Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

# Exploiting the Structure of Distributed Constraint Optimization Problems

**Ferdinando Fioretto**
Dept. of Computer Science
New Mexico State University, NM 88001, USA
Dept. of Mathematics & Computer Science
University of Udine, UD 33100, IT
ffiorett@cs.nmsu.edu

## Introduction

In the proposed thesis, we study *Distributed Constraint Optimization Problems (DCOPs)*, which are problems where several agents coordinate with each other to optimize a global cost function. The use of DCOPs has gained momentum, due to their capability of addressing complex and naturally distributed problems. A majority of the work in DCOP addresses the resolution problem by detaching the model from the resolution process, where they assume that each agent controls exclusively one variable of the problem (Burke et al. 2006). This assumption often is not reflected in the model specifications, and may lead to inefficient communication requirements. Another limitation of current DCOP resolution methods is their inability to capitalize on the presence of structural information, which may allow incoherent/unnecessary data to reticulate among the agents (Yokoo 2001).

The purpose of the proposed dissertation is to study how to adapt and integrate insights gained from centralized solving techniques in order to enhance DCOP performance and scalability, enabling their use for the resolution of real-world complex problems. To do so, we hypothesize that one can exploit the DCOP structure in both problem modeling and problem resolution phases.

## Progress to Date

### Exploiting Structure from Problem Modeling

Modeling many real-world problems as DCOPs often require each agent to own a large number of variables. However, most DCOP resolution approaches assume that each agent can handle only a single variable of the problem. As such, researchers have proposed a number of pre-processing techniques to reformulate DCOPs with multi-variable agents into DCOPs with single-variable agents (Yokoo 2001; Burke et al. 2006). Unfortunately, these techniques do not scale with the size of the problem due to their inefficient communication requirements. Therefore, in our work submitted to *AAAI 2015*, we proposed a new pre-processing framework, that defines a clear separation between the distributed DCOP resolution and the centralized agent sub-problem resolution. This separation allows the use of efficient centralized techniques to solve agent sub-problems, that can be solved independently from that of other agents, and to use a global DCOP algorithm to coordinate the agents. The use of centralized solutions within each agent, allows us to speed up several DCOP algorithms by up to several orders of magnitude, while the knowledge acquired from the DCOP model allows us to reduce the algorithms communication requirements, when compared to existing pre-processing techniques—which ignore the structural information dictated by the model. These results validate our hypothesis that one can exploit the information encoded in the DCOP model through the use of centralized solutions.



### Exploiting Structure from Problem Solving

A number of multi-agent systems require agents to run on battery-powered devices and communicate over wireless networks. This imposes constraints on the number and size of individual messages exchanged among agents (Wahbi et al. 2014). *Inference-based* DCOP algorithms (Petcu et al. 2005), can be effective in solving such problems. They use techniques from dynamic programming to propagate aggregate information among agents, and while their requirements on the number of messages is linear in the number of agents, their messages have a size that is exponential in the size of the treewidth, which can be up to the number of agents $-1$. Several works from the DCOP community have recognized the use of hard constraints to reduce the size of the search space and/or reduce the message size. However, they are limited in exploiting relational information expressed in form of tables and/or associated to the form of domain consistency.

We have contributed to this body of research by introducing a type of consistency, called *Branch Consistency* (Fioretto et al. 2014b), that applies to paths in pseudotrees. The effect of enforcing Branch Consistency is the ability to actively exploit hard constraints (either explicitly provided in the problem specification or implicitly described in constraints cost tables) to prune the search space and to reduce the size of the messages exchanged among agents. We show that such form of consistency enforces a more effective pruning than those based on domain-consistency, leading enhanced efficiency and scalability. These results validate our hypothesis that centralized reasoning can be adapted to exploit the structure of DCOPs during problem solving.

## Exploiting Structure from Both Problem Modeling and Problem Solving

The separation between the DCOP resolution process and the centralized agent problem, which was gained by exploiting the problem model, enabled agents to solve their local problem trough a variety of techniques. Motivated by the high complexity of the agent local problem, we proposed the use of hierarchical parallel models, where each agent can (a) solve its local problem independently from those of other agents, and (b) parallelize the computations within its own local problem. Such model builds on top of algorithm-specific characteristics, and may substantially reduces the runtime for several DCOP algorithms classes. For instance, in (Fioretto et al. 2014a), we suggest to solve independent local problems, in parallel, harnessing the multitude of computational units offered by GPGPUs, which led to significant improvements in the runtime of the algorithm resolution.

# Proposed Plan for the Future

## Efficient Local Search Strategies for DCOPs

In the work conducted so far we adapted centralized reasoning techniques to complete DCOP algorithms. Nevertheless, solving DCOPs optimally is NP-hard, therefore for large problems, incomplete DCOP algorithms are desirable. Current incomplete search techniques can either (a) find local minima without quality guarantees (Maheswaran et al. 2004), (b) provide quality assessment but inefficiently (such us those in the class of k-optimality (Pearce et al. 2007)), (c) cannot exploit problem structural information, such as hard constraints (Nguyen et al. 2013).

Therefore, capitalizing on strategies from the centralized constraint reasoning community, we propose to adapt the *Large Neighboring Search strategy (LNS)* (Van Hentenryck et al. 2009) to the DCOP resolution process. This technique allows to rapidly find solutions by fixing the variable assignments of a set of agents while optimizing over the others. We believe that LNS is a desirable candidate to DCOP local search because (a) emulating the centralized results, it can quickly find local minima, (b) it inherently uses insights from the CP techniques to take advantage on the presence of hard constraints, and to refine the solution quality—by constraining the solution bound during the resolution process—and (c) it is amenable to parallelization (e.g., if groups of agents can explore several neighbors at a time (Campeotto et al. 2014)). We plan in studying the use of machine learning techniques to select which set of variables (agents) to unlock during the DCOP solving phase, as well as ensuring quality guarantees on the solution found.

## Distributed Simulator, Modeling Language and Application to the Smart Grid Problems

Despite the wide applicability of the DCOP model, there is no general language being used to formally specify a DCOP. By and large, most stand-alone algorithms specify DCOPs in an ad-hoc manner. Moreover, current DCOP simulators model agents as entities runnig on the same single machine. Therefore we propose a new agent-based modeling language, which extends the MiniZinc language (Nethercote et al. 2007). Such a DCOP language is (a) more expressive than other adopted formalisms (such as XML-based DCOP descriptions) and (b) it allows the expression of constraints succinctly, in the form of rules, using a well adopted semantics from the constraint reasoning community and allows a fine integration with agents' centralized solvers. Our preliminary results show that such a representation may significantly affect performance, due to the stronger inference that may be derived from explicit constraint representation.

We are also implementing a DCOP solver that uses agent distributed over different machines, and can communicate using several network standard communication protocols (TCP-IP, wireless, etc.). We believe that this is a valuable contribution, as current DCOP simulators suffer from strong communication assumptions (e.g., they assume the same cost for all communications, and direct communications, with no routing), which may not fully reflect the DCOP algorithm behavior on real scenarios (Wahbi et al. 2014).

Leastly, we plan to apply the techniques produced in the proposed thesis on smart grid domains, where we model a network of building (e.g., homes, offices, power plants), each of which is managed by an agent, with its own power demands, including generators (e.g., solar panels, gas-based generators) and accumulators (e.g., batteries, electric vehicles). One of the problem that we plan to study is that of finding (sub-)optimal schedule for the local energy dispatch.

# References

Burke, D., and Brown, K. 2006. Efficiently handling complex local problems in distributed constraint optimisation. In *ECAI*, 701–702.

Campeotto, F.; Dovier, A.; Fioretto, F.; and Pontelli, E. 2014. A GPU implementation of large neighborhood search for solving constraint optimization problems. In *ECAI*, 189–194.

Fioretto, F.; Campeotto, F.; Fioretto, L. D. R.; Yeoh, W.; and Pontelli, E. 2014a. GD-Gibbs: A GPU-based sampling algorithm for solving distributed constraint optimization problems In *AAMAS*.

Fioretto, F.; Le, T.; Yeoh, W.; Pontelli, E.; and Son, T. 2014b. Improving DPOP with branch consistency for solving distributed constraint optimization problems. In *CP*.

Van Hentenryck, P., and Michel, L. 2009. *Constraint-based local search*. The MIT Press.

Maheswaran, R.; Pearce, J.; and Tambe, M. 2004. Distributed algorithms for DCOP: A graphical game-based approach. In *PDCS*, 432–439.

Nethercote, N.; Stuckey, P. J.; Becket, R.; Brand, S.; Duck, G. J.; and Tack, G. 2007. Minizinc: Towards a standard CP modelling language. In *CP*. 529–543.

Nguyen, D. T.; Yeoh, W.; and Lau, H. C. 2013. Distributed Gibbs: A memory-bounded sampling-based DCOP algorithm. In *AAMAS*, 167–174.

Pearce, J., and Tambe, M. 2007. Quality guarantees on k-optimal solutions for distributed constraint optimization problems. In *IJCAI*, 1446–1451.

Petcu, A., and Faltings, B. 2005. A scalable method for multiagent constraint optimization. In *IJCAI*, 1413–1420.

Wahbi, M., and Brown, K. N. 2014. The impact of wireless communication on distributed constraint satisfaction. In *CP*, 738–754.

Yokoo, M., ed. 2001. *Distributed Constraint Satisfaction: Foundation of Cooperation in Multi-agent Systems*. Springer.