A Multiagent Path Search Algorithm for Large-Scale Coalition Structure Generation

Extended Abstract

Redha Taguelmimt
Univ Lyon, UCBL, CNRS, INSA Lyon, Centrale Lyon, Univ Lyon 2, LIRIS, UMR5205
F-69622 Villeurbanne, France
redha.taguelmimt@gmail.com

Samir Aknine
Univ Lyon, UCBL, CNRS, INSA Lyon, Centrale Lyon, Univ Lyon 2, LIRIS, UMR5205
F-69622 Villeurbanne, France
samir.aknine@univ-lyon1.fr

Djamila Boukredera
University of Bejaia, Faculty of Exact Sciences, Laboratory of Applied Mathematics
06000 Bejaia, Algeria
djamila.boukredera@univ-bejaia.dz

Narayan Changder
TCG Centres for Research and Education in Science and Technology
Kolkata, India
narayan.changder@tcgcrest.org

Tuomas Sandholm
Carnegie Mellon University
Strategic Machine, Inc.
Strategy Robot, Inc.
Optimized Markets, Inc.
Pittsburgh, USA
sandholm@cs.cmu.edu

ABSTRACT

Coalition structure generation (CSG) is a critical problem in multiagent systems, involving the optimal partitioning of agents into disjoint coalitions to maximize social welfare. This paper introduces SALDAE, a novel multiagent path finding algorithm for CSG on a coalition structure graph. SALDAE employs various heuristics and strategies for efficient search, making it an anytime algorithm suitable for handling large-scale problems.

KEYWORDS

Coalition formation; Coalition structure generation; Multiagent path finding

ACM Reference Format:
Redha Taguelmimt, Samir Aknine, Djamila Boukredera, Narayan Changder, and Tuomas Sandholm. 2024. A Multiagent Path Search Algorithm for Large-Scale Coalition Structure Generation: Extended Abstract. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 3 pages.

1 INTRODUCTION

Coalition structure generation is a major problem in artificial intelligence that is central to many practical applications, including transportation [9], disaster response [15], distributed sensor networks [4], and e-commerce [14]. Several algorithms have been proposed for this problem, including optimal and approximate solutions [1–3, 5–8, 10, 12, 13, 15, 16]. In this paper we present SALDAE, a new algorithm to search the coalition structure graph with multiagent path finding. It is anytime and scales to hundreds and thousands of agents. A CSG problem defined on a set of n agents $\mathcal{A} = \{a_1, a_2, ..., a_n\}$ is a problem of size n. A coalition C is any non-empty subset of $\mathcal{A}$. In CSG, a characteristic function v assigns a real value to each coalition. This value determines the efficiency of the coalition. A coalition structure $\mathcal{CS}$ is a partition of the set of agents $\mathcal{A}$ into disjoint coalitions. Formally, given a set of non-empty coalitions $\{C_1, C_2, ..., C_k\}$, $\mathcal{CS} = \{C_1, C_2, ..., C_k\}$, where $k = |\mathcal{CS}|$ and $\mathcal{CS}$ satisfies the following constraints: $\bigcup_{j=1}^{k} C_i = \mathcal{A}$ and for all $i, j \in \{1, 2, ..., k\}$ where $i \neq j$, $C_i \cap C_j = \emptyset$. The value of $\mathcal{CS}$ is assessed as: $v(\mathcal{CS}) = \sum_{C \in \mathcal{CS}} v(C)$. The goal in CSG is to find the optimal solution $\mathcal{CS}^* = \arg\max_{\mathcal{CS} \in \Pi(\mathcal{A})} v(\mathcal{CS})$. The coalition structure graph (see Figure 1) divides the solutions into levels and each level l contains coalition structures that are formed of exactly l coalitions.

2 THE SALDAE ALGORITHM

In this paper, we introduce a novel path finding variant where each node represents a solution to the coalition structure generation problem—that is, a coalition structure. The objective is to identify the optimal or near-optimal coalition structure by navigating a graph with a search agent. The agent, starting at a designated node, traverses adjacent nodes, maintaining a sorted list based on coalition structure values.

Multiagent Path Finding has several variants, and our goal is to find the optimal solution using multiple search agents in the CSG context. Conflicts may arise when multiple agents evaluate the same coalition structure. While the paths taken by agents are crucial for finding high-quality solutions quickly, the optimal solution is the highest-valued coalition structure found among the nodes.

Our proposed path search algorithm uses a gradually constructed search graph (refer to Figure 1), starting from a designated start node representing a coalition structure. Each parent node in the graph connects to child nodes generated through coalition splitting or merging operations. The algorithm iteratively builds the search graph using a greedy approach.

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

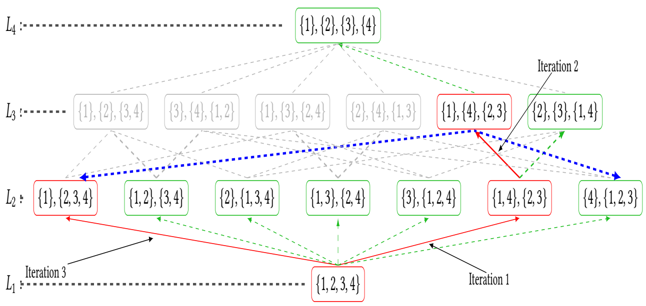

Figure 1: An illustration of the three phases of our algorithm. In the first iteration, the start node is the bottom node. The child nodes are represented by the nodes that are directly connected to the bottom node. The coalition structure $\{\{a_1, a_4\}, \{a_2, a_3\}\}$ in level 2 is assumed to have the highest value, becoming the new start node. Subsequent iterations involve generating child nodes, with selection based on coalition structure values. Blue edges represent the generation of coalition structure by merging two coalitions.

- Generation (Step 1): Child coalition structures are generated from the current start node by either splitting a coalition or joining two coalitions.
- Selection (Step 2): A start node selection procedure considers the most promising child coalition structure among a set of candidates generated in the previous step. The start node is chosen as the highest-valued child coalition structure.
- Comparison to Incumbent (Step 3): The selected coalition structure is evaluated, and if superior to the current best solution, it becomes the new best solution.

The construction of the search graph continues iteratively, with each node representing a potential solution. The algorithm's search process is illustrated through a 4-agent example (Figure 1), highlighting the generation, selection, and comparison steps.

The algorithm dynamically explores solutions by maintaining a list of nodes and iteratively expanding them based on the three fundamental steps. In addition to these steps, whenever a better solution is found, the algorithm constructs a path of nodes connecting the previous best solution to the new one. This path exploration aims to uncover potentially better solutions by evaluating intermediate solutions between the two. The rationale lies in the correlation between the distribution of coalitions and agents in current best solutions, as slight changes in these structures could lead to improved solutions.

SALDAE, the proposed algorithm, also employs memory management strategies that enhance its efficiency in seeking global optima. It maintains three lists of nodes: OPEN, SUBSTITUTE, and RESERVE. OPEN, constructed through the search process with the 3 steps, is sorted in descending order based on the values of coalition structures. SUBSTITUTE holds nodes visited during the path search between current best solutions, only expanded if they can improve the current solution, and RESERVE stores less promising nodes. The key strategies for memory management are:

(1) OPEN Replacement:
   - If OPEN becomes empty, SUBSTITUTE replaces OPEN.
   - Facilitates exploration of potentially more promising solution areas.

(2) OPEN to RESERVE Transition:
   - If OPEN and SUBSTITUTE are empty, RESERVE replaces OPEN.
   - Prioritizes promising solution areas and reverts to RESERVE if more promising areas are not found.

These memory management strategies empower SALDAE to adaptively explore and focus on promising areas in the solution space.

SALDAE accelerates the search of high-quality solutions by employing multiple search agents, each managing three lists: OPEN, RESERVE, and SUBSTITUTE. Agents navigate from node to node, aiming to enhance solutions. The key features are as follows:

(1) Agent Configuration:
   - Two agents assigned to bottom and top nodes, with random nodes for other agents.
   - Agents move through nodes, expanding and generating child nodes.

(2) Conflict Resolution: Conflicts arise when agents evaluate the same coalition structure. We consider two conflict resolution techniques:
   - Bypassing Conflicts: Prevents using conflicting child nodes found in other agents' lists and ensures agents explore distinct areas of the solution space.
   - Managing Conflicts: Compares rankings of conflicting child nodes in respective lists and allows the agent with higher ranking to keep and expand the node.

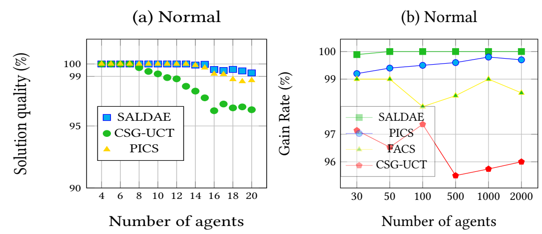

Figure 2: Solution quality and gain rate of SALDAE, PICS, FACS and CSG-UCT.

3 EMPIRICAL EVALUATION

A comprehensive evaluation of SALDAE's performance is conducted, covering small-scale and large-scale benchmarks. SALDAE consistently outperforms other algorithms, such as PICS [12], FACS [11], and CSG-UCT [15], in terms of solution quality (Figure 2.a) and gain rate (Figure 2.b). It provides optimal solutions more frequently and achieves higher solution quality compared to the other algorithms. The evaluation also shows that SALDAE can handle large problems with hundreds and thousands of agents.

ACKNOWLEDGMENTS

Tuomas Sandholm is supported by the Vannevar Bush Faculty Fellowship ONR N00014-23-1-2876, National Science Foundation grants RI-2312342 and RI-1901403, and ARO award W911NF2210266.

REFERENCES

[1] Narayan Changder, Samir Aknine, and Animesh Dutta. 2019. An Imperfect Algorithm for Coalition Structure Generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9923–9924.

[2] Narayan Changder, Samir Aknine, Sarvapali D Ramchurn, and Animesh Dutta. 2020. ODSS: Efficient Hybridization for Optimal Coalition Structure Generation.. In *Proc. of AAAI*. 7079–7086.

[3] Narayan Changder, Samir Aknine, Sarvapali D. Ramchurn, and Animesh Dutta. 2021. BOSS: A Bi-directional Search Technique for Optimal Coalition Structure Generation with Minimal Overlapping (Student Abstract). In *Proc. of AAAI*, Vol. 35. 15765–15766. https://ojs.aaai.org/index.php/AAAI/article/view/17879

[4] Viet Dung Dang, Rajdeep K Dash, Alex Rogers, and Nicholas R Jennings. 2006. Overlapping coalition formation for efficient data fusion in multi-sensor networks. In *Proc. of AAAI*, Vol. 6. 635–640.

[5] Tomasz Michalak, Talal Rahwan, Edith Elkind, Michael Wooldridge, and Nicholas R Jennings. 2016. A hybrid exact algorithm for complete set partitioning. *Artificial Intelligence* 230 (2016), 14–50.

[6] Talal Rahwan and Nicholas R Jennings. 2008. An improved dynamic programming algorithm for coalition structure generation. In *Proc. of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 3*. International Foundation for Autonomous Agents and Multiagent Systems, 1417–1420.

[7] Talal Rahwan, Tomasz P Michalak, Michael Wooldridge, and Nicholas R Jennings. 2015. Coalition structure generation: A survey. *Artificial Intelligence* 229 (2015), 139–174.

[8] Tuomas Sandholm, Kate Larson, Martin Andersson, Onn Shehory, and Fernando Tohmé. 1999. Coalition structure generation with worst case guarantees. *Artificial Intelligence* 111, 1 (1999), 209–238.

[9] Tuomas Sandholm and Victor R Lesser. 1997. Coalitions among computationally bounded agents. *Artificial intelligence* 94, 1 (1997), 99–138.

[10] Redha Taguelmimt, Samir Aknine, Djamila Boukredera, and Narayan Changder. 2021. Code-based Algorithm for Coalition Structure Generation. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*. 1075–1082.

[11] Redha Taguelmimt, Samir Aknine, Djamila Boukredera, and Narayan Changder. 2021. FACS: Fast Code-based Algorithm for Coalition Structure Generation (Student Abstract). *Proceedings of the AAAI Conference on Artificial Intelligence* 35, 18 (May 2021), 15907–15908. https://doi.org/10.1609/aaai.v35i18.17950

[12] Redha Taguelmimt, Samir Aknine, Djamila Boukredera, and Narayan Changder. 2022. PICS: Parallel Index-based Search Algorithm for Coalition Structure Generation. In *2022 IEEE 34th International Conference on Tools with Artificial Intelligence (ICTAI)*. 739–746.

[13] Redha Taguelmimt, Samir Aknine, Djamila Boukredera, Narayan Changder, and Tuomas Sandholm. 2023. Optimal Anytime Coalition Structure Generation Utilizing Compact Solution Space Representation. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, Edith Elkind (Ed.). International Joint Conferences on Artificial Intelligence Organization, 309–316. https://doi.org/10.24963/ijcai.2023/35 Main Track.

[14] Maksim Tsvetovat and Katia Sycara. 2000. Customer coalitions in the electronic marketplace. In *Proc. of the fourth international conference on Autonomous agents*. 263–264.

[15] Feng Wu and Sarvapali D Ramchurn. 2020. Monte-Carlo Tree Search for Scalable Coalition Formation. In *Proc. of IJCAI*. 407–413.

[16] D Yun Yeh. 1986. A dynamic programming approach to the complete set partitioning problem. *BIT Numerical Mathematics* 26, 4 (1986), 467–474.