Distributed Constraint Optimization Problems related with Soft Arc Consistency

Patricia Gutierrez and **Pedro Meseguer**
IIIA - CSIC, Universitat Autònoma de Barcelona, Bellaterra, Spain
{patricia|pedro}@iiia.csic.es

Abstract

Distributed Constraint Optimization Problems (DCOPs) can be optimally solved by distributed search algorithms, such as ADOPT and BnB-ADOPT. In centralized solving, maintaining soft arc consistency during search has proved to be beneficial for performance. In this thesis we aim to explore the maintenance of different levels of soft arc consistency in distributed search when solving DCOPs.

1 State of the Art

DCOPs. Distributed Constraint Optimization Problems (DCOPs) [Modi *et al.*, 2005] can be used for modeling many multi-agent coordination problems, such as distributed meeting scheduling [Maheswaran *et al.*, 2004], sensor networks [Jain *et al.*, 2009], traffic control [Junges and Bazzan, 2008], and others. DCOPs involve a finite number of agents, variables and cost functions. The cost of an assignment of a subset of variables is the evaluation of all cost functions on that assignment. The goal is to find a complete assignment with minimum cost. This is achieved among several agents handling the variables and exchanging information about their cost evaluation until an optimal solution is found.

Distributed Algorithms. Researchers have proposed several distributed search algorithms to optimally solve DCOPs. The first proposed complete algorithm was ADOPT [Modi *et al.*, 2005], which performs distributed search using a best-first strategy. Later on, the closely related BnB-ADOPT [Yeoh *et al.*, 2010] was presented. This algorithm changes the nature of the search from ADOPT best-first search to a depth-first branch-and-bound search strategy, obtaining a better performance. Both algorithms are complete, compute the optimum cost and terminate.

DCOPs are NP-hard, so an exponential time is needed in the worst case to find the optimum. This severely limits the application of existing solving approaches.

Soft Arc Consistency. In the centralized case, several techniques have been developed to speed up the solving of constraint optimization problems. In particular, search can be improved by enforcing soft arc consistency, which identifies inconsistent values that can be removed from the problem.

The simplest form of soft arc consistency is defined as follows: consider $\top$ an unacceptable cost (1) a value v of variable x_i is node consistent (NC) if the unary cost function $C_i(v) < \top$, variable x_i is NC if all its values are NC, the problem is NC if every variable is NC; (2) a value v of variable x_i is arc consistent (AC) with respect to binary cost function C_{ij} if it is NC and there exists another value $w \in D_j$ such that $C_{ij}(v, w) = 0$. Variable x_i is AC if all its values are AC, and a problem is AC if all variables are AC. Several stronguer soft arc consistency levels have been proposed [Larrosa and Schiex, 2003; de Givry *et al.*, 2005; Cooper *et al.*, 2008]. By enforcing them it is possible to detect suboptimal values that can be removed from the problem. In practical terms, the effect is that the search tree is reduced and there are fewer nodes to explore, but on the other hand more computational work must be done per node. Globally, the overall effect is very benefitial.

2 Thesis Goal

The thesis goal is to include in distributed search algorithms for DCOPs solving some techniques to enforce soft arc consistency during search. Such as it happens in the centralized case, we expect that this combination would cause performance improvements.

In the distributed context, maintaining arc consistency when solving satisfaction problems is beneficial for performance [Brito and Meseguer, 2008]. Taking inspiration from this fact, we want to explore soft arc consistency maintenance when solving DCOPs. Soft arc consistencies are conceptually equal in the centralized and distributed cases. However, maintaining soft arc consistencies during distributed search requires different techniques. While in the centralize case all problem elements are available to the single agent performing the search, in the distributed case agents only know some part of the problem and must exchange information in order to achieve the desired consistency level. In this process, operations that modify the problem structures should be done in such a way that the partial representation of the whole problem remains coherent on every agent. Maintaining soft arc consistencies during search does not compromise the optimality and termination of ADOPT and BnB-ADOPT.

We plan to measure the efficiency of the proposed algorithms with respect to existing ones in terms of (synchronous) cycles [Modi *et al.*, 2005], non-concurrent constraint checks (NCCCs) [Meisels *et al.*, 2002], and network load.

3 Research Done

3.1 Contributions to Distributed Search

We have experimented with existing complete DCOP algorithms, namely ADOPT and BnB-ADOPT. As result of this work, we have improved them to a large extent, as explained in the following. On their execution, ADOPT and BnB-ADOPT exchange a large number of messages, which is a major drawback for their practical use. Aiming at increasing their efficiency, we show that some of these messages are redundant and can be removed without compromising optimality and termination properties. Removing most of those redundant messages we obtain ADOPT$^+$ [Gutierrez and Meseguer, 2010c] and BnB-ADOPT$^+$ [Gutierrez and Meseguer, 2010b].

When tested on commonly used benchmarks, these algorithms obtain a large reductions in the number of messages, a slight reduction in NCCCs, and the number of cycles remains constant. BnB-ADOPT$^+$ was able to process only half of messages in the worst case and reach the optimal solution in almost the same number of cycles [Gutierrez and Meseguer, 2010b], while ADOPT$^+$ divided the number of messages by a factor from 1.1. to almost 3 on the benchmark tested [Gutierrez and Meseguer, 2010c].

3.2 BnB-ADOPT and Soft Arc Consistency

Specifically, we have experimented with BnB-ADOPT$^+$, on top of which we maintain AC* and FDAC* levels of soft arc consistency. During BnB-ADOPT$^+$ execution, we can assure in some cases that the value of a variable is not in the optimal solution. Then, this value can be deleted unconditionally. We propagate these unconditionally deleted values in such a way that they can be known by other neighboring agents. When deletions are propagated the consistency properties are reinforced on neighbor agents, which may generate some deletions that will also be propagated. The global effect is that we search in a smaller space, causing performance improvements.

We presented the new algorithms BnB-ADOPT$^+$- AC* and BnB-ADOPT$^+$-FDAC* [Gutierrez and Meseguer, 2010a], which combine distributed search with the levels of consistencies AC* and FDAC*. Maintaining AC* level (BnB-ADOPT$^+$-AC*) we observe a clear decrement in the number of messages and also in the number of cycles. Maintaining FDAC* level (BnB-ADOPT$^+$-FDAC*) enhances this reduction. In the worst case, when maintaining FDAC* our approach divides the number of required messages by a factor of 3, substantially decreasing the number of cycles as well. Although agents need to perform more local computation to maintain consistency, the number of NCCCs shows important reductions. This is the combination of two opposite trends: agents are doing more work enforcing the desired consistency level and processing new messages introduced to propagate deletions and exchange new information, but this is largely compensated by a decrement in the number of messages used to solve the problem. This combination turns out to be very beneficial, saving computational effort in all cases tested. In some cases, reduction reaches up to one order of magnitude [Gutierrez and Meseguer, 2010a].

4 Future Work

With regard to distributed search, we are working on a hybrid version of ADOPT and BnB-ADOPT, which are very similar except for their search strategies. We intent to present a new algorithm that generalizes them obtaining better results.

Also, we aim to extend our work to higher levels of soft arc consistency. To maintain these levels agents need to have a wider knowledge about the global problem. This may compromise privacy, which is an issue to resolve. Furthermore, we want to explore soft arc consistency maintenance on problems with non binary and global constrains.

Finally, we want to explore the interest of propagating conditional deletions. In a partial assigned solution, values may be found inconsistent in the remaining unassigned variables. These values can be removed but they need to be restored when the partial solution changes. This process reduces the search space but requires more computational effort.

References

[Brito and Meseguer, 2008] I. Brito and P. Meseguer. Connecting ABT with arc consistency. *CP-08*, 2008.

[Cooper *et al.*, 2008] M. Cooper, S. de Givry, M. Sanchez, T. Schiex, and M. Zytnicki. Virtual arc consistency for weighted csp. *AAAI*, 2008.

[de Givry *et al.*, 2005] S. de Givry, F. Heras, J. Larrosa, and M. Zytnicki. Existential arc consistency: getting closer to full arc consistency in weighted CSPs. *IJCAI*, 2005.

[Gutierrez and Meseguer, 2010a] P. Gutierrez and P. Meseguer. BnB-ADOPT$^+$ with several soft arc consistency levels. In *ECAI*, pages 67–72, 2010.

[Gutierrez and Meseguer, 2010b] P. Gutierrez and P. Meseguer. Saving redundant messages in ADOPT-based algorithms. In *DCR*, pages 53–64, 2010.

[Gutierrez and Meseguer, 2010c] P. Gutierrez and P. Meseguer. Saving redundant messages in BnB-ADOPT. In *AAAI*, pages 1259–1260, 2010.

[Jain *et al.*, 2009] M. Jain, M. Taylor, M. Tambe, and M. Yokoo. DCOPs meet the realworld: Exploring unknown reward matrices with applications to mobile sensor networks. In *IJCAI*, 2009.

[Junges and Bazzan, 2008] R. Junges and A. L. C. Bazzan. Evaluating the performance of DCOP algorithms in a real world dynamic problem. In *AAMAS*, pages 599–606, 2008.

[Larrosa and Schiex, 2003] J. Larrosa and T. Schiex. In the quest of the best form of local consistency for weighted CSP. *IJCAI*, 2003.

[Maheswaran *et al.*, 2004] R. Maheswaran, M. Tambe, E. Bowring, J. Pearce, and P. Varakantham. Taking DCOP to the real world: Efficient complete solutions for distributed event scheduling. In *AAMAS*, pages 310–317, 2004.

[Meisels *et al.*, 2002] A. Meisels, E. Kaplansky, I. Razgon, and R. Zivan. Comparing performance of distributed constraints processing algorithms. In *DCR*, pages 86–93, 2002.

[Modi *et al.*, 2005] P. Modi, W. Shen, M. Tambe, and M. Yokoo. ADOPT: Asynchronous distributed constraint optimization with quality guarantees. *Artificial Intelligence*, 161:149–180, 2005.

[Yeoh *et al.*, 2010] W. Yeoh, A. Felner, and S. Koenig. BnB-ADOPT: An asynchronous branch-and-bound DCOP algorithm. *Journal of Artificial Intelligence Research*, 38:85–133, 2010.