Improving DPOP with Branch Consistency for Solving Distributed Constraint Optimization Problems

Ferdinando Fioretto [1,2] Tiep Le [1] William Yeoh [1]
Enrico Pontelli [1] Tran Cao Son [1]

[1]Dept. Computer Science, New Mexico State University

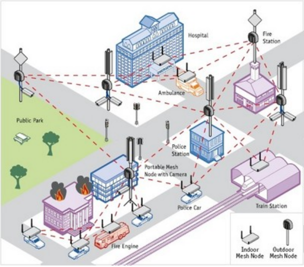
[2]Dept. Mathematics and Computer Science, University of Udine

Sept. 9, 2014

Talk Outline

Distributed Optimization: Motivations

- Some problems cannot be realistically addressed in a centralized fashion.
- Agents cooperate to achieve a common objective.
- Simultaneously they can purse private goals.
- Agents are constrained by limited communication capabilities.



Source: http://kenanaonline.com/users/antennamaker

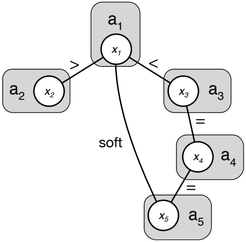
Distributed Optimization: Motivations

- Some problems cannot be realistically addressed in a centralized fashion.
- Agents cooperate to achieve a common objective.
- Simultaneously they can purse private goals.
- Agents are constrained by limited communication capabilities.
- Solving time is important!

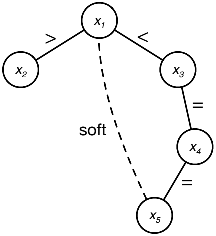
Distributed Constrained Optimization (DCOP)

A *DCOP* is defined by a tuple $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{F}\ \alpha \rangle$, where:

- $\mathcal{A}$ is a set of *agents*;
- $\mathcal{X}$ is a set of *variables*.
- $\mathcal{D}$ is a set of finite *domains*.
- $\mathcal{F}$ is a set of *utility functions*, $f_i : \times_{x_j \in scope(f_i)} D_j \mapsto \mathbb{N} \cup \{0, -\infty\}$.
- $\alpha : \mathcal{X} \rightarrow \mathcal{A}$ maps each variable to one agent.



Constraint Graph



Pseudo-tree

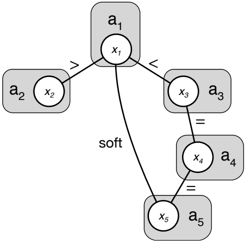
x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
	...	
3	3	2

Soft Constraint Table

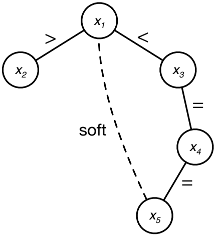
Distributed Constrained Optimization (DCOP)

A *DCOP* is defined by a tuple $\langle \mathcal{A}, \mathcal{X}, \mathcal{D}, \mathcal{F}\ \alpha \rangle$, where:

- $\mathcal{A}$ is a set of *agents*;
- $\mathcal{X}$ is a set of *variables*.
- $\mathcal{D}$ is a set of finite *domains*.
- $\mathcal{F}$ is a set of *utility functions*, $f_i : \times_{x_j \in scope(f_i)} D_j \mapsto \mathbb{N} \cup \{0, -\infty\}$.
- $\alpha : \mathcal{X} \rightarrow \mathcal{A}$ maps each variable to one agent.



Constraint Graph



Pseudo-tree

x_1	x_3	Utilities
0	0	$-\infty$
0	1	0
0	2	0
0	3	0
	…	
3	3	$-\infty$

Hard Constraint Table

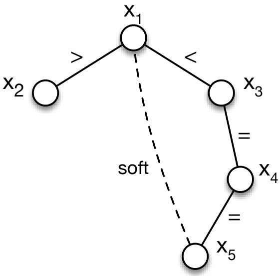
Solving DCOP

- Find an utility maximal assignment for all the variables of the problem.
- Agents communicate exchanging messages.
- This is often the bottleneck!

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

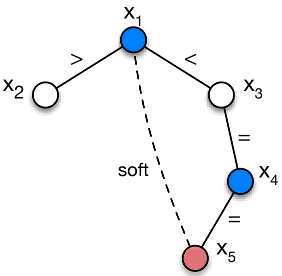
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
. . .		. . .		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
. . .		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

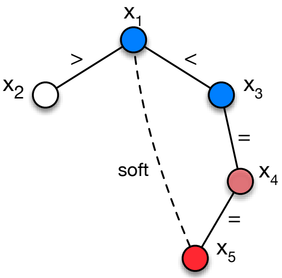
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
. . .		. . .		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
. . .		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

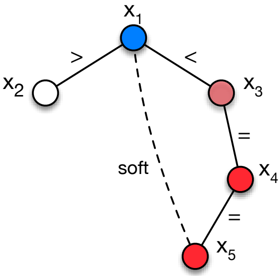
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

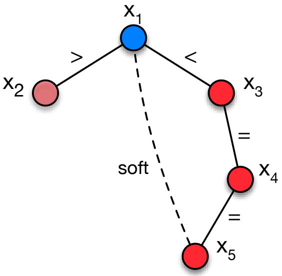
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

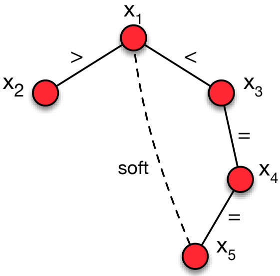
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

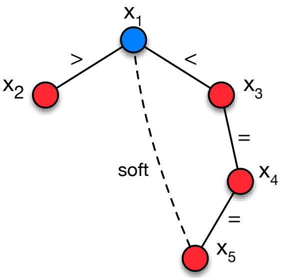
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

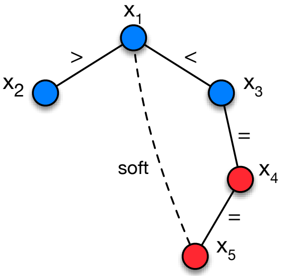
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

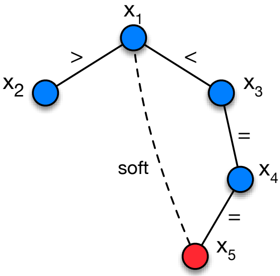
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

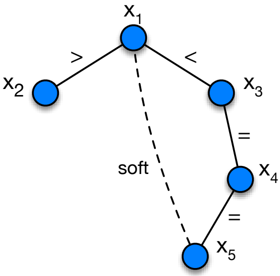
x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Solving DCOP (cont.)

Distributed Pseudo-Tree Optimization Procedure (DPOP)

1. Pseudo-Tree Construction Phase.
2. UTIL propagation phase.
3. VALUE propagation phase.



UTIL Phase Computations of a_5 (x_5): UTIL Table

x_1	x_4	Utilities		
0	0	max(20+0, 8-∞, 10-∞, 3-∞)	=	20
0	1	max(20-∞, 8+0, 10-∞, 3-∞)	=	8
0	2	max(20-∞, 8-∞, 10+0, 3-∞)	=	10
0	3	max(20-∞, 8-∞, 10-∞, 3+0)	=	3
...		...		

x_1	x_5	Utilities
0	0	20
0	1	8
0	2	10
0	3	3
...		

Distributed Optimization: Current Limitations

1. The DCOP community has not focused on exploiting hard constraints.[1]
2. Number of messages exchanged vs Message Size.

[1] With few exceptions that do not take advantage of CP technologies.

Distributed Optimization: Current Limitations

1. The DCOP community has not focused on exploiting hard constraints.[1]
2. Number of messages exchanged vs Message Size.
3. Can we improve on these limitations and achieve better results?

[1] With few exceptions that do not take advantage of CP technologies.

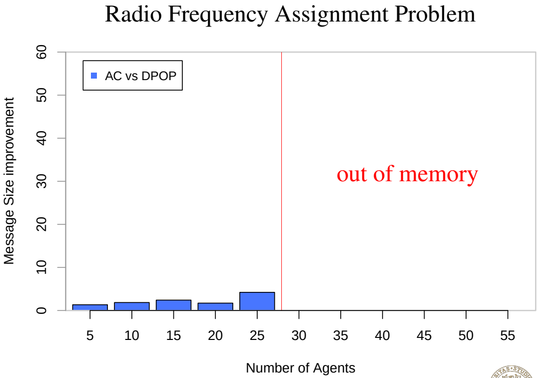
Exploiting Hard Constraints

- In the context of the DPOP algorithm.
- Assume that we are given a DCOP with hard constraints.
- First Trial: Integrating Arc Consistency to prune the UTIL table.

Exploiting Hard Constraints

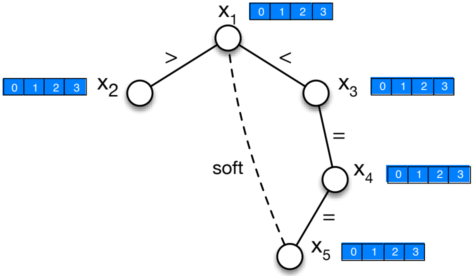
- In the context of the DPOP algorithm.
- Assume that we are given a DCOP with hard constraints.
- First Trial: Integrating Arc Consistency to prune the UTIL table.

- Given a set of links.
- Assign them a frequency.
- The inference at the receivers: $|x_i - x_j| > s$.
- Use as few frequencies as possible.



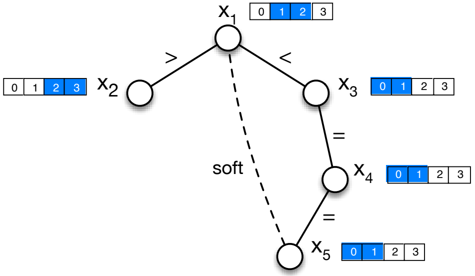
Exploiting Hard Constraints

- In the context of the DPOP algorithm.
- Assume that we are given a DCOP with hard constraints.
- First Trial: Integrating Arc Consistency to prune the UTIL table.



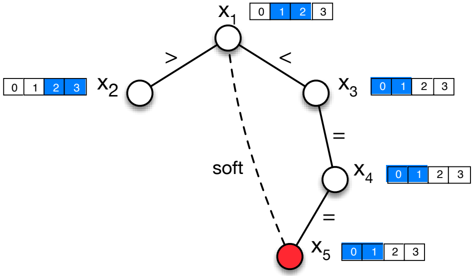
Exploiting Hard Constraints

- In the context of the DPOP algorithm.
- Assume that we are given a DCOP with hard constraints.
- First Trial: Integrating Arc Consistency to prune the UTIL table.



Exploiting Hard Constraints

- In the context of the DPOP algorithm.
- Assume that we are given a DCOP with hard constraints.
- First Trial: Integrating Arc Consistency to prune the UTIL table.

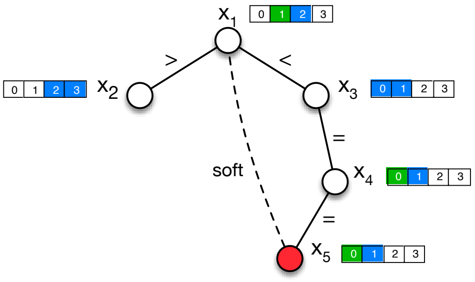


$UTIL_5$

x_5	x_4	x_1
0	0	1
0	0	2
0	1	1
0	1	2
1	0	1
1	0	2
1	1	1
1	1	2

Exploiting Hard Constraints

- In the context of the DPOP algorithm.
- Assume that we are given a DCOP with hard constraints.
- First Trial: Integrating Arc Consistency to prune the UTIL table.

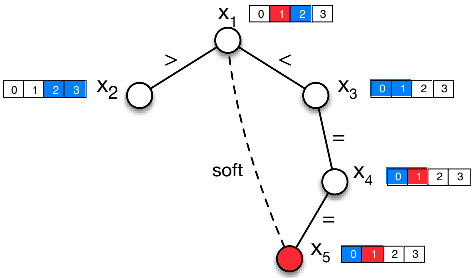


UTIL$_5$

x_5	x_4	x_1
0	0	1
0	0	2
0	1	1
0	1	2
1	0	1
1	0	2
1	1	1
1	1	2

Exploiting Hard Constraints

- In the context of the DPOP algorithm.
- Assume that we are given a DCOP with hard constraints.
- First Trial: Integrating Arc Consistency to prune the UTIL table.



$UTIL_5$

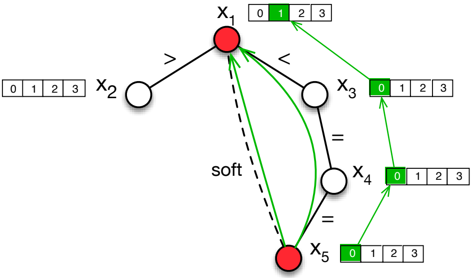
x_5	x_4	x_1
0	0	1
0	0	2
0	1	1
0	1	2
1	0	1
1	0	2
1	1	1
1	1	2

Branch Consistency (BrC)

- Obs. 1: The domain store transmits a limited amount of information. It accounts for no interaction among variables.
- Obs. 2: A Pseudo-Tree branch contains the relevant information to build the UTIL tables.

Branch Consistency (BrC) (cont.)

Def. Branch Consistency for pair of values:



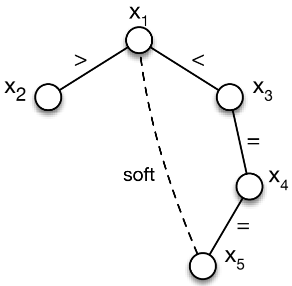
Def. A DCOP is Branch Consistent (BrC) iff for any pair of variables (x_i, x_j) with x_i and x_j in the same branch, and any $(u, v) \in f_{ij}$, (u, v) is branch consistent.

Def. The Value Reachability Matrix (VRM) M_{ij} of x_i and x_j, with x_i ancestor of x_j, is a binary matrix of size $|D_i| \times |D_j|$, where $M_{ij}[r, c]{=}1$ iff (r, c) is BrC.

BrC-DPOP

BrC-DPOP consists of 5 phases:

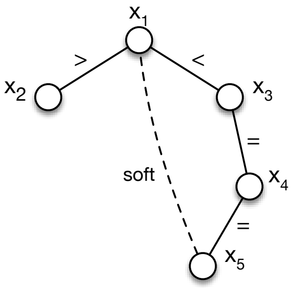
1. Pseudo-tree Generation Phase.



BrC-DPOP

BrC-DPOP consists of 5 phases:

2. Path Construction Phase.



$x_1 < x_2$

$$\begin{array}{c|c} & x_2 \quad 0 \; 1 \; 2 \; 3 \\ x_1 \begin{array}{c}0\\1\\2\\3\end{array} & \begin{bmatrix} 0&1&1&1\\ 0&0&1&1\\ 0&0&0&1\\ 0&0&0&0 \end{bmatrix} \end{array}$$

$x_1 > x_3$

$$\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0&0&0&0\\ 1&0&0&0\\ 1&1&0&0\\ 1&1&1&0 \end{bmatrix} \end{array}$$

$x_3 = x_4$

$$\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix} \end{array}$$

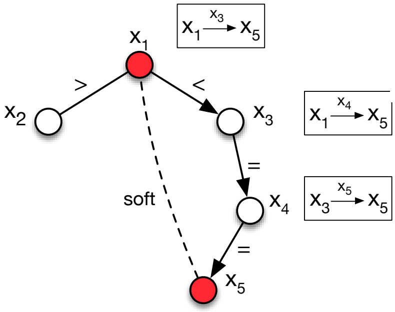
$x_4 = x_5$

$$\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix} \end{array}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

2. Path Construction Phase.



$x_1 < x_2$

$$\begin{array}{c|c} & x_2 \; 0\;1\;2\;3 \\ \begin{array}{c} x_1 \; 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_1 > x_3$

$$\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

$x_3 = x_4$

$$\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

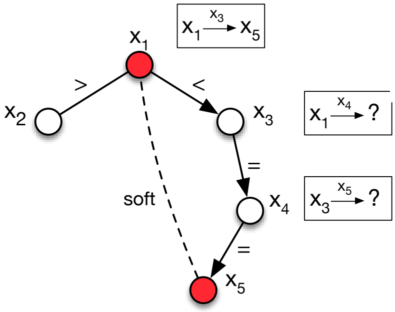
$x_4 = x_5$

$$\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

2. Path Construction Phase.



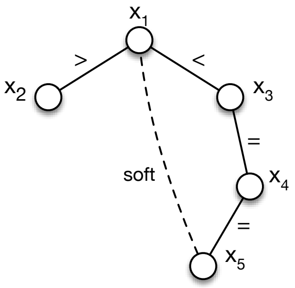
$$\underline{x_1 < x_2} \qquad \underline{x_1 > x_3} \qquad \underline{x_3 = x_4} \qquad \underline{x_4 = x_5}$$

$$\begin{array}{cc} & x_2 \ \ 0\ 1\ 2\ 3 \\ x_1 \begin{array}{c} 0\\1\\2\\3 \end{array} & \begin{bmatrix} 0&1&1&1\\0&0&1&1\\0&0&0&1\\0&0&0&0 \end{bmatrix} \end{array} \quad \begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0&0&0&0\\1&0&0&0\\1&1&0&0\\1&1&1&0 \end{bmatrix} \end{array} \quad \begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1 \end{bmatrix} \end{array} \quad \begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1 \end{bmatrix} \end{array}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

3. Arc Consistency Enforcement Phase.



$x_1 < x_2$:
$$\begin{array}{c|c} & x_2 \;\; 0 \;\; 1 \;\; 2 \;\; 3 \\ \hline x_1 \; \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_1 > x_3$:
$$\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

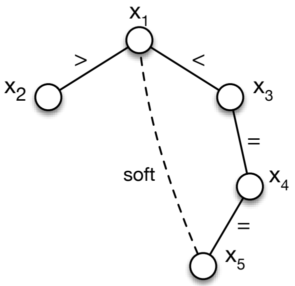
$x_3 = x_4$:
$$\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

$x_4 = x_5$:
$$\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

3. Arc Consistency Enforcement Phase.



$x_1 < x_2$

$$\begin{array}{cc} & x_2 \\ x_1 & \begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ \hline 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 2 & 0 & 0 & 0 & 1 \\ 3 & 0 & 0 & 0 & 0 \end{array} \end{array}$$

$x_1 > x_3$

$$\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_3 = x_4$

$$\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

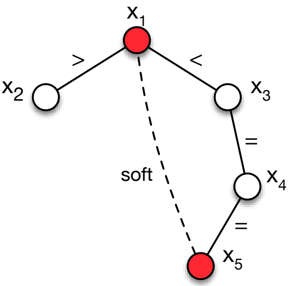
$x_4 = x_5$

$$\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

4. Branch Consistency Enforcement Phase.



$x_1 < x_2$

$$\begin{array}{cc} & x_2 \;\; 0 \;\; 1 \;\; 2 \;\; 3 \\ \begin{array}{c} x_1 \; 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_1 > x_3$

$$\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_3 = x_4$

$$\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

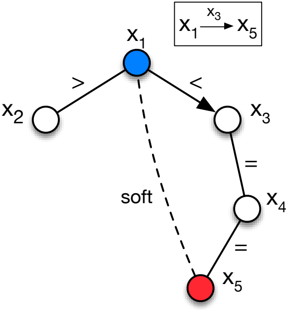
$x_4 = x_5$

$$\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

4. Branch Consistency Enforcement Phase.



$x_1 < x_2$

$$\begin{array}{c|c} & x_2 \; 0 \; 1 \; 2 \; 3 \\ \begin{array}{c} x_1 \; 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_1 > x_3$

$$\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_3 = x_4$

$$\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_4 = x_5$

$$\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

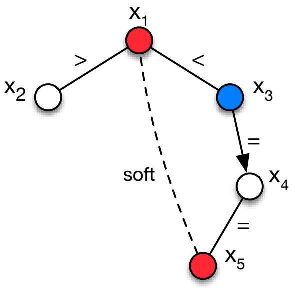
M_{11}

$$\begin{array}{cc} & x_1 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

4. Branch Consistency Enforcement Phase.



$$x_1 \xrightarrow{x_4} ?$$

$$x_1 < x_2: \quad \begin{array}{cc} & x_2 \;\; 0 \; 1 \; 2 \; 3 \\ x_1 \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \qquad x_1 > x_3: \quad \begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

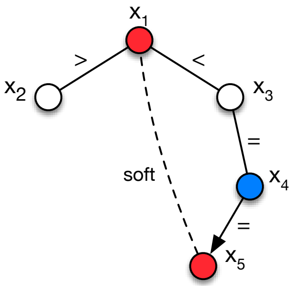
$$x_3 = x_4: \quad \begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \qquad x_4 = x_5: \quad \begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$\underset{M_{31}}{\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}} \times \underset{M_{11}}{\begin{array}{cc} & x_1 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}} = \underset{M_{31}}{\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

4. Branch Consistency Enforcement Phase.



$x_1 < x_2$: $\begin{array}{c|c} & x_2 \;\; 0\;1\;2\;3 \\ \hline x_1 \begin{array}{c}0\\1\\2\\3\end{array} & \begin{bmatrix}0&0&0&0\\0&0&1&1\\0&0&0&1\\0&0&0&0\end{bmatrix}\end{array}$

$x_1 > x_3$: $\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix}0&0&0&0\\1&0&0&0\\1&1&0&0\\0&0&0&0\end{bmatrix}\end{array}$

$x_3 = x_4$: $\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\end{array}$

$x_4 = x_5$: $\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\end{array}$

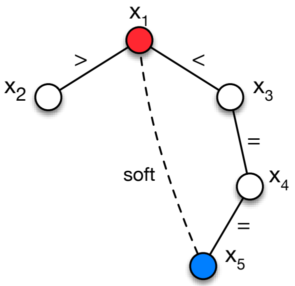
$\boxed{x_3 \xrightarrow{x_5} ?}$

$$\underset{M_{43}}{\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}\end{array}} \times \underset{M_{31}}{\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix}0&0&0&0\\1&0&0&0\\1&1&0&0\\0&0&0&0\end{bmatrix}\end{array}} = \underset{M_{41}}{\begin{array}{cc} & x_4 \\ x_1 & \begin{bmatrix}0&0&0&0\\1&0&0&0\\1&1&0&0\\0&0&0&0\end{bmatrix}\end{array}}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

4. Branch Consistency Enforcement Phase.



$x_1 < x_2$:
$$\begin{array}{cc} & x_2 \\ x_1 & \begin{array}{cccc} 0 & 1 & 2 & 3 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_1 > x_3$:
$$\begin{array}{cc} & x_3 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$x_3 = x_4$:
$$\begin{array}{cc} & x_4 \\ x_3 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

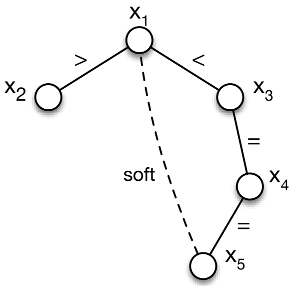
$x_4 = x_5$:
$$\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$\underset{M_{54}}{\begin{array}{cc} & x_5 \\ x_4 & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}} \times \underset{M_{41}}{\begin{array}{cc} & x_4 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}} = \underset{M_{51}}{\begin{array}{cc} & x_5 \\ x_1 & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{array}}$$

BrC-DPOP

BrC-DPOP consists of 5 phases:

5. UTIL and VALUE Phases.



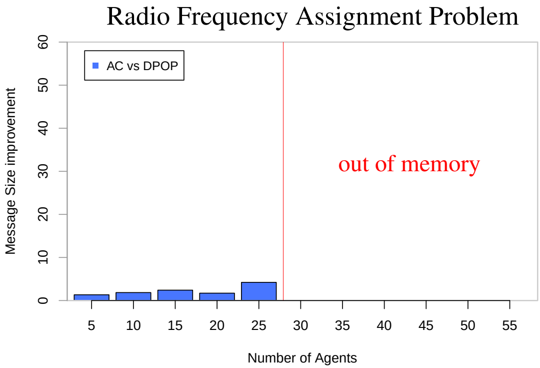
UTIL_5

x_5	x_4	x_1
0	0	1
0	0	2
1	1	2

BrC-DPOP

BrC-DPOP consists of 5 phases:

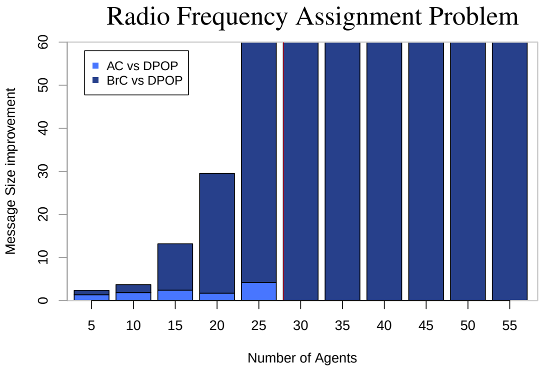
5. UTIL and VALUE Phases.



BrC-DPOP

BrC-DPOP consists of 5 phases:

5. UTIL and VALUE Phases.



BrC-DPOP: Theoretical Results

Let $n = |\mathcal{A}|$, $e = |\mathcal{F}|$ and $d = \max_{x_i \in \mathcal{X}} |D_i|$.

Theorem 1. The AC propagation phase requires $O(nde)$ messages, each of size $O(d)$.

Theorem 2. The BrC propagation phase requires $O(e)$ messages, each of size $O(d^2)$.

Theorem 3. The DCOP is arc consistent after the AC propagation phase.

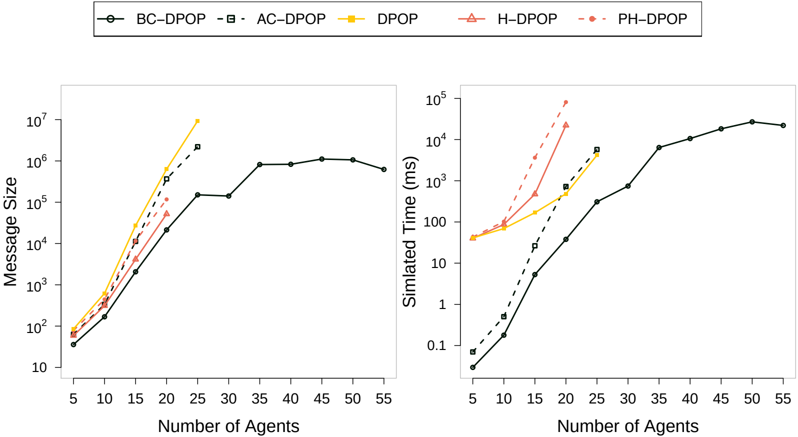
Theorem 4. The DCOP is branch consistent after the BrC propagation phase.

Theorem 5. BrC-DPOP is complete and correct.

Experiments: Algorithms

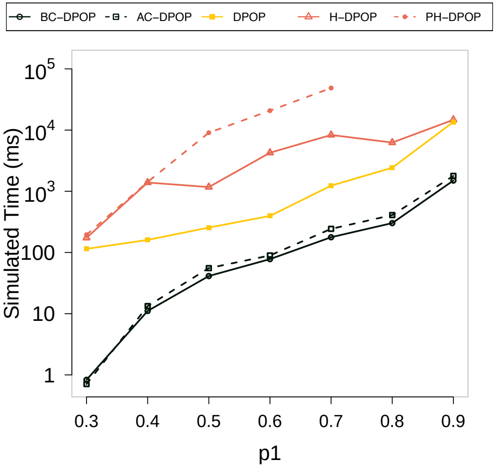
- DPOP
- H-DPOP
- P(rivacy-enhanced) H-DPOP
- AC-DPOP (only AC Propagation phase)
- BrC-DPOP

Experiments: Radio Link Frequency Assignment



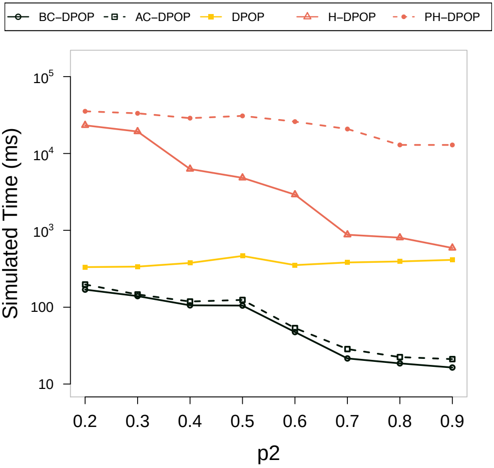
Experiments: Random Graphs (Varying p1)

- $|\mathcal{A}| = 10.$
- $|\mathcal{X}| = 10.$
- $|D_i| = 8, \forall x_i \in \mathcal{X}.$
- $p2 = 0.6$
- We randomly injected hard constraints of type $<$ or $\neq$.



Experiments: Random Graphs (Varying p2)

- $|\mathcal{A}| = 10.$
- $|\mathcal{X}| = 10.$
- $|D_i| = 8, \forall x_i \in \mathcal{X}.$
- $p1 = 0.6$
- We randomly injected hard constraints of type $<$ or $\neq$.



Conclusions and Future Works

- Message size is one of the most important bottleneck in solving DCOPs.
- We have introduced the concept of *Branch Consistency* for Pseudo-Trees.
- We have introduced BrC-DPOP.
- Enhanced performances and scalability on random graphs and RLFA problems.
- We plan to extend this approach by:
  - Exploring propagation of soft constraints.
  - Handling high arity constraints.
- We are also interested in studying memory bounded solutions to scale up even to larger problems.

Conclusions and Future Works

- Message size is one of the most important bottleneck in solving DCOPs.
- We have introduced the concept of *Branch Consistency* for Pseudo-Trees.
- We have introduced BrC-DPOP.
- Enhanced performances and scalability on random graphs and RLFA problems.
- We plan to extend this approach by:
  - Exploring propagation of soft constraints.
  - Handling high arity constraints.
- We are also interested in studying memory bounded solutions to scale up even to larger problems.

Thank You!