Incrementally Building PPC Qualitative Constraint Networks

Michael Sioutis and Jean-François Condotta

Université Lille-Nord de France, Université d'Artois
CRIL-CNRS UMR 8188

September 13, 2014

Table of Contents

1 Introduction

2 Approach

3 Experimental Results

4 Conclusions

5 Future Work

What is Qualitative Reasoning?

Incrementally Building PPC Qualitative Constraint Networks

Michael Sioutis and Jean-François Condotta

Outline

Introduction

Approach

Experimental Results

Conclusions

Future Work

Acknowledge

Bibliography

- Qualitative Reasoning is an area of research within Artificial Intelligence that automates reasoning about continuous aspects of the physical world, such as *space* and *time*, by abstracting from numerical quantities

Applications of Qualitative Reasoning

- Qualitative reasoning is an important subproblem in many *applications*, such as:
  - Problem Solving
  - Planning
  - SpatioTemporal representation and reasoning

Reasons for Qualitative Reasoning

- Main reasons why non-precise, qualitative information may be useful:
  1. Only partial information may be available
  2. Constraints are often most naturally stated in qualitative terms
  3. Abstraction from numeric quantities boosts research and applications

The RCC-8 Constraint Language

- RCC-8 is a fragment of the Region Connection Calculus (RCC) [1]
- RCC-8 encodes binary topological relations between regions that are non-empty regular subsets of some topological space

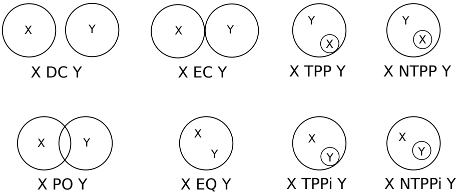


Figure: Two dimesional examples for the eight base relations of RCC-8

The IA Constraint Language

- Interval Algebra (IA) [2] encodes the possible binary relations between time intervals in a timeline

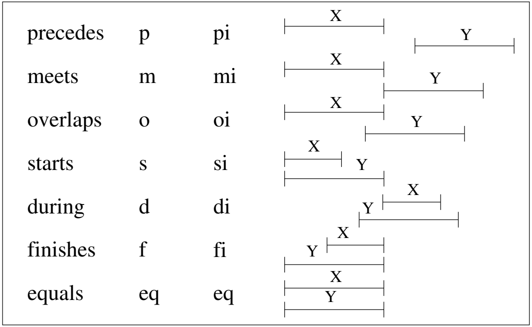


Figure: The thirteen base relations of IA

The RSAT Reasoning Problem

Incrementally Building PPC Qualitative Constraint Networks

Michael Sioutis and Jean-François Condotta

Outline

Introduction

Approach

Experimental Results

Conclusions

Future Work

Acknowledge

Bibliography

- RSAT is the reasoning problem of deciding whether a RCC-8 or IA network is satisfiable by a spatial or temporal configuration Θ respectively
- RSAT is NP-Complete [3, 4]
- However, tractable subclasses of RCC-8 and IA exist for which the consistency problem can be decided in polynomial time with a *path consistency* algorithm [3, 4]

The RSAT Reasoning Problem

- RSAT is the reasoning problem of deciding whether a RCC-8 or IA network is satisfiable by a spatial or temporal configuration Θ respectively
- RSAT is NP-Complete [3, 4]
- However, tractable subclasses of RCC-8 and IA exist for which the consistency problem can be decided in polynomial time with a *path consistency* algorithm [3, 4]

The RSAT Reasoning Problem

- RSAT is the reasoning problem of deciding whether a RCC-8 or IA network is satisfiable by a spatial or temporal configuration Θ respectively
- RSAT is NP-Complete [3, 4]
- However, tractable subclasses of RCC-8 and IA exist for which the consistency problem can be decided in polynomial time with a *path consistency* algorithm [3, 4]

Path Consistency

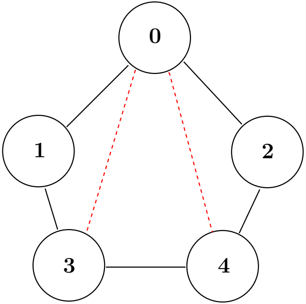
- Approximates consistency and realizes *forward checking* in a backtracking algorithm
- Checks the consistency of triples of relations and eliminates relations that are impossible though iteravely performing the operation

$$R_{ij} \leftarrow R_{ij} \cap R_{ik} \diamond R_{kj}$$

  until a fixed point $\overline{R}$ is reached
- If $R_{ij} = \emptyset$ for a pair (i, j) then R is inconsistent, otherwise $\overline{R}$ is *path consistent*.
- Computing $\overline{R}$ is upper bounded by $O(n^3)$ time

Chordal Graph

- A graph is *chordal* if each of its cycles of four or more nodes has a *chord*, which is an edge joining two nodes that are not adjacent in the cycle [8]
- An example of a chordal graph is shown below:

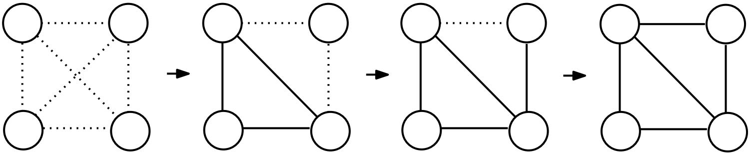


Partial Path Consistency [7]

- We triangulate the underlying graph of an input RCC-8 network in linear time
- Path consistency is then enforced on the chordal underlying graph of the input network
- Time complexity is upper bounded by $O(\delta \cdot |E|)$ for a chordal graph $G = (V, E)$ of the input network, where δ is the maximum degree of a vertex of G

State-of-the-Art Technique

- Partial Path consistency is enforced in an edge incremental manner [5]
- As noted, time complexity is $O(\delta \cdot |E|)$

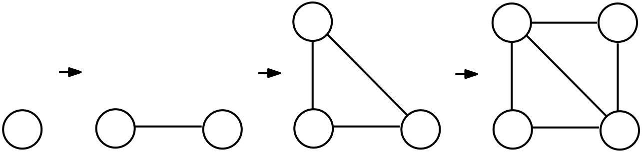


Motivation

- Huang showed that IA and RCC-8 have *canonical solutions* [6]
- Path consistent IA or RCC-8 networks with relations from some maximal tractable subset of their signatures can be extended arbitrarily with the addition of new temporal or spatial entities respectively
- We can construct a network vertex incrementally, and, thus, work with a small underlying graph at each step

Proposed Technique

- Partial Path consistency is enforced in a vertex incremental manner
- Time complexity is $O(\delta_2 \cdot |E_2| + \ldots + \delta_n \cdot |E_n|) \leq O(\delta \cdot |E|)$, as $\delta_2 \leq \ldots \leq \delta_n$ and $E_2 \cup \ldots \cup E_n = E$



iPPC+ Algorithm

Algorithm 1: iPPC+($\mathcal{N} \uplus \mathcal{N}'$, G, G')

in : A QCN $\mathcal{N} \uplus \mathcal{N}' = (V'', C'')$, and two chordal graphs $G = (V, E)$ and $G' = (V', E')$.

output : False if network $\mathcal{N} \uplus \mathcal{N}'$ results in a trivial inconsistency (contains the empty relation), True if the modified network $\mathcal{N} \uplus \mathcal{N}'$ is partially path consistent.

begin
- $Q \leftarrow \{(i,j) \mid (i,j) \in E'\}$;
- **while** $Q \neq \emptyset$ **do**
  - $(i,j) \leftarrow Q.pop()$;
  - **foreach** *k such that* $(i,k),(k,j) \in E \cup E'$ **do**
    - $t \leftarrow C''_{ik} \cap (C''_{ij} \diamond C''_{jk})$;
    - **if** $t \neq C''_{ik}$ **then**
      - **if** $t = \emptyset$ **then return** False;
      - $C''_{ik} \leftarrow t$; $C''_{ki} \leftarrow t^{-1}$;
      - $Q \leftarrow Q \cup \{(i,k)\}$;
    - $t \leftarrow C''_{kj} \cap (C''_{ki} \diamond C''_{ij})$;
    - **if** $t \neq C''_{kj}$ **then**
      - **if** $t = \emptyset$ **then return** False;
      - $C''_{kj} \leftarrow t$; $C''_{jk} \leftarrow t^{-1}$;
      - $Q \leftarrow Q \cup \{(k,j)\}$;
- **return** True;

Outline
Introduction
Approach
Experimental Results
Conclusions
Future Work
Acknowledge
Bibliography

GiPPC Algorithm

Algorithm 2: GiPPC($\mathcal{N}$, G)

in : A QCN $\mathcal{N} = (V, C)$, and a chordal graph $G = (V, E)$.
output : False if network $\mathcal{N}$ results in a trivial inconsistency, True if the modified network $\mathcal{N}$ is partially path consistent.

```
begin
    N_1 ⊎ N_2 ⊎ ... ⊎ N_i ← N; N' ← N_1;
    foreach k ← 2 to i do
        if ! iPPC+(N' ⊎ N_k, G', G_k) then return False;
        N' ← N' ⊎ N_k;
    N ← N';
    return True;
```

Advantages

- Smaller memory footprint
- Faster processing time
- Less consistency checks

Implementations Considered

Incrementally Building PPC Qualitative Constraint Networks

Michael Sioutis and Jean-François Condotta

Outline
Introduction
Approach
Experimental Results
Conclusions
Future Work
Acknowledge
Bibliography

- iPPC+ (Python code + vertex incremental partial path consistency)
- PPC (Python code + state-of-the-art edge incremental partial path consistency)

Datasets Considered

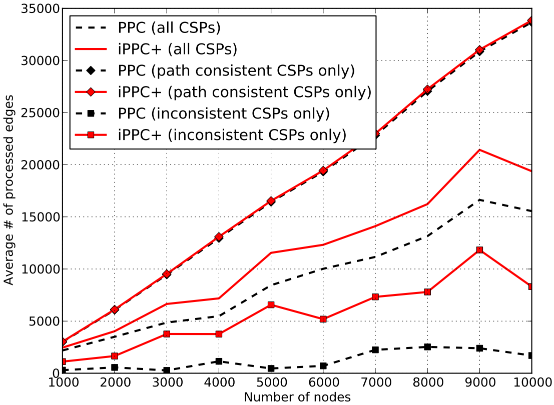
- Random IA and RCC-8 networks generated by the BA(n, m) model [7]
  - BA(n, m) creates random scale-free-like networks of size n and a preferential attachment value m [9]
- Random IA and RCC-8 networks using the A(n, d, l) model [3]
  - Model A(n, d, l) creates random networks of size n, degree d, and an average number l of relations per edge
- Real RCC-8 datasets that consist of `admingeo` [10] and `gadm-rdf`[1] comprising 11 761/77 907 nodes/edges and 276 728/590 865 nodes/edges respectively

[1] http://gadm.geovocab.org/

Experimental Evaluation (1/3)

- Performance comparison of iPPC+ and PPC for scale-free-like RCC-8 networks of the $BA(n, m)$ model

Comparison on Processed Edges



iPPC+ processes $\sim$ 27% more edges than PPC on average

Comparison on Consistency Checks

Incrementally Building PPC Qualitative Constraint Networks

Michael Sioutis and Jean-François Condotta

Outline

Introduction

Approach

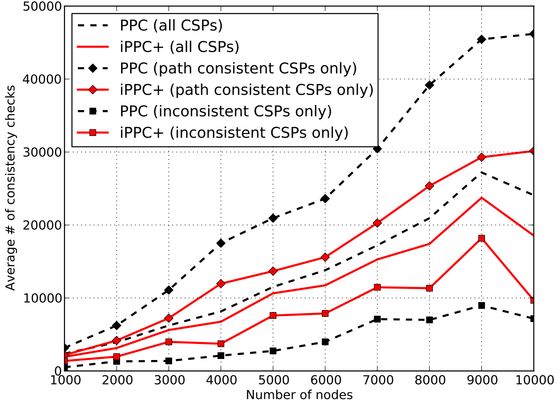
Experimental Results

Conclusions

Future Work

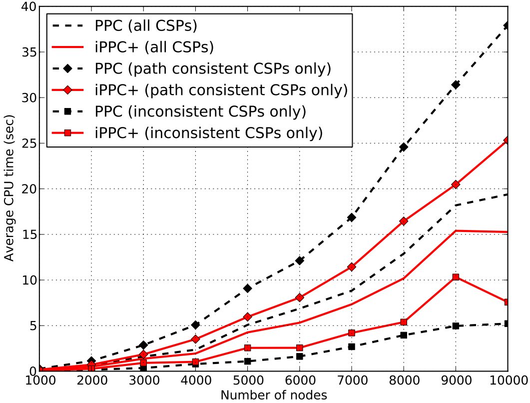
Acknowledge

Bibliography



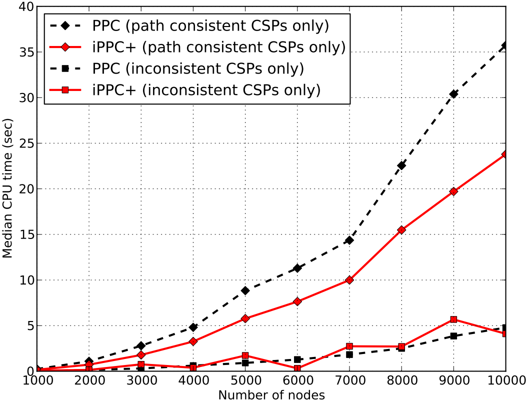
iPPC+ performs $\sim$ 15% less consistency checks than PPC on average

Comparison on Processing Time



iPPC+ runs $\sim$ 19% faster than PPC on average

Comparison on Median Processing Time



iPPC+ is sensitive regarding inconsistent networks

Performance Comparison for IA Networks

- iPPC+ runs $\sim$ 27% faster than PPC on average

Experimental Evaluation (2/3)

- Performance comparison of iPPC+ and PPC for random networks of the $A(n, d, l)$ model

Performance Comparison for RCC-8 and IA Networks

Incrementally Building PPC Qualitative Constraint Networks

Michael Sioutis and Jean-François Condotta

Outline

Introduction

Approach

Experimental Results

Conclusions

Future Work

Acknowledge

Bibliography

- iPPC+ runs $\sim$ 20% faster than PPC on average for RCC-8 networks
- iPPC+ runs $\sim$ 30% faster than PPC on average for IA networks

Experimental Evaluation (3/3)

- Performance comparison of iPPC+ and PPC for real RCC-8 datasets `admingeo` and `gadm-rdf`

Performance Comparison

- iPPC+ runs $\sim$ 23% faster than PPC for `admingeo`
- iPPC+ runs $\sim$ 42% faster than PPC for `gadm-rdf`

Test Setup

- All experiments were carried out on a computer with an Intel Core 2 Duo P7350 processor with a CPU frequency of 2.00 GHz, 4 GB RAM, and the Lucid Lynx x86_64 OS (Ubuntu Linux)
- iPC+, and PC$\triangledown$ were run with the CPython interpreter (`http://www.python.org/`), which implements Python 2
- Only one of the CPU cores was used for the experiments

Main Points

- We presented a novel algorithm that performs vertex incremental partial path consistency, viz., iPPC+
- We showed that iPPC+ performs better than state-of-the-art PPC

Main Directions

- Explore if we can also use vertex incrementality in a backtracking algorithm
- Make iPPC+ online by implementing a mechanism to incrementally maintain chordality [11]

Acknowledge

Incrementally Building PPC Qualitative Constraint Networks

Michael Sioutis and Jean-François Condotta

Outline

Introduction

Approach

Experimental Results

Conclusions

Future Work

Acknowledge

Bibliography

- This work was funded by a PhD grant from Université d'Artois and region Nord-Pas-de-Calais
- I would also like to thank Katia Papakonstantinopoulou and Panagiotis Liakos for proof-reading this presentation

References

[1] Randell, Cui, and Cohn: A Spatial Logic Based on Regions and Connection. In *KR*, 1996

[2] Allen: Maintaining knowledge about temporal intervals. *CACM*, vol. 26, 832–843, 1983

[3] Renz and Nebel: Efficient Methods for Qualitative Spatial Reasoning. *JAIR*, vol. 15, pages 289–318, 2001

[4] Nebel: Solving Hard Qualitative Temporal Reasoning Problems: Evaluating the Efficiency of Using the ORD-Horn Class. In *ECAI*, 1996

[5] Gerevini: Incremental qualitative temporal reasoning: Algorithms for the point algebra and the ord-horn class. *Artif. Intell.*, vol. 166, pages 37–80, 2005

[6] Huang: Compactness and its implications for qualitative spatial and temporal reasoning. In *KR*, 2012

[7] Sioutis and Koubarakis: Consistency of Chordal RCC-8 Networks. In *ICTAI*, 2012

[8] Golumbic: Algorithmic Graph Theory and Perfect Graphs. *Elsevier Science*, 2004

[9] Barabasi and Bonabeau: Scale-Free Networks. *Scientific American*, pages 50–59, 2003

[10] Goodwin, Dolbear, and Hart: Geographical Linked Data: The Administrative Geography of Great Britain on the Semantic Web. *TGIS*, vol. 12, 19–30, 2008

[11] Berry, Heggernes, and Villanger: A vertex incremental approach for maintaining chordality. *Discrete Mathematics*, vol. 306, 318–336, 2006

The End

Any Questions?