

A Multiagent System for Solving the Activity Selection and Scheduling Coordination Problem

James C. Boerkoel Jr.

Department of Computer Science and Engineering
University of Michigan
boerkoel@umich.edu

1 Introduction

Deadline pressures, unexpected events, and combinatorial numbers of possible courses of action often lead a person to decide which activity she will begin next without considering a full enumeration of possible schedules, and thus, without a full awareness of the implications that her choice will have on the rest of her day's schedule. Furthermore, people suffering from cognitive impairments may lack the abilities to perform such reasoning in the first place. The goal of my thesis is to develop foundational technologies for computational agents that augment the abilities of people who face the above challenges to reason about the implications of scheduling. In particular, I develop, integrate, and evaluate new techniques for solving multi-agent Hybrid Scheduling Problems, which support coordinated activity selection and scheduling for human users.

2 Problem Representation

In my work, I show the suitability of using the Hybrid Scheduling Problem (HSP) [Schwartz 2007] for representing the constraints and variables of scheduling problem (see Figure 1 for a small example). The HSP combines typical finite-domain CSPs that are capable of representing selection problems, with Disjunctive Temporal Problems (DTP) that can represent (possibly disjunctive) temporal difference constraints between timepoints. The HSP combines finite-domain and temporal reasoning with hybrid constraints, which are disjunctive choices between enforcing a finite-domain constraint or temporal constraint. This formulation allows my algorithms to answer scheduling questions of not only when to perform activities, but also which activities to perform and how to perform them.

Temporal consistency is determined in polynomial time by applying an all-pairs-shortest-path algorithm to detect negative cycles [Schwartz 2007]. Maintaining a consistent, tightened temporal network of all the problem timepoints is a key component of many constraint-based scheduling algorithms, since it helps prune the scheduling search space.

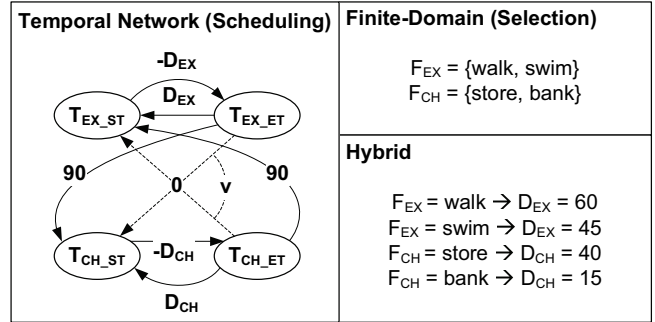


Figure 1. Finite-domain variables F_{EX} and F_{CH} represent the possible selections for completing my exercise and my chore, respectively, a non-disjunctive temporal difference constraint ($T_{CH_ET} - T_{EX_ST} \leq 90$) represents that I must complete my chores within 90 minutes of starting my exercise, a disjunctive temporal constraint ($T_{CH_ET} - T_{EX_ST} \leq 0 \vee T_{EX_ET} - T_{CH_ST} \leq 0$) represents that either exercise must end before chores, or vice versa, and finally, hybrid constraints dictate that the bounds on duration constraints depend on which activity is selected to be performed.

My work aims to increase efficiency in HSP search, in part, by realizing further opportunities to consistently tighten this temporal network in order to prune the search space.

3 Hybrid Constraint Tightening

In my past work, I show how a polynomial time meta-level introspection technique called Hybrid Constraint Tightening (HCT) lifts information from the structure of scheduling problems, leading to reformulated constraints that assist solution techniques in achieving orders of magnitude speedup over untightened constraints [Boerkoel and Durfee 2008, 2009]. HCT works by reformulating groups of hybrid constraints in a way that applies tighter temporal bounds, sooner, tightening the underlying temporal network to promote search space pruning, exploiting the expressiveness of HSP and the solution strategies of off-the-shelf solvers to greatly enhance the efficiency of the overall integration of the two individual components. Additionally, I have explored properties of HSPs that influence HCT's efficacy, quantifying the influence empirically [Boerkoel and Durfee 2009].

4 Asynchronous Interval Maintenance

My ongoing work is related to establishing temporal consistency in a temporal network distributed among

multiple agents. I propose an algorithm that exploits the structure of problems where there are relatively few inter-agent constraints and uses a reference timepoint to promote decoupling and concurrency. For this algorithm, I demonstrate empirically that the synchronous computation time and number of message cycles scales sub-linearly with the number of agents, the number inter-agent constraints, and the size of the intra-agent problem. While my current distributed temporal consistency algorithm involves some synchrony to establish consistency, I hope to extend this to an asynchronous version that I call Asynchronous Interval Maintenance (AIM). The goal of AIM is to asynchronously increase the temporal consistency between agents, allowing AIM to be interleaved into current, asynchronous distributed CSP search techniques leading to increased pruning in distributed temporal search.

5 Solving the Multiagent HSP

Efficient CSP search algorithms strike a balance between proposing commitments (assignments of values to variables) and least-commitment practices, such as achieving higher levels of arc consistency. Many current algorithms for solving distributed (finite-domain) CSPs focus heavily on either asynchronous commitment-based practices, which forward check implicitly through no-good exchanges, or on asynchronous forward-checking at the cost of some concurrency. My goal is to interleave the higher temporal consistency achieved through AIM with the principles from current distributed CSP algorithms to develop and evaluate more efficient algorithms for solving the distributed HSP. This will require heuristics for deciding when and how to effectively explore an agent's local search space for information that will highly impact other agents. Additionally, I hope to develop heuristics for deciding which inter-agent constraints and which incremental temporal consistency updates warrant the cost of a message exchange. I will evaluate such techniques in terms of their communication and synchronous time costs.

5 Commitment Abstraction Model

An important open question in this research is to find the appropriate level of abstraction at which to make scheduling decisions. Agents must commit to enough scheduling detail to guarantee the feasibility of the schedules they propose and provide sufficient feedback to the user. However, there may be significant computational and representational benefits for delaying certain scheduling decisions until these decisions are relevant. To answer this question, we are developing a simulation environment based on the Repast toolkit [Collier] that can be populated with many agents, each subject to environmental constraints and stochastic behavior. We hope to inform this simulation environment with opinions from experts in activity coordination applications, such as cognitive impairment care. At one extreme, we could fully instantiate all details of a schedule,

though it is likely that many details will be forced to change due to the dynamics of the user's environment. A constantly shifting and updating schedule could be perceived as burdensome and annoying to a user and could incur the unnecessary complexity of specifying scheduling details that may be unlikely to play out exactly as planned. At the other extreme, we could create a system that simply reacts to the environment, only making advisory or admonitory comments immediately as the user is about to make a choice that could heavily influence the success or utility of his or her schedule. While the solver prevents specifying many potentially irrelevant search details, in this scenario the user gets very little feedback, and may make decisions that limit him/herself in the future. Real-world user communities will likely demand a model somewhere between these two extremes. The correct choice depends on the efficiency of the multiagent HSP algorithm and will dictate which system elements require efficiency improvements, and when and how we can exploit periods of low system demand to prepare for periods of more up-tempo scheduling demands.

6 Discussion

The goal of this thesis centers on creating enabling technologies. Accomplishing this requires successful synthesis of many individual technologies that are themselves efficient. Approaches that I use to integrate individual components, such as abstraction and problem structure exploitation through introspection, are powerful tools that can be used when integrating such technology for use in other contexts. As I integrate my HCT and decoupled optimization contributions with commitment abstraction and communication models to create agents that assist people in solving interesting activity selection and scheduling problems, I will discuss how the lessons generalize to many activity selection, scheduling, search and multiagent applications.

References

- [Boerkoel and Durfee 2009] Boerkoel, J. and Durfee E., 2009. Evaluating Hybrid Constraint Tightening for Scheduling Agents, *Proc. of AAMAS 2009*, May, 10–15, 2009, Budapest, Hungary, To Appear.
- [Boerkoel and Durfee 2008] Boerkoel, J. and Durfee, E. 2008. Hybrid Constraint Tightening for Hybrid Constraint Scheduling. In *Proc. Of AAAI-2008*, 1446-1449.
- [Collier] Collier, N. RePast: An Extensible Framework for Agent Simulation.
<http://www.econ.iastate.edu/tesfatsi/RepastTutorial/Collier.pdf>.
- [Schwartz 2007] Schwartz, P. 2007. Managing Complex Scheduling Problems with Dynamic and Hybrid Constraints. PhD. Diss., CSE., Univ. of Mich., Ann Arbor.