A Hybrid Approach to Solving Coarse-grained DisCSPs (Extended Abstract)

David Lee
School of Computing, RGU
Aberdeen, Scotland, UK
dl@comp.rgu.ac.uk

Inés Arana
School of Computing, RGU
Aberdeen, Scotland, UK
ia@comp.rgu.ac.uk

Hatem Ahriz
School of Computing, RGU
Aberdeen, Scotland, UK
ha@comp.rgu.ac.uk

Kit-Ying Hui
School of Computing, RGU
Aberdeen, Scotland, UK
khui@comp.rgu.ac.uk

ABSTRACT

A coarse-grained Distributed Constraint Satisfaction Problem (DisCSP) consists of several loosely connected constraint satisfaction subproblems, each assigned to an individual agent. We present Multi-Hyb, a two-phase concurrent hybrid approach for solving DisCSPs. In the first phase, each agent's subproblem is solved using systematic search which generates the key partial solutions to the global problem. Concurrently, a penalty-based local search algorithm attempts to find a global solution from these partial solutions. If phase 1 fails to find a solution, a phase 2 systematic search algorithm solves the problem using the knowledge gained from phase 1. We show that our approach is highly competitive in comparison with other coarse-grained DisCSP algorithms.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence; I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

General Terms

Algorithms

Keywords

Artificial Intelligence, Constraint Satisfaction, Agent Cooperation, Distributed Problem Solving

1. INTRODUCTION AND BACKGROUND

A Constraint Satisfaction Problem (CSP) is represented by a set of variables, a corresponding set of domains (one per variable) and a set of constraints restricting values that variables can take simultaneously. A solution is an assignment of a value to each variable that satisfies all constraints. Resolution of CSPs is usually performed by: (i) systematic search algorithms, which ensure completeness, but are slow or; (ii) local search algorithms which are incomplete, but faster than systematic algorithms for larger problems [5].

Coarse-grained Distributed Constraint Satisfaction Problems (DisCSPs) [7] are CSPs which are naturally distributed over a number of agents, i.e. each agent solves a related subproblem which can be represented by a CSP. Variables in a subproblem are related by intra-agent constraints. Subproblems are linked together by a set of constraints which relate variables in two or more subproblems (inter-agent constraints). We concentrate on naturally distributed problems, i.e. those for which an imbalance exists between inter-agent and intra-agent constraints, with a higher number of the latter.

Existing systematic search techniques for solving coarse-grained DisCSPs include Multi-AWCS [6] and Multi-ABT [3]. In both algorithms, a local solver ensures satisfaction of intra-agent constraints, whilst a global solver ensures satisfaction of inter-agent constraints. Whilst Multi-ABT and Multi-AWCS are complete, they may take exponential time and Multi-AWCS may require an exponential number of nogoods to be stored.

Local search techniques for DisCSPs differ in the strategy used when (potential) quasi-local-optima are discovered. Thus, DisBO-wd [2] attaches weights to constraints with a decay mechanism whilst Multi-DisPeL [2] assigns penalties to values.

Hybrid approaches combining systematic search and local search can potentially benefit from fast(er) resolution while ensuring completeness. Although there have been several hybrid approaches for solving DisCSPs, e.g. [4], there are no hybrid approaches specifically for solving coarse-grained DisCSPs.

2. THE MULTI-HYB ALGORITHM

We present Multi-Hyb, a novel two-phase distributed hybrid algorithm for solving naturally distributed coarse-grained DisCSPs, i.e. with high intra-agent to inter-agent constraint ratio.

Phase 1: Each agent finds all 'relevant' solutions to its local subproblem using the Synchronous Exhaustive Backjumping (SEBJ - see below) algorithm. Concurrently, a DisPeL-1C search (see below) takes place, attempting to solve the global problem. Phase 1 finishes when either all agents have generated all possible 'relevant' solutions to their

Cite as: A Hybrid Approach to Solving Coarse-grained DisCSPs, (Extended Abstract), David Lee, Inés Arana, Hatem Ahriz, Kit-Ying Hui, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. 1235–1236

local subproblem using SEBJ or DisPeL-1C has found a solution to the global problem. If a global solution has not been found, Multi-Hyb moves to the second phase and stops the DisPeL-1C search.

SEBJ is a new algorithm to find all solutions to a subproblem which have external relevance, i.e. the set of all solutions which differ on at least one value for an external variable (a variable linked to another subproblem). This generation of only externally relevant solutions to subproblems significantly reduces computational cost. The SEBJ algorithm is complete (guaranteed to discover all solutions to local variables with external relevance).

DisPeL-1C is inspired by Stoch-DisPeL [1] but continuously imposes penalties to inconsistent values without waiting for quasi-local-minima to be detected. These penalties are accumulated, representing the level of difficulty that DisPeL-1C had when trying to find a suitable value for that variable. In addition, DisPeL-1C: (i) has complex variables, aggregating all variables for one subproblem in one agent; (ii) adds values for variables with inter-agent constraints dynamically to their domain over time (as SEBJ finds them) (iii) only considers the inter-agent constraints since intra-agent constraints have been handled by SEBJ. If DisPeL-1C finds a global solution, Multi-Hyb terminates.

Phase 2: The Phase 2 Algorithm (inspired by PenDHyb [4]) runs the systematic search algorithm SynCBJ [8] and exploits the knowledge learned from phase 1. Like DisPeL-1C, the Phase 2 algorithm uses complex variables and only considers inter-agent constraints. The solutions generated by SEBJ form the domain of each agent and agents are ordered according to their penalty count from DisPeL-1C. Consequently, agents with difficult subproblems are ordered first.

3. EXPERIMENTAL EVALUATION

We compared Multi-Hyb (HYB) with Multi-DisPeL (DI) and DisBO-wd (BO) [2], since these algorithms were proved to be superior to other coarse-grained DisCSP algorithms, measuring the number of messages sent and the number of concurrent constraint checks performed. Median results for solvable randomly generated problems are shown in Table 1. We used 5 agents, domain size of 8, constraint density of 0.2 and constraint tightness of 0.35. In the table, (1) indicates a ratio of 80% intra-agent constraints to 20% inter-agent constraints; (2) indicates a ratio of 70% intra-agent constraints to 30% inter-agent constraints.

Table 1: Results for solvable random problems

	N. of messages			N. of CCCs		
n	HYB	DI	BO	HYB	DI	BO
70 (1)	**159**	208	435	**151678**	745608	252678
70 (2)	**112**	194	420	**291421**	673099	244962
100 (1)	**56**	**56**	235	**107836**	690977	339423
100 (2)	78	**60**	225	**132031**	690455	324668
150 (1)	**20**	28	215	**100020**	1362161	728427
150 (2)	**30**	32	195	**120105**	1281866	682116
175 (1)	**20**	24	210	**98875**	1926771	976712
175 (2)	**20**	24	190	**110325**	1831216	908710

Multi-Hyb outperforms both Multi-DisPeL and DisBO-wd on number of messages for problems with less than 150 variables. For problems with more than 150 variables, Multi-Hyb performs similarly to Multi-DisPeL with very few messages. However, for concurrent constraint checks, Multi-Hyb outperforms both algorithms by several orders of magnitude.

We conducted an extensive empirical evaluation on 3 problem classes: distributed randomly generated problems (as shown above), 3-colour distributed graph colouring problems and distributed scheduling problems. For each problem class, we evaluated many different problem parameters and found similar trends to those shown above namely that Multi-Hyb outperformed Multi-DisPeL and DisBO-wd by several orders of magnitude.

Experiments on both solvable and unsolvable problems comparing Multi-Hyb and Multi-AWCS were also carried out showing a competitive advantage for Multi-Hyb. Note that Multi-DisPeL and DisBO-wd are incomplete and, therefore, they cannot be compared on unsolvable problems.

4. CONCLUSIONS

We have presented Multi-Hyb, a concurrent two-phase distributed complete algorithm for solving coarse-grained DisCSPs. Phase 1 runs SEBJ to concurrently solve each agent's local subproblem whilst also running DisPeL-1C to solve the inter-agent constraints. A Phase 2 algorithm which exploits the knowledge learnt in phase 1 guarantees completeness if DisPeL-1C does not solve the problem. Multi-Hyb outperforms Multi-DisPeL, DisBO-wd and Multi-AWCS.

In summary, Multi-Hyb is a successful concurrent algorithm for solving coarse-grained DisCSPs which uses both local search and systematic search to guarantee completeness whilst finding solutions faster than competitive approaches.

5. REFERENCES

[1] M. Basharu, I. Arana, and H. Ahriz. StochDisPeL Exploiting Randomisation in DisPeL. In *Proceedings of 7th International Workshop on Distributed Constraint Reasoning*, pages 117–132, Hakodate, Japan, May 2006.

[2] M. Basharu, I. Arana, and H. Ahriz. Solving Coarse-grained DisCSPs with Multi-DisPeL and DisBO-wd. In *Proceedings of 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, pages 335–341. IEEE Press, 2007.

[3] K. Hirayama, M. Yokoo, and K. Sycara. An Easy-Hard-Easy Cost Profile in Distributed Constraint Satisfaction. *Transactions of Information Processing Society of Japan*, 45:2217–2225, 2004.

[4] D. Lee, I. Arana, H. Ahriz, and K.-Y. Hui. A Hybrid Approach to Distributed Constraint Satisfaction. In D. Dochev, M. Pistore, and P. Traverso, editors, *Artificial Intelligence: Methodology, Systems,and Applications. 13th International Conference, AIMSA 2008 Varna, Bulgaria, September 4-6, 2008 Proceedings*, pages 375–379, Varna, Bulgaria, September 2008.

[5] F. Rossi, P. van Beek, and T. Walsh. *Handbook of Constraint Programming*. Elsevier, 2006.

[6] M. Yokoo and K. Hirayama. Distributed Constraint Satisfaction Algorithm for Complex Local Problems. In *ICMAS*, pages 372–379, 1998.

[7] M. Yokoo and K. Hirayama. Algorithms for Distributed Constraint Satisfaction: A Review. *Autonomous Agents and Multi-Agent Systems*, 3(2):185–207, 2000.

[8] R. Zivan and A. Meisels. Synchronous vs Asynchronous search on DisCSPs. In *Proceedings of the First European Workshop on Multi-Agent Systems (EUMA)*, Oxford, December 2003.