Towards Multi-Agent Learning of Causal Networks

Extended Abstract

Stefano Mariani
University of Modena and Reggio Emilia
Reggio Emilia, Italy
stefano.mariani@unimore.it

Pasquale Roseti
University of Modena and Reggio Emilia
Reggio Emilia, Italy
229627@studenti.unimore.it

Franco Zambonelli
University of Modena and Reggio Emilia
Reggio Emilia, Italy
franco.zambonelli@unimore.it

ABSTRACT

We propose a multi-agent protocol for distributed learning of causal networks, aimed both at *(i)* reducing the complexity of learning large causal networks and *(ii)* letting agents in a MAS cooperate to unveil causal relationships that individuals could not reveal by themselves, due to partial observability.

KEYWORDS

Multi-agent learning; Causal networks; Coordination

ACM Reference Format:

Stefano Mariani, Pasquale Roseti, and Franco Zambonelli. 2023. Towards Multi-Agent Learning of Causal Networks: Extended Abstract. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 3 pages.

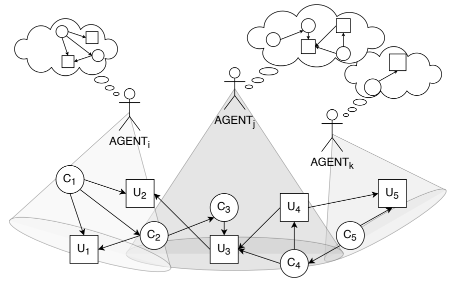

Figure 1: Multiple agents with partial observability over a shared environment want to learn their causal networks.

1 INTRODUCTION

Causal networks aim at modelling cause-effect relationships between variables of a domain of interest. For agents, learning the causal model of the environment in which they operate is fundamental to properly decide plans of actions and being able to explain such decisions [6]. Learning causal network, though, is typically out of reach for statistical machine learning models [11], which rely on purely *observational* data and ignore the notion of *interventions*.

Interventions are the fundamental mechanisms to check whether two variables are linked in a cause-effect relationship and not by a mere correlation [10]. Intuitively, an intervention deliberately changes the value of a variable, all others being untouched, to see whether it affects others. The intervention operation has been formalized with the introduction of the do-calculus by Pearl [9]: $do(C = c)$ means that the controlled variable C is forced to take value c by an environment action. As an example, by assuming to be able to control (i.e., intervene on) the air conditioning system (A/C), the following simple causal network can be learnt: A/C $\rightarrow$ Temperature. Calling $\mathcal{P}(X)$ the probability distribution of variable X, in fact, we would have:

$$\mathcal{P}(\text{Temperature}) \neq \mathcal{P}(\text{Temperature} \mid \mathbf{do}(\text{A/C=on})) \quad (1)$$

$$\mathcal{P}(\text{A/C}) = \mathcal{P}(\text{A/C} \mid \mathbf{do}(\text{Temperature=t})) \quad (2)$$

where t is any temperature value. That is, the status of the temperature does not affect the status of the A/C, but the other way around (as the causal network expresses). The intervention operation clearly cannot apply to *uncontrolled* variables, yet techniques have been identified to be able to learn causal relationships from merely observational data [3, 13].

The process of learning causal networks, in most existing proposals, assumes a *centralized* setting, with a single learner/agent having full observability and control over the environment variables, or with a central site aggregating distributed observations or merging local models [2–4, 13]. This makes such proposals hardly scalable. In addition, the centralized assumption can rarely hold for MAS, where multiple agents are typically deployed in different regions of an environment [7, 8]. Take as reference the situation in Figure 1, where multiple agents have partial observability (U, i.e. sensors) and control (C, i.e. actuators) over the variables of a shared environments. The need for agent collaboration in causal learning is readily explained: agent i cannot fully explain the values it sees for variable U_2, as it is influenced (also) by variable U_3 that agent i is not even aware of. Or, agent k is not fully aware of the implications of its operations on variable C_5, as it influences variable C_4 that agent k is not aware of.

Against this background, we developed a protocol that let agents interact in order to each build a complete view of their underlying causal network.

2 MULTI-AGENT LEARNING PROTOCOL

Consider N agents $\mathcal{A} = \{\mathcal{A}_i\}, \ i = 1, \ldots, N$ willing to learn a causal network M_i of the relationships between V environment variables, partitioned into two (possibly, empty) sets, one of *controlled* variables C (i.e. corresponding to actuators, where interventions are possible) and one of *uncontrolled* variables U (i.e. sensors, where agents cannot intervene), such that $V = C \cup U$. Each agent knows an algorithm – it does not matter which specific one, but for our implementation we chose ref. [1] – for *independently* learning its own local causal network $\mathcal{L}_i$, that is, the one learnt by relying

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom.

solely on its own known variables P_i (i.e. without our multi-agent protocol). Each agent also has *partial observability of the environment*, that is, is aware of only P_i out of the V variables. Formally: $P_i = C_i \cup U_i,\ P_i \subset V \Rightarrow C_i \subset C, U_i \subset U$. However, no variable in V can be unknown to every agents, that is: $V = \bigcup_{i=1}^{N} P_i$—note that an overlap between the different P_i is admitted.

Main protocol. The main protocol unfolds as follows, where agents are assumed to exploit multiple threads to carry out interactions with others in parallel:

Thread 1 whenever an agent, let's say $\mathcal{A}_i$, recognizes that its own locally learnt causal network $\mathcal{L}_i$ is not correct and complete
  (1) it asks for help by communicating to other agents, let's say $\mathcal{A}_j, j \neq i$, the set of variables it suspects to have missing links—that we call its "frontier" (F_i);
  (2) then collects replies as they come in, and re-starts the single-agent learning algorithm with the newly acquired information (e.g. past observations of the variables);

Thread 2 in parallel, agent $\mathcal{A}_i$ (as well as every other agent $\mathcal{A}_{j \neq i}$, in turn), replies to incoming help requests, if any, by considering each received variable $f \in F_j$
  (1) if it is known to $\mathcal{A}_i$ (i.e. $f \in P_i$), it replies to the help request with what it knows about f, that is, the sub-network $\mathcal{L}_f \subseteq \mathcal{L}_i$ with its links to variables in P_i;
  (2) if it is not known to $\mathcal{A}_i$ (i.e. $f \notin P_i$), the intervention-observation sub-protocol, formalized in the following, starts.

Intervention-observation sub-protocol. This unfolds as follows:

(1) if the unknown variable f, received by $\mathcal{A}_i$ as part of a help request by, let's say, $\mathcal{A}_j$, is controlled by $\mathcal{A}_j$, then $\mathcal{A}_j$ does the interventions on f while $\mathcal{A}_i$ observes its $\mathcal{P}(P_i)$;
(2) if, instead, f is not controlled by $\mathcal{A}_j$, then
  (a) for each $c \in C_i$, it is $\mathcal{A}_i$ that does the interventions on c while $\mathcal{A}_j$ observes changes in its $\mathcal{P}(P_j)$;
  (b) for each $u \in U_i$, $\mathcal{A}_i$ sends a batch $\mathcal{B}_i$ of its past data, that is, its observed values for that variable u, to $\mathcal{A}_j$.

In either case, when interventions are concluded, both agents can correct/complete their own local model $\mathcal{L}$.

It is worth to emphasize here that, since in the real world operating on actuators may require time, and observing the results of actions on sensors' readings may incur delays, to correctly carry out the intervention-observation sub-protocol the agents need to agree on an "intervention time window". During such time, agents need to monitor both controlled and uncontrolled variables, *while a single agent in the whole MAS carries out a single intervention.* As a consequence, and due to the very definition of interventions, the intervention sub-protocol is a critical section calling for *distributed mutual exclusion* among the agents participating in the protocol [12]. In our current implementation, tested on a simulated smart home scenario, we adopted the token ring protocol.

The following *coordination aspects* have also to be addressed:

- let agents become aware of who to ask for help to, in refining the local causal network $\mathcal{L}$. *Gossiping* and the notion of *sphere of influence*, while assuming an externally imposed communication topology with no partitions (e.g. from deployment constraints), for instance, can serve the purpose;

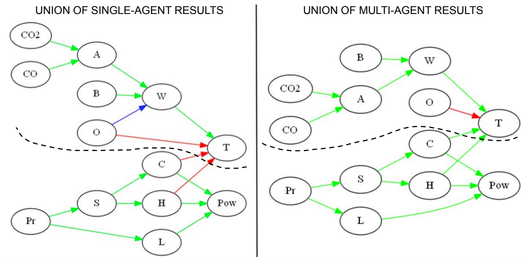

Figure 2: Merge of independently learnt causal networks $\mathcal{U}$ (left) vs. merge of networks refined with multi-agent learning R (right).

- establishing who deserves the attention of other agents when multiple help requests can simultaneously exist. Here, *distributed leader election* algorithms serve the purpose.

3 PRELIMINARY ASSESSMENT

We implemented the proposed multi-agent protocol for causal networks learning, and tested it in a smart home scenario with 2 agents, simulated by adopting the iCasa simulation platform [5]. The simulated smart home enabled both to generate observational data and to perform interventions. The codebase for the learning algorithms, both the single-agent and the multi-agent protocol, is available at https://github.com/smarianimore/multiagent_algorithm, whereas the source code for the iCasa simulation environment is available at https://github.com/smarianimore/iCasa.

The key results of our experiments can be summarized as follows:

- learning *accuracy* is always better with multi-agent learning than with single-agent learning. We measured it with a variation of *structural Hamming distance* (SHD) [14] accounting for "unknown edges", that is, edges missing from the network because they involve variables not known by the learning agent. Formally, SHD = unknown edges + false positives (correlation mistaken for causation) + false negatives (missed causation);
- learning *performance* is always better in the multi-agent case, to an extent increasing as the causal network size and complexity increases (e.g. number of edges and indirect causal paths length). We measured it as "wall clock" time taken by the multi-agent and the single-agent computations.

Figure 2 exemplifies what can be achieved through multi-agent cooperation. Network $\mathcal{U}$ (left side) is the union of two single-agent networks (dotted line expresses partial observability): as the frontier variables causally connecting sub-network $\mathcal{L}_2$ (top) with $\mathcal{L}_1$ (bottom) are not shared by agents, the three arrows connecting $\mathcal{L}_2$ with $\mathcal{L}_1$ are all red, denoting missed causal connections. On the contrary, network R is connected, and only one causal relationships is missing, $O \rightarrow T$. In fact, R also improved accuracy over $\mathcal{U}$, as $SHD(R) = 1 < SHD(\mathcal{U}) = 4$.

Our ongoing and future work will refine the proof-of-concept implementation, add a comparative evaluation of different single-agent algorithms plugged into our multi-agent protocol, and add a more extensive evaluation of performances (accuracy and scalability) for increasing networks sizes and number of agents.

ACKNOWLEDGMENTS

This work has been supported by the MIUR PRIN 2017 Project "Fluidware" (N. 2017KRC7KT).

REFERENCES

[1] Kanvaly Fadiga, Étienne Houzé, Ada Diaconescu, and Jean-Louis Dessalles. 2021. To do or not to do: finding causal relations in smart homes. In *IEEE International Conference on Autonomic Computing and Self-Organizing Systems, ACSOS 2021, Washington, DC, USA, September 27 - Oct. 1, 2021*, Esam El-Araby, Vana Kalogeraki, Danilo Pianini, Frédéric Lassabe, Barry Porter, Sona Ghahremani, Ingrid Nunes, Mohamed Bakhouya, and Sven Tomforde (Eds.). IEEE, 110–119. https://doi.org/10.1109/ACSOS52086.2021.00030

[2] St John Grimbly, Jonathan P. Shock, and Arnu Pretorius. 2021. Causal Multi-Agent Reinforcement Learning: Review and Open Problems. *CoRR* abs/2111.06721 (2021). arXiv:2111.06721 https://arxiv.org/abs/2111.06721

[3] Ruocheng Guo, Lu Cheng, Jundong Li, P. Richard Hahn, and Huan Liu. 2020. A Survey of Learning Causality with Data: Problems and Methods. *ACM Comput. Surv.* 53, 4 (2020), 75:1–75:37. https://doi.org/10.1145/3397269

[4] Christina Heinze-Deml, Marloes H. Maathuis, and Nicolai Meinshausen. 2018. Causal Structure Learning. *Annual Review of Statistics and Its Application* 5, 1 (2018), 371–391. https://doi.org/10.1146/annurev-statistics-031017-100630 arXiv:https://doi.org/10.1146/annurev-statistics-031017-100630

[5] Philippe Lalanda, Catherine Hamon, Clement Escoffier, and Thomas Lévèque. 2014. iCasa, a development and simulation environment for pervasive home applications. In *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*. IEEE, Las Vegas, France, 1142–1143. https://hal.archives-ouvertes.fr/hal-02023545

[6] Marco Lippi, Stefano Mariani, Matteo Martinelli, and Franco Zambonelli. 2022. Individual and Collective Self-Development: Concepts and Challenges. In *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022.* 15–21.

[7] Marco Lippi, Stefano Mariani, and Franco Zambonelli. 2021. Developing a "Sense of Agency" in IoT Systems: Preliminary Experiments in a Smart Home Scenario. In *19th IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events, PerCom Workshops 2021, Kassel, Germany, March 22-26, 2021*. IEEE, 44–49. https://doi.org/10.1109/PerComWorkshops51409.2021.9431003

[8] Matteo Martinelli, Stefano Mariani, Marco Lippi, and Franco Zambonelli. 2022. Self-Development and Causality in Intelligent Environments. In *Workshops at 18th International Conference on Intelligent Environments (IE2022), Biarritz, France, 20-23 June 2022.* 248–257.

[9] Judea Pearl. 2009. *Causality* (2 ed.). Cambridge University Press, Cambridge, UK. https://doi.org/10.1017/CBO9780511803161

[10] Judea Pearl and Dana Mackenzie. 2018. *The Book of Why*. Basic Books, New York.

[11] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. 2021. Toward Causal Representation Learning. *Proc. IEEE* 109, 5 (2021), 612–634. https://doi.org/10.1109/JPROC.2021.3058954

[12] Mukesh Singhal. 1993. A taxonomy of distributed mutual exclusion. *J. Parallel and Distrib. Comput.* 18, 1 (1993), 94–101.

[13] Matthew J. Vowels, Necati Cihan Camgöz, and Richard Bowden. 2021. D'ya like DAGs? A Survey on Structure Learning and Causal Discovery. *CoRR* abs/2103.02582 (2021). arXiv:2103.02582 https://arxiv.org/abs/2103.02582

[14] Qiaoling Ye, Arash A. Amini, and Qing Zhou. 2022. Distributed Learning of Generalized Linear Causal Networks. *CoRR* abs/2201.09194 (2022). arXiv:2201.09194 https://arxiv.org/abs/2201.09194