A COMPARISON OF ALGORITHMS FOR SOLVING THE MULTIAGENT SIMPLE TEMPORAL PROBLEM

JIM BOERKOEL AND ED DURFEE
COMPUTER SCIENCE AND ENGINEERING, UNIVERSITY OF MICHIGAN
Motivation

- Consider Amy’s agenda:
  - Study for exam
  - Take exam
  - Work on group project
    - Must exchange project deliverables with partner Ben
  - Work on research project
Motivation

- How does Amy choose a schedule for accomplishing her agenda that is compatible with Ben’s schedule?
  - Option 1: Ignore Ben
    - Schedule may fail to coordinate with Ben’s
  - Option 2: Collect Ben’s scheduling commitments / constraints, and choose a compatible joint schedule
    - Ben may not want to reveal private schedule commitments
    - Introduces extra burden on Amy, which grows with every person she coordinates with

Amy’s Agenda:
- Study session (SS)
- Exam
- Group Project (GP)
- Research Project (RP)
This talk introduces multiagent scheduling algorithms that:

- Find complete set of sound joint schedules
- Exploit the problem’s structure and natural distribution across computational agents to concurrently compute joint schedules and achieve speedup over centralized algorithms
- Have provable privacy properties
Background: Simple Temporal Problem (STP)

- A temporal CSP
- Timepoint Variables (V)
  - Represent events
  - Continuous (infinite) domain
- Temporal Difference Constraints (E)
  - Constraints are represented by a bound on the difference between two variables
  - Represented graphically with directed edges

Amy's Agenda:
- Study session (SS)
- Exam
- Group Project (GP)
- Research Project (RP)
Extending to Multiagent STP (MaSTP)

- A MaSTP is composed of $n$ agent subproblems
- For each agent problem, the set of constraints is composed of intra-agent and inter-agent constraints

<table>
<thead>
<tr>
<th>Amy’s Agenda:</th>
<th>Ben’s Agenda:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Study session (SS)</td>
<td>- Programming Assignment (PA)</td>
</tr>
<tr>
<td>- Exam</td>
<td>- Homework (HW)</td>
</tr>
<tr>
<td>- Group Project (GP1)</td>
<td>- Group Project (GP2)</td>
</tr>
<tr>
<td>- Research Project (RP)</td>
<td>- Exercise (RUN)</td>
</tr>
</tbody>
</table>
Example MaSTP

Amy's Agenda:
- Study session (SS) (GP1)
- Exam
- Research Project (RP)

Ben's Agenda:
- Group Project (GP2)
- Prog. Assign. (PA)
- Homework (HW)
- Exercise (RUN)

Intra-agent constraints:
- SS.ST [30, 240] [0, ∞]
- EXAM.ST [20, 60] [0, ∞]
- GP1.ST [30, 90] [0, ∞]
- RP.ST [90, 120] [0, ∞]

Inter-agent constraints:
- SS.ET [0, 0]
- EXAM.ET [0, 0]
- GP1.ET [0, 0]
- RP.ET [0, 0]

Amy's Agenda:
- Study session (SS) (GP1)
- Exam
- Research Project (RP)

Ben's Agenda:
- Group Project (GP2)
- Prog. Assign. (PA)
- Homework (HW)
- Exercise (RUN)

Intra-agent constraints:
- SS.ST [30, 240] [0, ∞]
- EXAM.ST [20, 60] [0, ∞]
- GP1.ST [30, 90] [0, ∞]
- RP.ST [90, 120] [0, ∞]

Inter-agent constraints:
- SS.ET [0, 0]
- EXAM.ET [0, 0]
- GP1.ET [0, 0]
- RP.ET [0, 0]
Establishing Decomposability

- **Decomposable STP:**
  - Represents a complete set of solutions using ranges of times for each event, where each time can be extended to a sound schedule.

- **Full Path Consistency**
  - All-pairs-shortest-path
  - Calculate min/max time between Amy’s study session and Ben’s run?

- **Partial Path Consistency**
  - Step 1: Triangulate graph
  - Step 2: Tighten triangles
Our Approach

- Goals
  - Soundness
  - Concurrency
  - Privacy
- Partition the MaSTP into $n+1$ subproblems:
  - $n$ Private STPs: for each agent, the timepoints involved in NO inter-agent constraints, and the constraints involving them
  - 1 Shared STP: the timepoints involved in inter-agent constraints, and the constraints between them
Multiagent STP Partitioning

The set of shared timepoint variables include all variables that DO participate in an interagent constraint.

- Study session (SS) (GP1)
- Exam
- Group Project (RP)

The set of private timepoint variables include all variables that DO NOT participate in an interagent constraint.

- Prog. Assign. (PA) (GP2)
- Homework (HW)
- Exercise (RUN)

Amy’s Agenda:
- Study session (SS) (GP1)
- Exam

Ben’s Agenda:
- Prog. Assign. (PA) (GP2)
- Homework (HW)
- Exercise (RUN)
Privacy Properties of our Multiagent STP Algorithms

- The information an agent must reveal to (or conversely learn of) another agent is necessarily limited to the shared STP
- Everything else remains private!
Adapt state-of-the-art partial path consistency algorithm, P^3C [Planken, de Weerdt, van der Kriempt 2008], to explore our partitioning.
We focus on private timepoints, and can ignore interagent constraints.

Both agents can begin concurrently triangulating and updating their private STPs.
Partially Centralized: Shared
Amy's Agenda:
- Study session (SS) (GP1)
- Exam
- Research Project (RP)

Ben's Agenda:
- Prog. Assign. (PA) (GP2)
- Homework (HW)
- Exercise (RUN)
Solving a Multiagent STP: Summary

- 6 constraint checks per triangle
- Total constraint checks (centralized): 132 (22 triangles)
  - Total shared constraint checks: 12 (2 triangles)
  - Total private constraint checks per agent: 60 (10 triangles)
- Partially Centralized approach: 72
- Distributed approach: 66
Empirical Evaluation

- Randomly generated problems with 25 agents, 25 timepoints per agent
- Vary parameter $P$ – the proportion of timepoints that are private
- Number of constraints scaled so that centralized computation remains constant
- Record non-concurrent constraint checks (and messages)
When $P$ is low, most timepoints are shared, so most of the problem is solved centrally.

As $P$ grows (and shared STP shrinks), partially centralized algorithm approaches perfect speedup (25x).
Computation: Non-concurrency

Distributed algorithm achieves better load-balancing for solving shared STP

Assumes zero message latency!
Computation: Non-concurrency

 Assumes message latency equal to computational time for a constraint
Computation: Non-concurrency

Assumes message latency equal to 10x constraint check time

Regardless of communication costs, our algorithms perform well on loosely coupled problems!
By exploiting the weakly-coupled structure of multiagent STPs, our partially centralized and distributed algorithms achieve significant solution time speedup through concurrency.

Our partially centralized and distributed algorithms maintain a high-level of user privacy.

Exploiting timepoint partitioning information can lead to smaller triangulated graphs (result not shown).

Future work: Incorporate Multiagent STP algorithms as the foundation for more complex scheduling agents that can coordinate schedules on behalf of users.
Thanks!

- Questions?
References

Future Work

- Develop multiagent approaches for solving:
  - Disjunctive Temporal Problems
  - Hybrid Scheduling Problems
  - Preferences
  - Evaluate in a dynamic environment
Computation: Scalability

![Graph showing scalability across different numbers of agents. The graph compares nonconcurrent computation time for different categories: Centralized (Cent.), Partially Centralized (Part. Cent.), Distributed + Message (Dist. + Mess.), and Distributed (Dist.).](image-url)
Number of Fill Edges (triangles)

Number of Fill Edges

Private to Global Timepoint Ratio ($P$)

- Distributed
- Partially Centralized
- Centralized
The private-before-shared restriction helps the variable elimination heuristic exploit problem structure!
Solving a STP: Partial Path Consistency

- All-pairs-shortest-path
- Step 1: Triangulate
  - Triangulated graph
    - A graph whose largest non-bisected cycle is of size 3
  - Algorithm
    - Remove node
    - Moralize
    - Repeat
  - Try to minimize # of triangles
Step 2: Tighten STP
- Add all $\Delta$’s to a queue, $Q$
- Until $Q$ is empty
  - $\Delta = Q.dequeue()$
  - Tighten($\Delta$)
  - Enqueue any affected neighboring $\Delta$

Add to queue

-45

-75

-45 + -30 < -60
-45 + -30 = -75
Solving a Multiagent STP: Shared
Solving a Multiagent STP: Shared
Solving a Multiagent STP: Private1

Amy's Agenda:
- Study session (SS) (GP1)
- Exam
- Research Project (RP)

Ben's Agenda:
- Group Project (GP2)
- Prog. Assign. (PA)
- Homework (HW)
- Exercise (RUN)

Amy
- SS.ST [30, 240] SS.ET [0, ∞]
- EXAM.ST [20, 60] EXAM.ET [0, ∞]
- GP1.ST [30, 90] GP1.ET [0, 0]
- RP.ST [50, 360] RP.ET [120, 405]
- 8AM1 [-∞, 390] [-∞, 480]

Ben
- PA.ST [60, 120] PA.ET [0, ∞]
- HW.ST [60, 90] HW.ET [0, ∞]
- GP2.ST [45, 120] GP2.ET [0, ∞]
- RUN.ST [30, 60] RUN.ET [0, 0]
- 8AM2 [-∞, 450] [-∞, 480]
Solving a Multiagent STP: Private 1

Amy’s Agenda:
- Study session (SS) (GP1)
- Exam
- Research Project (RP)

Ben’s Agenda:
- Prog. Assign. (PA) (GP2)
- Homework (HW)
- Exercise (RUN)

Amy

Study session (SS) [30,240] [0,∞]
Exam [20,60] [0,∞]
Group Project (GP1) [30,90] [0,∞]
Research Project (RP) [90,120] [0,∞]
8AM1 [∞,390] [-∞,480] [0,0]

Ben

Prog. Assign. (PA) [60,120] [0,∞]
Homework (HW) [60,90] [0,∞]
Group Project (GP2) [45,120] [0,∞]
Exercise (RUN) [-∞,450] [-∞,480] [0,0]
8AM2 [∞,450] [-∞,480] [0,0]
Private STPs

Amy's Private STP

- SS.ST
- EXAM.ST
- GP1.ST
- RP.ST
- 8AM1

Ben's Private STP

- PA.ST
- HW.ST
- GP2.ST
- RUN.ST
- 8AM2
Solving a Multiagent STP: Private2