Resolving Over-constrained Temporal Problems with Uncertain Durations

Peng Yu
March 7, 2014
Over-constrained temporal problems can be better resolved by relaxing the temporal constraints continuously, instead of removing them discretely.

The fundamental concepts of conflicts and minimal relaxations naturally generalize to the continuous case.

Restoring controllability requires both tightening uncertain durations and relaxing constraints.

We can efficiently enumerate relaxations in best-first order, by generalizing the Conflict-Directed A* algorithm, first developed for diagnosis.
Ongoing Projects

- Personal Transportation System.

- Trip advisor for commuters.

- Mission advisory system for deep sea exploration.
Robotic Personal Transportation System

• A personal air taxi with an intelligent trip advisory
Key features

- Find alternative solutions that are **simple** and **preferred**.
- Provide **insights** into cause of failure and its resolution.
  - Minimize the perturbations;
  - Prioritize alternatives;
  - Explain the cause of failure;
  - Adapt incrementally to new constraints.

“Delay your arrival by 5 minutes”.

“OK, then how about having lunch at restaurant Y”.

“Because of the extended travel time”.

“if you want to shop for at least 25 minutes, you can have lunch at restaurant Y for 55 minutes”.
It is **6pm** now and Brian is leaving NICTA for home.

- He wants to **be home in 40 minutes**, and is only willing to take buses.

- Right now, he is looking up Google Map for directions…
Which Bus To Take

• Google Map returns two options (leaving NICTA at 1800), ranked based on trip duration

• Option 1:
  – Take the **18:08 Bus #3** (Ride time 23 mins).
  – Walking to departure stop: 8 mins.
  – Walking from arrival stop to home: 3 mins.

• Option 2:
  – Take the **18:11 Bus #934** (Ride time 26 mins).
  – Walking to departure stop: 10 minutes.
  – Walking from arrival stop to home: 3 minutes.
Uncertainty Affects Our Decision

• Buses may be late or early:
  – Bus #3: 18:08 ± 2 minutes.
  – Bus #934: 18:11 ± 1 minute.

• Brian may miss the bus if he takes the Google preferred option.
Cope With the Uncertainty

• “You can catch Bus #934 and arrive home 3 minutes late.”
• “Or, you can take Bus #3 and arrive home on time, but taking the risk of missing the bus, if it arrives early.”
During an expedition cruise, the chief scientist needs assistance for planning and scheduling activities, especially when things go wrong.

- Task sequencing and scheduling.
- Goal relaxation and failure recovery.
- Human resources and assets management.
A WHOI Mission

• Duration: Sep 26\textsuperscript{th} – Oct 17\textsuperscript{th}.
• Vessel: R/V Atlantis.
• Location: Along the coast between SF and LA.
• Objectives:
  – Find and sample methane seeps near the coast.
  – Locate and sample a 60 year-old DDT dumping site.
  – Recover and replace incubators on the seafloor.
A 3-day Plan From the Cruise

Draft Cruise Plan 9/28-9/30

9/27/13
Sentry Dive at Partington Canyon 6hrs Target start time 2000

9/28/13
Depart Partington Canyon Estimated Departure Time 0230
Transit to Paull’s Pingo 27 hrs ETA 0530 hrs (9/29/13)
Science Meeting 10AM!
Multibeam pass of SBB-2H Pockmark
Multibeam pass in Southern SB Channel and D to D’
Multibeam pass of SW Mounds area

9/29/13
Arrive at Paull’s Pingo ETA 0530 hrs
Jason Operations at Paull’s Pingo 15 hrs Deploy by 0730; 2 Elevator Deployments
Transit to SW mounds 1.5 hrs Arrive SW Mounds ~2400

9/30/13 and beyond
Sentry deployment at SW Mounds 16 hrs Deploy at 0000
Jason Deployment at SW Mounds 24 hrs Multiple Elevators
Everything can Go Wrong

- [Day 1] Jason failed after 30 min into its first dive, entered an uncontrollable spin and broke its optic fiber tether.

- [Day 1] The new camera installed on Sentry did not work well in low light situations. It had been replaced during its second dive.
• [Day 2] Jason entered an uncontrollable spin and broke its optic fiber tether again during its second dive. It turned out that there is a bug in its newly updated code.

• [Day 3] Sentry’s mass spectrometer failed during its second dive. They sent Rich to Pittsburg to get it fixed.

... ...  

• [Day 7] Sentry aborted its mission 1 hour after launch. Atlantis aborted its mapping routes and went back to recover Sentry. The failure was caused by a burned wire.
Our Deliverable

• A mission advisory system that assists the chief scientists of expeditions on the following tasks:
  – Scheduling Activities with Uncertainty.
  – Failure and Downtime Recovery Scheduling.
  – Assets Managements.
Contents

• Relaxations of Conditional Temporal Problems;

• Continuous Relaxation and Conflict Resolution;

• Restoring Controllability with Uncertainty Durations;

• Best-first Enumeration through Conflict-directed Relaxation;

• Experiments.
Problem Formulation

• Model: (Over-constrained) Controllable Conditional Temporal Problems with Uncertainty.
  – All choices are **controllable**.
  – Allowing temporal constraints to be **relaxed**.
  – Allowing uncertain durations to be **tightened**.

• A solution is a pair with:
  – A complete set of **decisions**.
  – A set of continuous **relaxations** for temporal constraints.
  – A set of continuous **tightening** for uncertain durations.
  such that the set of activated durations and constraints is consistent/controllable.
Define User Preferences

- Preference functions are defined over decisions and constraint relaxations.
  - Each decision is mapped to a positive reward by function $f_p$.
  - Each constraint relaxation/duration tightening is mapped to a positive cost by function $f_e$.

<table>
<thead>
<tr>
<th>Store</th>
<th>A</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>B</td>
<td>100</td>
</tr>
<tr>
<td>Lunch</td>
<td>X</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>Y</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>Z</td>
<td>30</td>
</tr>
</tbody>
</table>

Assignment: $\{Store = B, Lunch = Y\}$
Reward: $100 + 80 = 180$

Relaxation: $Reservation[0,180] \rightarrow [0,200]$
Cost: $f_e(200 - 180) = 40$
Contents

- Relaxations of Conditional Temporal Problems;
- **Continuous Relaxation** and Conflict Resolution;
- Restoring Controllability with Uncertainty Durations;
- Best-first Enumeration through Conflict-directed Relaxation;
- Experiments.
 Resolve over-constrained temporal problem $C$ by removing constraints.  
  - Resolved: $M \subseteq C$ such that $C \setminus M$ is consistent.  
  - Minimal: $\forall c \in M$ $(C \setminus M) \cup \{c\}$ is inconsistent.

**Remove arrival:**

**Remove lunch:**

OR
Continuous Relaxation

- Relax a constraint partially by **continuously** modifying its temporal bounds:
  - A continuous relaxation, $CR_i$, weakens a temporal constraint: $[LB, UB] \rightarrow [LB', UB']$ where $LB' \leq LB$ and $UB' \geq UB$.
  - Continuous relaxations only apply to **relaxable** constraints.

“Shorten lunch to 25 minutes and delay arrival by 5 minutes”
Contents

• Relaxations of Conditional Temporal Problems;

• Continuous Relaxation and Conflict Resolution;

• Restoring Controllability with Uncertainty Durations;

• Best-first Enumeration through Conflict-directed Relaxation;

• Experiments.
1. Learn Discrete Conflicts

- A discrete conflict is an inconsistent set of temporal constraints.

Choosing Store=B and Lunch=Y produces:

Discrete Conflict: Store = B; Home → B ≥ 35; Drive B → Y ≥ 25; Lunch at Y ≥ 75; Y → Home ≥ 40; Arrive Home ≤ 180.
2. Weaken to Continuous Conflicts

- A continuous conflict is an equation formed from the discrete conflict.
- It specifies the deviation needed to resolve the conflict.

Discrete Conflict:

HometoB $\geq 35$;
ShopatB $\geq 35$;
BtoY $\geq 25$;
LunchatY $\geq 75$;
YtoHome $\geq 40$;
ArriveHome $\leq 180$.

Continuous Conflict:

$\text{ArriveHome} - \text{HometoB} - \text{ShopatB} - BtoY - LunchatY - YtoHome = -30$
3. Map to Constituent Continuous Relaxations

- Relaxations specified by linear inequalities:

\[
\Delta_{ShopatB} + \Delta_{LunchatY} + \Delta_{ArriveHome} \geq 30
\]

\[
\text{ArriveHome} - \text{Home to } B - \text{Shop at } B - BtoY - Lunch at Y - YtoHome = -30
\]
Discrete vs. Continuous Relaxations

- Resolve a conflict by relaxing constraints completely or partially.

**Conflict:**

Store = B, Lunch = Y;
Home → B ≥ 35; Shop at B ≥ 35;
Drive B → Y ≥ 25; Lunch at Y ≥ 75;
Y → Home ≥ 40; Arrive Home ≤ 180.

**Discrete Resolutions**
- Remove Shop at B ≥ 35;
- Remove Lunch at Y ≥ 75;
- Remove Arrive Home ≤ 180

**Continuous Resolutions**
- Lunch at Y ≥ 45;
- Arrive Home ≤ 210;
- Shop at B ≥ 25 and Lunch at Y ≥ 55;
- ... ...
- and many more
• Relaxations of Conditional Temporal Problems;

• Continuous Relaxation and Conflict Resolution;

• Restoring Controllability with Uncertainty Durations;

• Best-first Enumeration through Conflict-Directed Relaxation;

• Experiments.
Learn Conflicts From Uncontrollable Problems

- Learning conflicts from controllability checking algorithms is difficult.
  - For consistency checking, there is a one-to-one mapping between the distance edges and the bounds of constraints.
  - No such mapping exists for controllability checking (strong and dynamic) due to the reduction procedures, making it difficult to extract conflicts from the reduced graph.

- Key: during the reduction, record the ‘contribution’ of each constraint and duration in the temporal problem.
A Strong Controllability Example

1. S1 \( \rightarrow \) E1
   - A: \([5, 10]\)
   - D: \(\geq 4\)
   - B: \([1, 2]\)

2. S1 \(\rightarrow\) E1
   - 10: \(A_u\)
   - -5: \(A_l\)
   - -4: \(D_l\)

3. S1 \(\rightarrow\) E1
   - 10: \(A_u\)
   - -5: \(A_l\)
   - -4: \(D_l\)
   - -2: \(C_l, B_u\)

4. S1 \(\rightarrow\) E1
   - 10: \(A_u\)
   - -5: \(A_l\)
   - -4: \(D_l\)
   - 3: \(C_l, B_u, A_l\)

Resolving Over-constrained Temporal Problems with Uncertain Durations
Resolving Uncontrollable Conflicts

- Constraint for resolving continuous conflict (negative value -1):

\[ \Delta D_L + \Delta C_L + \Delta B_U + \Delta A_L \geq 1 \]

where:

- \( \Delta C_L, \Delta D_L \) are relaxations for C and D.
- \( \Delta A_L, \Delta B_U \) are tightening for A and B.

and

\[ \Delta A_L \leq 5; \Delta B_U \leq 1. \]
Learn Dynamically Uncontrollable Conflict

- Record supporting constraints and durations during the iterative reduction procedure, and extract conflicts using the them.

```
1  DG ← GETNORMALDISTANCEGRAPH(T);
2  for l to K do
3     NCycle ← ALLMAXCONSISTENT(DG);
4     if NCycle == null then
5         for E in LOWERCASEEDGES(DG) do
6             moatPaths ← PROPAGATE(E);
7             for Path in moatPaths do
8                 E' ← REDUCE(E, Path);
9                 SUPPORTS(E') ← SUPPORTS(E, Path);
10                ADDTOGRAPH(E', DG)
11         end
12     else
13         return GETSUPPORTS(NCycle);
14     endif
15  end
16  NCycle ← ALLMAXCONSISTENT(DG);
17  return GETSUPPORTS(NCycle);
```

Contents

• Relaxations of Conditional Temporal Problems;

• Continuous Relaxation and Conflict Resolution;

• Restoring Controllability with Uncertainty Durations;

• Best-first Enumeration through Conflict-Directed Relaxation;

• Experiments.
Conflict-Directed A\(^*\) (Williams and Ragno, 2004) can be applied to discrete relaxation problems:

- Efficiently prunes search space using learned conflicts.
- Enumerates minimal discrete relaxations in best-first order.

To solve a relaxation problem:

- Frame an equivalent constraint optimization problem.
  - Discrete relaxation: add binary variables.
  - Continuous relaxation/tightening: add non-negative continuous variables.
- The objective function represents the preference.
Best-first Conflict Directed Relaxation

- BCDR generalizes the conflict resolution procedure in CDA* to include constituent continuous relaxations.
• Key Ideas:
  – Split on conflict;
  – Best-first enumeration.

Conflict-Directed A*
CDA* with Constituent Continuous Relaxation

- Split a conflict using its constituent continuous relaxations.

\[
\Delta_{\text{Shop at } B} + \Delta_{\text{Lunch at } Y} + \Delta_{\text{Arrive Home}} \geq 30
\]

\[
\min (f(\Delta_{\text{Shop at } B}) + f(\Delta_{\text{Lunch at } Y}) + f(\Delta_{\text{Arrive Home}}))
\]

s.t. \ \Delta_{\text{Shop at } B} + \Delta_{\text{Lunch at } Y} + \Delta_{\text{Arrive Home}} \geq 30
Continuous Relaxations for Multiple Conflicts

- For two or more continuous relaxations on the same branch, the utility is determined by the grounded solution that respects both inequalities.

\[ \Delta_{\text{Shop at } B} + \Delta_{\text{Lunch at } Y} + \Delta_{\text{Arrive Home}} \geq 30 \]

\[ \Delta_{\text{Drive to } B} + \Delta_{\text{Drive } B \text{ to } Y} + \Delta_{\text{Travel Time}} \geq 10 \]

\[ \min (f(\Delta_{\text{Shop at } B}) + f(\Delta_{\text{Lunch at } Y}) + f(\Delta_{\text{Arrive Home}}) + f(\Delta_{\text{Drive to } B}) + f(\Delta_{\text{Drive } B \text{ to } Y}) + f(\Delta_{\text{Travel Time}})) \]

s.t.

\[ \Delta_{\text{Shop at } B} + \Delta_{\text{Lunch at } Y} + \Delta_{\text{Arrive Home}} \geq 30 \]

and

\[ \Delta_{\text{Drive to } B} + \Delta_{\text{Drive } B \text{ to } Y} + \Delta_{\text{Travel}} \geq 10 \]
Incorporating User Responses

- BCDR incrementally adapts to new requirements.
- These requirements are recorded as new conflicts.

Required Continuous Relaxations

\[ \Delta_{\text{Arrive Home}} \leq 0; \]

\[ \Delta_{\text{Shop at B}} \leq 10; \]

- No, I do not want to extend my reservation time.
- No, I want to spend at least 25 minutes on shopping.
New Requirements as Conflicts

- Expand search tree using user response conflicts.

\[
\Delta_{ShopatB} + \Delta_{LunchatY} + \Delta_{ArriveHome} \geq 30
\]

\[
\min (f(\Delta_{ShopatB}) + f(\Delta_{LunchatY}) + f(\Delta_{ArriveHome}))
\]

\[
\text{s.t. } \Delta_{ShopatB} + \Delta_{LunchatY} + \Delta_{ArriveHome} \geq 30;
\Delta_{ArriveHome} \leq 0;
\Delta_{ShopatB} \leq 10.
\]
Split on Conflicts for Conditional Problems

- If a node has an unresolved conflict, we expand it using both constituent **continuous** relaxation and **decisions** that deactivates its constraints.

Store=B; Lunch=Y;
Home → B ≥ 35;
Shop at B ≥ 35;
Drive B → Y ≥ 25;
Lunch at Y ≥ 75;
Y → Home ≥ 40;
Arrive Home ≤ 180.
Contents

• Relaxations of Conditional Temporal Problems;

• Continuous Relaxation and Conflict Resolution;

• Restoring Controllability with Uncertainty Durations;

• Best-first Enumeration through Conflict-directed Relaxation;

• Experiments.
Experiment Setup

- We simulated a car-sharing network in Boston using randomly generated car locations and destinations.

- Test cases are characterized by:
  - Number of reservations per car.
  - Number of cars in the network.
  - Number of activities per reservation.
  - Number of alternative options per activity.

- Time change may affect neighboring reservations.
Empirical Results - Runtime

- We compare the performance of three algorithms:
  - BCDR (consistency).
  - CDRU-SC (strong controllability).
  - CDRU-DC (dynamic controllability).
Solution Utility and Conflicts Detected

**Utility Difference**

- BCDR over CDRU-SC
- CDRU-DC over CDRU-SC
- CDRU-SC (baseline)

**Conflicts Detected**

- BCDR Conflicts
- CDRU-SC Conflicts
- CDRU-DC Conflicts

Resolving Over-constrained Temporal Problems with Uncertain Durations
Contributions

• Over-constrained temporal problems can be resolved by relaxing the temporal constraints continuously.

• The fundamental concepts of conflicts and minimal relaxations naturally generalize to the continuous case.

• The framework naturally extends to resolving uncontrollable problems with uncertain durations.

• We can efficiently enumerate discrete and continuous relaxations in best-first order, by generalizing the Conflict-Directed A* algorithm.
Temporal Constraint Problems:
- Restore temporal consistency.
- Restore temporal controllability for uncertain durations.
- Resolve chance constrained problems with probabilistic durations.

VRP-TWs:
- Resolve over-constrained VRP-TWs through temporal and resource relaxations.

Temporal Planning Problems:
- Find semantically similar alternatives for goal and domain relaxations.
- Relax goals and domain specifications for resolving over-constrained planning problems.