Towards Scaling Up Search Algorithms for Solving Distributed Constraint Optimization Problems

(Extended Abstract)

William Yeoh
Computer Science
University of Southern California
Los Angeles, CA 90089
wyeoh@usc.edu

My thesis will demonstrate that distributed constraint optimization (DCOP) search algorithms can be scaled up (= applied to larger problems) by applying the knowledge gained from centralized search algorithms.

1. INTRODUCTION

Agent-coordination problems can be modeled as distributed constraint optimization (DCOP) problems [10, 12, 20]. A DCOP problem consists of a set of agents, each responsible for taking on (= assigning itself) a value from their finite domains. The agents coordinate their value assignments subject to a set of constraints. Two agents are said to be constrained if they share a constraint. Each constraint has an associated cost which depends on the values taken on by the constrained agents. A complete solution is an assignment of values to all agents. The cost of a complete solution is the sum of the constraint costs of all constraints resulting from the given value assignments. Solving a DCOP problem optimally means to find a complete solution such that the sum of all constraint costs is minimized. Finding such a cost-minimal solution is NP-hard [10].

This model is rapidly becoming popular for formulating and solving agent-coordination problems [6, 7, 5]. As a result, DCOP algorithms that use search techniques such as ADOPT (Asynchronous Distributed Constraint Optimization) [10] have been developed.

2. CONTRIBUTIONS

Since solving DCOP problems is NP-hard, my research concentrates on finding intelligent ways to scale up DCOP search algorithms such that they can be used in larger applications. DCOP search algorithms can be viewed as distributed versions of centralized search algorithms with assumptions that are specific to DCOP problems. For example, the solution space (= space of all possible solutions) of DCOP problems is bounded by the number of agents in the problem. Therefore, some of the knowledge gained by researchers investigating centralized search algorithms might apply to DCOP search algorithms as well.

To avoid reinventing the wheel, my thesis will center around scaling up DCOP search algorithms by applying the knowledge gained from centralized search algorithms. I made a design choice to reuse the framework of ADOPT, which is one of the pioneering DCOP search algorithms, as the starting platform for the work in my dissertation. The motivation for this decision is that ADOPT has been extended very widely [9, 1, 11, 2, 14]. In particular, my contributions lie along two axes: (1) Memory availability of agents and (2) Requirement of solution optimality.

For problems where the agents have a *minimal amount of memory* and the *cost-minimal solution is required*, I introduced a new DCOP search algorithm called Branch-and-Bound ADOPT (BnB-ADOPT) in [16], that speeds up ADOPT by one order of magnitude for sufficiently large DCOP problems. BnB-ADOPT is a memory-bounded asynchronous DCOP search algorithm that uses the message passing and communication framework of ADOPT but changes the search strategy of ADOPT from best-first search to depth-first branch-and-bound search. Experimental results show that BnB-ADOPT is faster than ADOPT for sufficiently large DCOP problems because the available heuristics for these problems are often uninformed. The key contribution of this work is the identification and verification of depth-first branch-and-bound search instead of best-first search as the preferred search strategy for DCOP problems, which is consistent with findings for centralized search algorithms [19].

For problems where the agents have *more than the minimal amount of memory*, I introduced new caching schemes called MaxPriority, MaxEffort and MaxUtility in [18], that are tailored to DCOP search algorithms including ADOPT and BnB-ADOPT, and thus speed up both algorithms further. These caching schemes make use of the lower and upper bounds maintained by agents in ADOPT and BnB-ADOPT, as well as the knowledge of which search strategy is employed by ADOPT and BnB-ADOPT. Our experimental results show that the MaxEffort and MaxUtility schemes perform better than the other schemes for ADOPT, and the MaxPriority scheme is generally no worse than the other schemes for BnB-ADOPT. The speedup from caching for ADOPT is significantly larger than that for BnB-ADOPT since ADOPT needs to re-acquire information that was purged due to memory limitations. The key contribution of this work is the investigation of the different caching schemes and the identification of preferred schemes for the different algorithms. In general, these schemes should apply to other DCOP search algorithms as well since they also maintain lower and upper bounds on the solution quality.

For problems where the *cost-minimal solution is not required*, I introduced two approximation mechanisms for ADOPT and BnB-ADOPT that trade off solution quality for faster computation time in [17]. The approximation mechanisms, namely the Relative Error Mechanism and the Weighted Heuristics Mechanism, provide relative error bounds (i.e. a percentage off the minimal cost). These mechanisms complement existing mechanisms that only allow absolute error bounds (i.e. an absolute off the minimal cost). Additionally, experimental results show that the Weighted Heuristics Mechanism dominates the other mechanism. The key contribution of this work is the introduction of the Weighted Heuristics Mechanism, which should also apply to other DCOP search algorithms that use heuristics to guide their search. This mechanism was motivated by Weighted A* [13], an approximation algorithm based on the centralized search algorithm A* [3].

I conducted my experiments in three problem types that are commonly used to evaluate DCOP algorithms. The three problem types are the problem of coloring graphs, the problem of allocating targets to sensor networks and the problem of scheduling meetings. I measured the runtime of the algorithms using two commonly used metrics, namely time slices called cycles [10] and non-concurrent constraint checks [8].

3. FUTURE WORK

So far, my contributions only apply to static problems (i.e. problems that do not change over time). To complete my thesis, I plan to extend my work to dynamic problems by developing new DCOP search algorithms that operate efficiently in these environments. Specifically, I have two objectives in mind: (1) algorithms that find cost-minimal solutions of dynamic DCOP problems; and (2) algorithms that find error-bounded solutions of DCOP problems that are most similar to the solutions of the problems before they changed (due to changes in the environment). I plan to measure the similarity of two solutions by the number of agents that take on the same value in both solutions.

To achieve the first objective, I plan to develop DCOP search algorithms that perform a new search every time the DCOP problem changes but reuse information from the previous searches. Therefore, these algorithms should be faster than those that run each search from scratch. This plan is motivated by incremental centralized search algorithms [15].

To achieve the second objective, I plan to develop a DCOP search algorithm that employs limited discrepancy search [4]. Limited discrepancy search searches for solutions in the order of increasing numbers of discrepancies, i.e. numbers of agents that take on values different from their previous values, and is thus ideally suited for finding the most similar error-bounded solution.

Acknowledgments

This research has been partly supported by an NSF award to Sven Koenig under contract IIS-0413196. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the sponsoring organizations, agencies, companies or the U.S. government.

4. REFERENCES

[1] E. Bowring, M. Tambe, and M. Yokoo. Multiply-constrained distributed constraint optimization. In *Proceedings of AAMAS*, pages 1413–1420, 2006.

[2] J. Davin and P. Modi. Hierarchical variable ordering for multiagent agreement problems. In *Proceedings of AAMAS*, pages 1433–1435, 2006.

[3] P. Hart, N. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, SSC4(2):100–107, 1968.

[4] W. Harvey and M. Ginsberg. Limited discrepancy search. In *Proceedings of IJCAI*, pages 607–615, 1995.

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

[6] V. Lesser, C. Ortiz, and M. Tambe, editors. *Distributed Sensor Networks: A Multiagent Perspective*. Kluwer, 2003.

[7] 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 *Proceedings of AAMAS*, pages 310–317, 2004.

[8] A. Meisels, E. Kaplansky, I. Razgon, and R. Zivan. Comparing performance of distributed constraints processing algorithms. In *Proceedings of the Distributed Constraint Reasoning Workshop*, pages 86–93, 2002.

[9] P. Modi and S. Ali. Distributed constraint reasoning under unreliable communication. In W. Zhang and V. Sorge, editors, *Frontiers in Artificial Intelligence and Applications*, volume 112, pages 141–150. IOS Press, 2004.

[10] P. Modi, W.-M. Shen, M. Tambe, and M. Yokoo. ADOPT: Asynchronous distributed constraint optimization with quality guarantees. *Artificial Intelligence*, 161(1-2):149–180, 2005.

[11] F. Pecora, P. Modi, and P. Scerri. Reasoning about and dynamically posting n-ary constraints in ADOPT. In *Proceedings of the Distributed Constraint Reasoning Workshop*, pages 57–71, 2006.

[12] A. Petcu and B. Faltings. A scalable method for multiagent constraint optimization. In *Proceedings of IJCAI*, pages 1413–1420, 2005.

[13] I. Pohl. First results on the effect of error in heuristic search. *Machine Intelligence*, 5:219–236, 1970.

[14] M. Silaghi and M. Yokoo. Dynamic DFS tree in ADOPT-ing. In *Proceedings of AAAI*, pages 763–769, 2007.

[15] X. Sun and S. Koenig. The fringe-saving A* search algorithm – a feasibility study. In *Proceedings of IJCAI*, pages 2391–2397, 2007.

[16] W. Yeoh, A. Felner, and S. Koenig. BnB-ADOPT: An asynchronous branch-and-bound DCOP algorithm. In *Proceedings of AAMAS*, pages 591–598, 2008.

[17] W. Yeoh, S. Koenig, and X. Sun. Trading off solution cost for smaller runtime in DCOP search algorithms (short paper). In *Proceedings of AAMAS*, pages 1445–1448, 2008.

[18] W. Yeoh, P. Varakantham, and S. Koenig. Caching schemes for DCOP search algorithms. In *Proceedings of AAMAS*, 2009.

[19] W. Zhang and R. Korf. Performance of linear-space search algorithms. *Artificial Intelligence*, 79(2):241–292, 1995.

[20] W. Zhang, G. Wang, Z. Xing, and L. Wittenberg. Distributed stochastic search and distributed breakout: Properties, comparison and applications to constraint optimization problems in sensor networks. *Artificial Intelligence*, 161(1-2):55–87, 2005.