughly, a MCS is a set of contexts, each one composed of a knowledge base with an underlying logic, and a set of bridge rules, which enable adding information to a context based on what is believed or disbelieved in other contexts.

3.1 Modeling dependencies

We model each robot in a multirobot system as a context in a MCS. The context knowledge base describes the goals of the robot and the actions that it can perform. Goals and actions are represented as literals of the form g_k, a_j, respectively. Bridge rules represent the dependencies of the robot on other robots to achieve its goals. A dependence relation $dp : basic\_dep(r_i, r_j, g_k, p_l, a_m)$ denotes that robot r_i depends on robot r_j to achieve goal g_k, because r_j may perform action a_m needed in the plan p_l, which achieves the goal [15]. For a goal g_k of robot r_i, which is achieved through plan $p_l = (r_1 : a_1, r_2 : a_2, ..., r_n : a_n)$, where $r_j : a_j$ represents action a_j performed by robot r_j, the following dependence relations hold:

$$dp_j : basic\_dep(r_i, r_j, g_k, p_l, a_j), j = \{1, ..., n\}$$

We denote this set of dependencies as $DP(r_i, g_k, p_l)$. And we define bridge rules describing dependence relations as follows:

Definition 1 *For a robot r_i with goal g_k achieved though plan p_l, the set of dependencies $DP(r_i, g_k, p_l)$ - where $p_l = (r_1 : a_1, r_2 : a_2, ..., r_n : a_n)$ - is represented by a bridge rule of the form:*

$$(c_i : g_k) \leftarrow (c_1 : a_1), (c_2 : a_2), ..., (c_n : a_n)$$

where c_j, $j = 1, ..., i, ..., n$, represents robot r_j.

Definition 2 extends the notion of MCS to multirobot systems.

Definition 2 *A MCS $M(R)$ corresponding to a multirobot system R is a set of contexts $c_i = \{L_i, kb_i, br_i\}$, where $L_i = (\boldsymbol{KB}_i, \boldsymbol{BS}_i, \boldsymbol{ACC}_i)$ is the logic of robot $r_i \in R$, $kb_i \in \boldsymbol{KB}_i$ is a knowledge base that describes the actions that r_i can perform and its goals, and br_i is a set of bridge rules, a subset of which represents the dependencies $DP(r_i, g_k, p_l)$ of r_i on other robots in R for all goals g_k of r_i and all plans p_l with which these goals can be achieved.*

Example 1 *In our example, we assume that the four robots use propositional logic. Robots r_1-r_4 are modelled as contexts c_1-c_4, respectively, with the following knowledge bases:*

$$kb_1 = \{a_{2s}, a_{1d}, a_{3d}, a_{1c} \vee a_{3c}\}, kb_2 = \{a_{1s}, a_{4d}, a_{2c}\}$$
$$kb_3 = \{a_{4s}, a_{2d}, a_{3c} \vee a_{4c}\}, kb_4 = \{a_{3s}, a_{1c} \vee a_{4c}\}$$

where a_{ij} represents the actions that a robot can perform. i stands for the object to be delivered: 1 stands for the pen, 2 for the paper, 3 for the glue and 4 for the cutter. j stands for the kind of action that the robot can perform: c stands for carrying the object, s stands for providing information about the source, while d stands for providing information about the destination of the object.

We represent the four tasks that the robots have to perform, t_i, as goals, g_i. For example g_1 represents the task of delivering the pen to desk A (t_1). Assuming that each robot can achieve goal g_i, i.e. deliver an object i, only if it can perform action a_{ic} (i.e. carry object i) the following bridge rules describe how the four goals can be achieved.

$$(r_1 : g_1) \leftarrow (r_1 : a_{1c}), (r_2 : a_{1s})$$
$$(r_4 : g_1) \leftarrow (r_4 : a_{1c}), (r_2 : a_{1s}), (r_1 : a_{1d})$$
$$(r_2 : g_2) \leftarrow (r_2 : a_{2c}), (r_1 : a_{2s}), (r_3 : a_{2d})$$
$$(r_1 : g_3) \leftarrow (r_1 : a_{3c}), (r_4 : a_{3s})$$
$$(r_3 : g_3) \leftarrow (r_3 : a_{3c}), (r_4 : a_{3s}), (r_1 : a_{3d})$$
$$(r_3 : g_4) \leftarrow (r_3 : a_{4c}), (r_2 : a_{4d})$$
$$(r_4 : g_4) \leftarrow (r_4 : a_{4c}), (r_3 : a_{4s}), (r_2 : a_{4d})$$

One constraint is that each goal cannot be achieved by more than one robots, e.g. the pen must be delivered to desk A by one robot only. This is described with bridge rules of the form

$$\neg r_l : g_i \leftarrow r_k : g_i$$

where $k, l = \{1...4\}$ and $k \neq l$.

3.2 Computing and evaluating coalitions

An equilibrium in MCS represents an acceptable belief state of the system. Each belief set in this state is derived from the knowledge base of the corresponding context and is compatible with the applicable bridge rules. For a MCS $M(R)$ that corresponds to a multirobot system R, a belief set S_i in an equilibrium $S = \{S_1, ..., S_n\}$ of $M(R)$ describes the goals that robot $r_i \in R$ can achieve and the actions that it can perform. S, therefore, here represents a coalition among the robots in R.

In order to compute the potential coalitions in a multirobot system R, one then has to formulate the MCS $M(R)$ that corresponds to R, and compute the equilibria S of $M(R)$. The computation of equilibria can either be done by a central entity that monitors the bridge rules of all robots [2]; or in a distributed fashion using the distributed algorithm proposed in [6].

Example 2 *In our example, the MCS that represents the four robots has two equilibria, S_0 and S_1. These represent, respectively, coalitions C_0 and C_1, with which the robots can achieve their goals..*

$$\begin{aligned} S_0 = &\{\{a_{2s}, a_{1d}, a_{3d}, a_{3c}, g_3\}, \{a_{1s}, a_{4d}, a_{2c}, g_2\}, \\ &\{a_{4s}, a_{2d}, a_{4c}, g_4\}, \{a_{3s}, a_{1c}, g_1\}\} \\ S_1 = &\{\{a_{2s}, a_{1d}, a_{3d}, a_{1c}, g_1\}, \{a_{1s}, a_{4d}, a_{2c}, g_2\}, \\ &\{a_{4s}, a_{2d}, a_{3c}, g_3\}, \{a_{3s}, a_{4c}, g_4\}\} \end{aligned}$$

Typically, efficiency and stability are chosen to evaluate coalitions. For example, using the values in Figure 1 we can compute the costs (total distances) of each task in each coalition. C_0 is economically more efficient as at least one robot is better off without making anyone worse off, all else being equal. C_0 is also more cost efficient than C_1, as its total cost (81) is lower than the cost of C_1 (87). By applying the metrics used in [3], we can also compare coalitions in terms of conviviality, measured by the number of reciprocity based coalitions that can be formed: The conviviality of C_1 (0.000143) is greater than that of C_0 (0.0000897), and therefore C_1 is preferred.

4 Summary and Outlook

Compared to most previous works on multirobot cooperation, which focus on single-robot, independent tasks, we address multi-robot tasks, which require multiple robots to cooperate by forming coalitions. Other works from different areas have handled the same problem including variants of the contract net protocol [9, 11]; formal approaches from multiagent systems [10, 13, 14]; and solutions from robotics based on schema theory [16, 17] or synergy [12]. The contributions of our MCS approach are: (a) it can represent robots with different knowledge representation models and different kinds of interrobot relationships such as dependencies, constraints and conflicting goals; and (b) it allows to compute coalitions either in a centralized or in a distributed way, and can therefore be applied in settings with different requirements for information hiding and sharing.

The next step for this work is to integrate in our model preferences on robots and goals and develop algorithms for preference-based coalition formation in the presence of conflicting goals, based on previous work on inconsistency resolution in MCS [1, 4, 7, 8]. We also plan to extend our approach with elements of dynamic MCS [5], i.e. schematic bridge rules that are instantiated at run time with concrete contexts. This will enable handling changes such as the failure of a robot, the arrival of a new robot or any change in the operating environment. We will apply and test our methods in different kinds of robot-based systems, such as Ambient Intelligence systems.

Acknowledgments

The present research is supported by the National Research Fund, Luxembourg, CoPAInS project (code: CO11/IS/1239572).

REFERENCES

[1] Antonis Bikakis, Grigoris Antoniou, and Panayiotis Hassapis, 'Strategies for contextual reasoning with conflicts in Ambient Intelligence', *Knowledge and Information Systems*, **27**(1), 45–84, (2011).
[2] Gerhard Brewka and Thomas Eiter, 'Equilibria in Heterogeneous Nonmonotonic Multi-Context Systems', in *AAAI*, pp. 385–390, (2007).
[3] Patrice Caire, Baptiste Alcade, Leendert van der Torre, and Chattrakul Sombattheera, 'Conviviality measures', in *10th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011*, (2011).
[4] Patrice Caire, Antonis Bikakis, and Yves Le Traon, 'Information Dependencies in MCS: Conviviality-Based Model and Metrics', in *PRIMA*, pp. 405–412, (2013).
[5] Minh Dao-Tran, Thomas Eiter, Michael Fink, and Thomas Krennwallner, 'Dynamic distributed nonmonotonic multi-context systems', *Nonmonotonic Reasoning, Essays Celebrating its 30th Anniversary, Studies in Logic*, **31**.
[6] Minh Dao-Tran, Thomas Eiter, Michael Fink, and Thomas Krennwallner, 'Distributed nonmonotonic multi-context systems', in *KR*, (2010).
[7] Thomas Eiter, Michael Fink, Peter Schüller, and Antonius Weinzierl, 'Finding Explanations of Inconsistency in Multi-Context Systems', in *Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010*. AAAI Press, (2010).
[8] Thomas Eiter, Michael Fink, and Antonius Weinzierl, 'Preference-Based Inconsistency Assessment in Multi-Context Systems', in *Logics in Artificial Intelligence - 12th European Conference, JELIA 2010, Helsinki, Finland, September 13-15, 2010. Proceedings*, volume 6341 of *Lecture Notes in Computer Science*, pp. 143–155. Springer, (2010).
[9] Brian P. Gerkey and Maja J. Matarić, 'Sold!: Auction methods for multi-robot coordination', *IEEE Transactions on Robotics and Automation, Special Issue on Multi-Robot Systems*, **18**(5), 758–768, (Oct 2002). (Also Technical Report IRIS-01-399).
[10] Matthias Klusch and Andreas Gerber, 'Dynamic coalition formation among rational agents.', *IEEE Intelligent Systems*, **17**(3), 42–47, (2002).
[11] Thomas Lemaire, Rachid Alami, and Simon Lacroix, 'A distributed tasks allocation scheme in multi-uav context', in *ICRA*, pp. 3622–3627, (2004).
[12] Somchaya Liemhetcharat and Manuela M. Veloso, 'Weighted synergy graphs for effective team formation with heterogeneous ad hoc agents', *Artif. Intell.*, **208**, 41–65, (2014).
[13] Tuomas Sandholm, Kate Larson, Martin Andersson, Onn Shehory, and Fernando Tohmé, 'Coalition structure generation with worst case guarantees', *Artif. Intell.*, **111**(1-2), 209–238, (July 1999).
[14] Onn Shehory and Sarit Kraus, 'Methods for task allocation via agent coalition formation', *Artif. Intell.*, **101**(1-2), 165–200, (May 1998).
[15] Jaime Simao Sichman and Yves Demazeau, 'On social reasoning in multi-agent systems', *Revista Iberoamericana de Inteligencia Artificial*, **13**, 68–84, (2001).
[16] Fang Tang and Lynne E. Parker, 'Asymtre: Automated synthesis of multi-robot task solutions through software reconfiguration', in *ICRA*, pp. 1501–1508, (2005).
[17] Yu Zhang and Lynne E. Parker, 'Iq-asymtre: Forming executable coalitions for tightly coupled multirobot tasks', *IEEE Transactions on Robotics*, **29**(2), 400–416, (2013).