Constructing Junction Tree Agent Organization with Privacy

JAAMAS Track

Yang Xiang
University of Guelph
yxiang@uoguelph.ca

Abdulrahman Alshememry
King Saud University
akalshememry@ksu.edu.sa

ABSTRACT

Several frameworks for decentralized reasoning assume a junction tree agent organization (JT-org). JT-org construction involves 3 related tasks on existence recognition, construction, and environment re-decomposition, where re-decomposition incurs loss of JT-org linked privacy, including privacy on agent, topology, private and shared variables. We propose a novel algorithm DAER that accomplishes all 3 tasks distributively. For Tasks 1 and 2, DAER incurs no loss of JT-org linked privacy. For Task 3, it incurs significantly less privacy loss than existing JT-org construction methods. Its performance is formally analyzed and empirically evaluated [1].

KEYWORDS

Coordination model for MAS, Self-organization, Privacy in MAS

ACM Reference Format:

Yang Xiang and Abdulrahman Alshememry. 2021. Constructing Junction Tree Agent Organization with Privacy: JAAMAS Track. 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

Decentralized probabilistic, constraint-based, and decision theoretic reasonings are essential tasks of cooperative multiagent systems (MAS). Many frameworks exist: Some do not assume specific organization, e.g., [5, 13]. Some assume a total order among agents, e.g., [3, 8]. Some are based on a lattice, e.g., [10]. Some use a pseudotree, e.g., [4, 6, 11, 12]. Multiply Sectioned Bayesian Networks (MSBNs) [15, 17] are earliest MAS based on JT-orgs. JT-orgs have now been applied to other frameworks, e.g., distributed constraint optimization [2, 14] and decentralized decision theoretic reasoning [19]. It has been shown [14] that JT-orgs are superior than pseudotrees. More generally, for a number of distinct information processing tasks, JT-orgs are shown to guarantee exact answers [1].

Privacy is important in MAS [9, 21], e.g. four types of privacy are identified [7] in distributed constraint reasoning on agent, topology, constraint, and decision. Very few works exist on protecting privacy during JT-org construction. Construction techniques employed by several frameworks that depend on JT-orgs, e.g., [2, 14, 16], compromise privacy on private and shared variables, agent identities and adjacency relations, as shown in [20]. The above four types of privacy from distributed constraint reasoning do not adequately capture these privacy losses. First, JT-org construction does not involve constraint and decision privacy. Second, although loss on agent identities maps to agent privacy, and loss on agent adjacency relations maps to topology privacy, privacy losses on private and shared variables are not covered by any of the four types.

We refer to overall privacy related to JT-org construction as *JT-org linked privacy*. It includes four specific types: We refer to those on agent identities and adjacency relations as *agent privacy* and *topology privacy*. We refer to the remaining two types as *privacy on private variables* and *privacy on shared variables*.

A natural decomposition of agent environment (env) may not admit a JT-org. Hence, JT-org construction involves three related tasks: (1) Recognize whether a JT-org exists for a given env decomposition. (2) When JT-orgs exist, construct one. (3) If no JT-org exists, revise env decomposition so that one exists and then construct it.

HTBS is an algorithm that performs Tasks 1 and 2 [20] without JT-org linked privacy loss. When env admits JT-orgs, HTBS is superior than alternative construction methods, such as those in Action-GDL [14] and DCTE [2], which incur JT-org linked privacy loss. Action-GDL and DCTE do not perform Task 1 explicitly. When no JT-orgs exist for the given env decomposition, methods in Action-GDL and DCTE perform Task 3, while incurring privacy loss. HTBS, on the other hand, terminates after Task 1, without constructing a JT-org.

The main contribution is a novel algorithm DAER that integrates and significantly extends HTBS to accomplish all three tasks. When agent env decomposition admits no JT-orgs, DAER performs Task 3 by modifying the decomposition, as Action-GDL and DCTE do, but with considerably less JT-org linked privacy loss. This advancement significantly improves privacy in JT-org based MAS, such as MSBN [15], Action-GDL [14], and DCTE [2].

2 BACKGROUND

Consider a set $A = \{A_0, ..., A_{\eta-1}\}$ of cooperative agents, whose environment is described by a collection V of variables. The *environment* V is decomposed into a set of overlapping *subenvironments* (subenv) $\Omega = \{V_0, ..., V_{\eta-1}\}$, where $\cup_{i=0}^{\eta-1} V_i = V$, such that agent A_i controls V_i. A variable that appears in a unique subenv V_i is a *private* variable. Otherwise, it is a *shared* variable. If A_i and A_j ($j \neq i$) share variables, their *border* is the set of variables that they share, $I_{ij} = V_i \cap V_j \neq \emptyset$, and the two agents are *adjacent*.

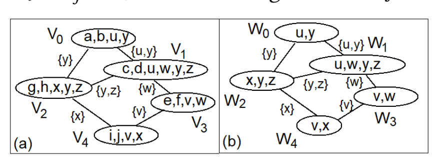

Figure 1: Env dec cluster graph (a) and commu graph (b)

Env decomposition Ω can be depicted by an *env decomposition cluster graph* (Fig. 1 (a)), where each cluster is a subenv and each link between two clusters is labeled by their border.

[1]This is an extended abstract of [18].

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.

For distributed probabilistic reasoning, knowledge over each V_i can be encoded as a Bayes subnet, e.g., MSBN [17]. For decentralized constraint optimization, constraints over each V_i can be encoded as a constraint subnet, e.g., Action-GDL [14] and DCTE [2].

Boundary of an agent is the union of its borders. *Boundary set* of a MAS is the collection of boundaries of its agents. A boundary set can be depicted by a cluster graph, called *communication graph* (CG), where each cluster is a boundary and each link between two clusters is a border (Fig. 1 (b)). CG involves shared variables only.

A JT-org is a tree subgraph (including all clusters) of env decomposition cluster graph, such that intersection of any two subenv is contained in every subenv on the path between the two. Fig. 2 (b) is a JT-org of (a). A JT-org can be expressed (without private variables) as a tree subgraph of communication graph, as in (c).

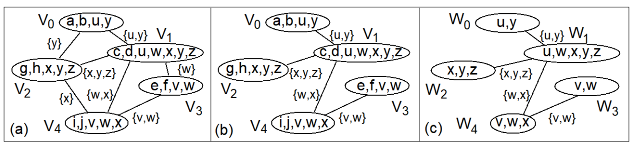

Figure 2: Env decomposition cluster graph (a), JT-org (b), and JT-org over boundaries (c)

An env decomposition may not admit a JT-org, e.g., subenvs of Fig. 1 (a) cannot be organized into a JT-org. Hence, construction of JT-org involves the three related tasks in Section 1.

HTBS is an algorithm for Tasks 1 and 2 without privacy loss [20]. It consists of recursive agent self-elimination. Agent A_i with boundary W_i can *self-eliminate relative to* adjacent A_j with W_j, if W_i equals their border $W_i = I_{ij}$. After A_i is eliminated, W_j is updated by removing any variable that A_j uniquely shared with A_i.

3 ENV RE-DECOMPOSITION BY DAER

For a MAS based on JT-org, if env decomposition does not admit a JT-org, it must be revised. Assuming that no variable can be removed, the only option is to share some variables beyond original scope, resulting in privacy loss. If they are shared with non-adjacent agents, privacy loss occurs on agent identity, as well as on agent adjacency relations. The challenge is to minimize such loss.

We present DAER, a novel algorithm, that integrates and significantly extends HTBS to accomplish all 3 tasks of JT-org construction distributively. DAER performs Tasks 1 and 2 without privacy loss. When agent env decomposition admits no JT-org, DAER re-decomposes env and constructs a JT-org, with significantly lower privacy loss than Action-GDL and DCTE. It differs from methods in Action-GDL and DCTE: (1) DAER operates on CGs, rather than env decomposition cluster graphs. Hence, DAER is free from privacy loss on private variables. (2) DAER is based on a numerical evaluation of privacy loss, so that agent developers can influence privacy loss and trade among different types of privacy loss.

DAER assumes reliable communication: no message loss, and receiving in order of sending. Agents are honest but curious: They follow intended protocol, but are interested in learning private information of other agents from messages.

Given an agent environment (A, Ω, W), DAER runs in multiple rounds. Each round consists of an HTBS stage and an elimination-expansion (EE) stage. During HTBS stage, agents self-eliminate (becoming inactive), until no such elimination is possible. If a single active agent is left, then (A, Ω, W) admits a JT-org that emerges distributively [20], completing Tasks 1 and 2. Otherwise, (A, Ω, W) does not admit JT-org (Task 1) and EE stage starts, in which only remaining active agents participate. The EE stage of the first round and remaining rounds of DAER perform Task 3.

During EE stage, active agents operate on a reduced CG without boundaries of eliminated agents. During the EE stage, each active agent generates a boundary expansion plan that enables self-elimination and has the minimum privacy loss among alternative (local) plans. Subsequently, active agents select a best (global) expansion plan through a distributed depth-first-search (DFS) over the CG. A best plan is one with the (global) minimum privacy loss. The winning agent then performs the expansion and self-eliminates, terminating the current round of DAER.

Each new round of DAER repeats the HTBS stage and EE stage. When the HTBS stags ends with a single active agent, the new env decomposition defined by all boundary expansions performed so far admits a JT-org that emerges distributively.

The boundary expansion plan that an agent A_i generates is selected locally among alternatives, one per neighbor agent A_k. For each A_k, there exists a boundary expansion that enables self-elimination of A_i by sharing a mininum number of private variables. To evaluate its privacy loss, a numerical measure is desirable.

JT-org construction may suffer privacy loss on private and shared variables, agent identities and adjacency relations. Few measures exist to evaluate such loss. Agent identities are assumed publicly known in [9]. Four types of privacy are identified in [7], but JT-org does not involve constraint and decision privacy, and the 4 types do not cover loss on private and shared variables. Leak of each piece of private info counts 1 unit to total loss in [20]. It does not admit difference in info sensitivity. It is suited for loss evaluation at system level, not agent level. We develop a new measure of privacy loss: (1) It allows agents to trade off disclosure of private info of different sensitivity, and (2) privacy loss of each boundary expansion is evaluated locally.

We formally show that DAER accomplishes all 3 tasks of JT-org construction, and upon termination, the final expanded env decomposition has a JT-org, while privacy loss is greedily minimized.

4 EXPERIMENTAL EVALUATION

To evaluate effectiveness of DAER in preserving JT-org linked privacy and its efficiency, we empirically compare DAER with alternative distributed JT-org construction methods in ActionGDL and DCTE. DAER dominates the alternative methods with significantly lower privacy loss (up to 4 orders of magnitude relative to DCTE). It is categorically superior over alternative methods by being immune to privacy loss over private variables. DAER is efficient with linear time on the number of agents and the number of adjacent agent pairs. Experimentally, its runtime is significantly less than alternative methods across a range of values for number of agents, size of subenvs, and ratio of private variables.

ACKNOWLEDGMENTS

Support from NSERC Discovery Grant and Scholarship from Saudi Arabian Cultural Bureau are acknowledged.

REFERENCES

[1] S.M. Aji and R.J. McEliece. 2000. The generalized distributive law. *IEEE Trans. Information Theory* 46, 2 (2000), 325–343.

[2] I. Brito and P. Meseguer. 2010. Cluster tree elimination for distributed constraint optimization with quality guarantees. *Fundamenta Informaticae* 102, 3-4 (2010), 263–286.

[3] Y. Gao, F. Toni, H. Wang, and F. Xu. 2016. Argumentation-Based Multi-Agent Decision Making with Privacy Preserved. In *Proc. 15th Inter. Conf. on Autonomous Agents and Multiagent Systems*, J. Thangarajah, K. Tuyls, C. Jonker, and S. Marsella (Eds.). 1153–1161.

[4] K.D. Hoang, F. Fioretto, P. Hou, M. Yokoo, W. Yeoh, and R. Zivan. 2016. Proactive Dynamic Distributed Constraint Optimization. In *Proc. 15th Inter. Conf. on Autonomous Agents and Multiagent Systems*, J. Thangarajah, K. Tuyls, C. Jonker, and S. Marsella (Eds.). 597–605.

[5] D. Koller and B. Milch. 2001. Multi-agent influence diagrams for representing and solving games. In *Proc. 17th Inter. Joint Conf. on Artificial Intelligence*. 1027–1034.

[6] T. Le, F. Fioretto, W. Yeoh, T.C. Son, and E. Pontelli. 2016. ER-DCOPs: A Framework for Distributed Constraint Optimization with Uncertainty in Constraint Utilities. In *Proc. 15th Inter. Conf. on Autonomous Agents and Multiagent Systems*, J. Thangarajah, K. Tuyls, C. Jonker, and S. Marsella (Eds.). 606–614.

[7] T. Leaute and B. Faltings. 2013. Protecting privacy through fistributed vomputation in multi-agent decision making. *J. Artificial Intelligence Research* 47 (2013), 649–695.

[8] A. Maestre and C. Bessiere. 2004. Improving Asynchronous Backtracking for Dealing with Complex Local Problems. In *Proc. 16th European Conf. on Artificial Intelligence*. 206–210.

[9] R.T. Maheswaran, J.P. Pearce, E. Bowring, P. Varakantham, and M. Tambe. 2006. Privacy Loss in distributed constraint reasoning: a quantitative framework for analysis and its applications. *J. Autonomous Agents and Multi-Agent Systems* 13, 1 (2006), 27–60.

[10] S. Mahmoud, S. Miles, and M. Luck. 2016. Cooperation Emergence under Resource-Constrained Peer Punishment. In *Proc. 15th Inter. Conf. on Autonomous Agents and Multiagent Systems*, J. Thangarajah, K. Tuyls, C. Jonker, and S. Marsella (Eds.). 900–908.

[11] P.J. Modi, W. Shen, M. Tambe, and M. Yokoo. 2005. Adopt: asynchronous distributed constraint optimization with quality guarantees. *Artificial Intelligences* 161, 1-2 (2005), 149–180.

[12] A. Petcu and B. Faltings. 2005. A scalable method for multiagent constraint optimization. In *Proc. 19th Inter. Joint Conf. on Artificial Intelligence*. 266–271.

[13] M. Valtorta, Y.G. Kim, and J. Vomlel. 2002. Soft evidential update for probabilistic multiagent systems. *Int. J. Approximate Reasoning* 29, 1 (2002), 71–106.

[14] M. Vinyals, J.A. Rodriguez-Aguilar, and J. Cerquides. 2010. Constructing a unifying theory of dynamic programming DCOP algorithms via the generalized distributive law. *J. Autonomous Agents and Multi-Agent Systems* 22, 3 (2010), 439–464.

[15] Yang Xiang. 1996. A Probabilistic Framework for Cooperative Multi-agent Distributed Interpretation and Optimization of Communication. *Artificial Intelligence* 87, 1-2 (1996), 295–342.

[16] Yang Xiang. 2002. *Probabilistic Reasoning in Multiagent Systems: A Graphical Models Approach*. Cambridge University Press, Cambridge, UK.

[17] Yang Xiang. 2008. Building Intelligent Sensor Networks With Multiagent Graphical Models. In *Intelligent Decision Making: An AI-Based Approach*, N. Ichalkaranje G.P. Wren and L.C. Jain (Eds.). Springer-Verlag, 289–320.

[18] Yang Xiang and Abdulrahman Alshememry. 2020. Privacy sensitive environment re-decomposition for junction tree agent organization construction. *Autonomous Agents and Multi-Agent Systems* 34, 15 (2020), https://doi.org/10.1007/s10458–019–09438–6.

[19] Yang Xiang and Frank Hanshar. 2015. Multiagent Decision Making in Collaborative Decision Networks by Utility Cluster Based Partial Evaluation. *Inter. J. Uncertainty, Fuzziness and Knowledge-Based Systems* 23, 2 (2015), 149–191.

[20] Yang Xiang and Kamala Srinivasan. 2016. Privacy Preserving Existence Recognition and Construction of Hypertree Agent Organization. *J. Autonomous Agents and Multi-Agent Systems* 30, 2 (2016), 220–258.

[21] M. Yokoo, K. Suzuki, and K. Hirayama. 2005. Secure distributed constraint satisfaction: reaching agreement without revealing private information. *Artificial Intelligence* 161 (2005), 229–246.