

Decoupling the Multiagent Disjunctive Temporal Problem (Extended Abstract)

James C. Boerkoel Jr.
 Computer Science and Artificial Intelligence Lab
 Massachusetts Institute of Technology
 Cambridge, MA
 boerkoel@csail.mit.edu

Edmund H. Durfee
 Computer Science and Engineering
 University of Michigan
 Ann Arbor, MI
 durfee@umich.edu

ABSTRACT

In multiagent scheduling, each agent has to schedule its activities to respect its local (internal) temporal constraints, and also to satisfy external constraints between its activities and activities of other agents. A scheduling problem is *decoupled* if each agent can independently (and thus privately, autonomously, etc.) form a solution to its local problem such that agents' combined solutions are guaranteed to satisfy all external constraints. We expand previous work that decouples multiagent scheduling problems containing strictly conjunctive temporal constraints to more general problems containing disjunctive constraints. While this raises a host of challenging issues, agents can leverage shared information as early and as often as possible to quickly adopt additional temporal constraints within their local problems that sacrifice some local scheduling flexibility in favor of decoupled, independent, and rapid local scheduling.

Categories and Subject Descriptors

I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence

General Terms

Algorithms, Experimentation, Theory

Keywords

Disjunctive Multiagent Scheduling, Temporal Decoupling

1. MOTIVATION

Many scheduling problems can be represented using temporal constraint networks, where events are represented as variables whose domains are the possible execution times, and where constraints restrict the timings between events in terms of bounds on the differences between variables' values. Figure 1(a) represents one such scheduling problem involving four tasks, with constraints between the variables representing the start and end times of each task. For instance the edge from $T3_{ST}^B$ to $T3_{ET}^B$ represents the constraint $T3_{ET}^B - T3_{ST}^B \in [50, 80]$, that is, the duration of task T3 is between 50 and 80 minutes. The *Disjunctive Temporal*

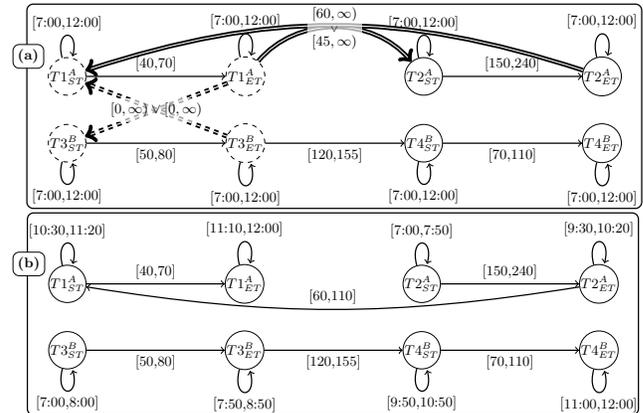


Figure 1: An example MaDTP (a) and corresponding temporal decoupling (b)

Problem (DTP) [4] is a general version of such problems, where constraints represent a choice among many constituent temporal difference constraints, each of which has its own bounds expressed over its own pair of timepoints. The disjunctive constraints in Figure 1(a) are represented with double lines, where all edges belonging to a single disjunctive constraint intersect (e.g., T1 must follow T2 by 60 minutes or precede T2 by 45). A *consistent* DTP is one that has a *solution*—a scheduling of specific times to each variable that respects all constraints. There are flexibility benefits to representing DTP *solution spaces*—sets of solutions naturally captured within the flexible ranges of times between temporal bounds—rather than a single, possibly brittle solution. The DTP is known to be an NP-hard problem, where for general DTPs with $|C|$ disjunctive temporal constraints each with k possibilities, each of the $\mathcal{O}(k^{|C|})$ possible networks of constraints must be explored in the worst case [4]. A *multiagent* DTP (MaDTP) [2] is one whose variables, and constraints among them, are partitioned among n agents. For example, the top and bottom rows in Figure 1(a) represent tasks belonging to two different agents, A and B respectively. The DTPs of different agents are constrained through *external constraints*, represented using dashed lines.

While external constraints capture key relationships between different agents' activities, they also introduce *coupling* between agents' local problems. For example, to modify the completion time for a deliverable (e.g., a data analysis or query answer), an agent needs to check that others can adjust their schedules to accommodate the change, which can trig-

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013)*, Ito, Jonker, Gini, and Shehory (eds.), May 6–10, 2013, Saint Paul, Minnesota, USA. Copyright © 2013, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

ger a further cascade of adjustments by other agents to their schedules. Here we extend the original definition of the *Multiagent Temporal Decoupling Problem (MaTDP)* [3, 1], which was previously defined based on problems containing strictly conjunctive constraints. Agents’ local DTP subproblems form a *temporal decoupling* of a consistent MaDTP \mathbf{D} if (i) each agent’s local DTP subproblem is consistent; and (ii) *any* combination of solutions to each agent’s local DTP subproblem yields a joint solution to \mathbf{D} . The MaTDP is defined as finding, for each agent, a set of additional constraints (e.g., tighter bounds on the timings of activities) that, when added to the agent’s local DTP, creates a temporal decoupling of MaDTP \mathbf{D} . Figure 1(b) represents a decoupling of the MaDTP in Figure 1(a), where any solution to agent A ’s DTP in the top row can be combined with any solution to agent B ’s DTP in the bottom row to form a joint solution. A challenge is that finding a decoupling requires ensuring that at least one of the solutions to the MaDTP, if any exist, must survive the decoupling [3], and so is an NP-hard problem. A second challenge is that disjunctive temporal constraints involve arbitrarily many pairs of variables, and so may induce combinatorially many different network structures, making efficient representation of the set of these possible structures particularly challenging.

2. INFLUENCE-BASED DECOUPLING

Our approach for decoupling the MaDTP builds on our distributed MaDTP Local Decomposability (MaDTP-LD) algorithm [2] for computing the entire MaDTP solution space. The MaDTP-LD algorithm recognizes that not all local solutions qualitatively change how an agent’s problem will impact other agents. Thus, instead of computing its entire solution space, an agent can instead focus on computing its *influence space*, the space of solutions that lead to distinct assignments of its externally constrained variables. An agent’s influence space summarizes how its local constraints impact other agents so that all coordination can be limited to these smaller influence spaces. Our approach differs from the MaDTP-LD algorithm by incorporating information from the shared DTP as early and often as possible, rather than waiting for each agent to completely enumerate its local influence space before shared reasoning occurs. Incorporating shared information has the effect of pruning globally infeasible schedules from an agent’s local search space early on and then, once a temporal decoupling as been found, short-circuiting agents’ reasoning by eliminating those local schedules that are no longer consistent with respect to the new decoupling constraints. The shared DTP solution space can be thought of as the cross-product of agents’ influence spaces. Thus, as agents construct their local influence spaces, they can also build the shared DTP solution space in a way that is provably sound and progressively more complete over time. Then, as soon as a solution to the shared DTP is found, it can be used to construct and install a temporal decoupling, which in turn saves computing the entire joint solution space, representing a potentially combinatorial savings. Our approach is both provably sound and complete.

3. DISCUSSION

We compare our new decoupling approach against our MaDTP-LD algorithm that computes the entire joint solution space, replicating our previous experimental setup [2].

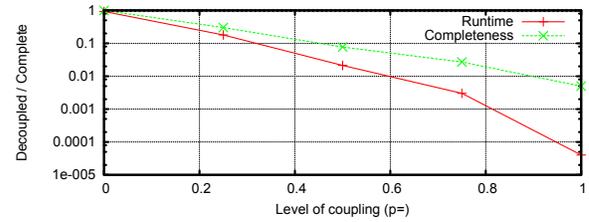


Figure 2: Decoupling vs. complete approaches

We measure the maximum processing time across agents (i.e., the time the last agent completes execution) and the number of distinct, consistent local temporal constraint networks. As shown in Figure 2, for problems containing just two agents, as the proportion of external constraints increases (p), our decoupling approach demonstrates upwards of a four orders-magnitude decrease in runtime over the complete MaTDP-LD algorithm. This is because as soon as agents find a decoupling, they immediately commence with finding only solutions that are consistent with the new decoupling constraints rather than fully enumerating the entire joint solution space. However, Figure 2 also illustrates that these gains come at the cost of limiting the completeness of local solution spaces as measured by the number of distinct consistent local temporal networks. This limits the amount of flexibility an agent has to react to scheduling disturbances.

In conclusion, we discuss a new distributed, decoupling approach for calculating solution spaces to MaDTPs where agents independently and incrementally build their influence spaces until a valid temporal decoupling can be found. Overall, we believe the gains in runtime efficiency of our decoupling approach over the MaDTP-LD algorithm outpace the relative sacrifices in solution space completeness—our approach solves loosely-coupled problems containing 64 agents in under a second while maintaining at least a tenth of all consistent local temporal networks, whereas the MaDTP-LD algorithm consistently exceeds 100 seconds for problems with only 4 agents. In the future, we would like to investigate optimal and heuristic variants of our decoupling approach where, for example, agents produce influence spaces in a best-first manner in an attempt to guide the coordinator to a more flexible temporal decoupling in an anytime manner.

4. ACKNOWLEDGMENTS

We thank the anonymous reviewers for their suggestions. This work was supported, in part, by NSF under grant IIS-0964512, and by a UM Rackham Fellowship

5. REFERENCES

- [1] J. Boerkoel and E. Durfee. Distributed algorithms for solving the multiagent temporal decoupling problem. In *Proc. of AAMAS-11*, pages 141–148, 2011.
- [2] J. Boerkoel and E. Durfee. A distributed approach to summarizing spaces of multiagent schedules. *Proc. of AAAI-12*, pages 1742–1748, 2012.
- [3] L. Hunsberger. Algorithms for a temporal decoupling problem in multi-agent planning. In *Proc. of AAAI-02*, pages 468–475, 2002.
- [4] K. Stergiou and M. Koubarakis. Backtracking algorithms for disjunctions of temporal constraints. *Artificial Intelligence*, 120(1):81–117, 2000.