ER-DCOPS: A FRAMEWORK FOR DCOP WITH UNCERTAINTY IN CONSTRAINT UTILITIES

Tiep Le, Ferdinando Fioretto, William Yeoh,
Tran Cao Son, Enrico Pontelli

Computer Science Department
New Mexico State University

OUTLINE

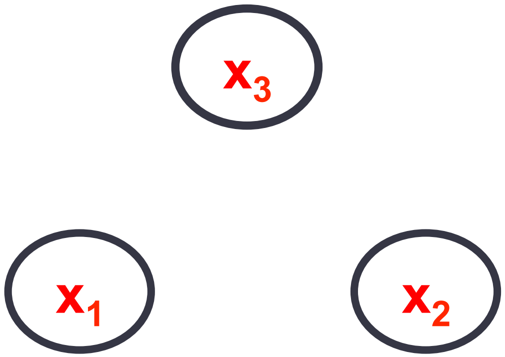
- BACKGROUND & MOTIVATION
- ER-DCOP
- ER-DPOP ALGORITHM
- EXPERIMENTAL RESULTS
- CONCLUSION

OUTLINE

- **BACKGROUND & MOTIVATION**
- ER-DCOP
- ER-DPOP ALGORITHM
- EXPERIMENTAL RESULTS
- CONCLUSION

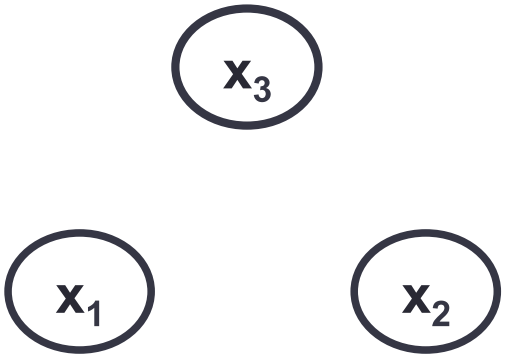
DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

- DCOP P = <X, D, F, A, α>



DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

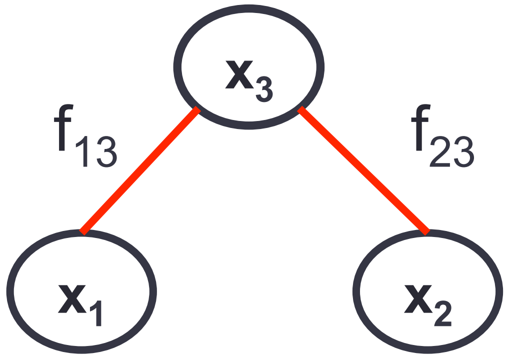
- DCOP P = <X, D, F, A, α>



$D_1 = D_2 = \{0\}$
$D_3 = \{0, 1\}$

DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

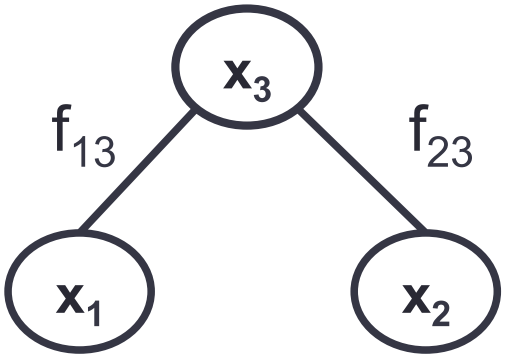
- DCOP P = <X, D, F, A, α>



$D_1 = D_2 = \{0\}$
$D_3 = \{0, 1\}$

DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

- DCOP P = <X, D, F, A, α>



f_{13}

x_1	x_3	U_{13}
0	0	50
0	1	30

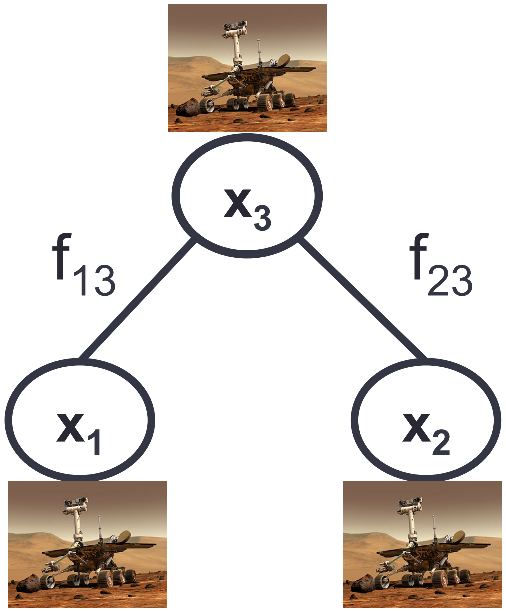
f_{23}

x_2	x_3	U_{23}
0	0	40
0	1	50

$D_1 = D_2 = \{0\}$
$D_3 = \{0, 1\}$

DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

- DCOP P = <X, D, F, A, α>



f_{13}

x_1	x_3	U_{13}
0	0	50
0	1	30

f_{23}

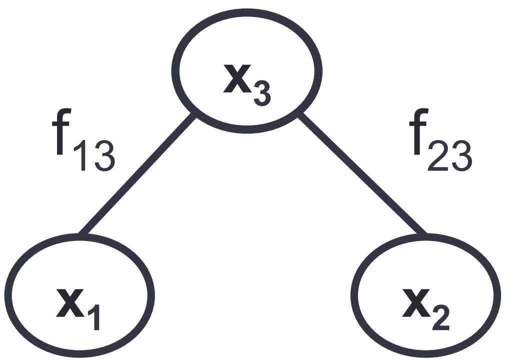
x_2	x_3	U_{23}
0	0	40
0	1	50

$D_1 = D_2 = \{0\}$
$D_3 = \{0, 1\}$

Worker x_1 owns variable x_1
Worker x_2 owns variable x_2
Assistant robot x_3 owns variable x_3

DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

- DCOP P = <X, D, F, A, α>
- Goal: The assignment for all variables maximizes the aggregate utility



f_{13}

x_1	x_3	U_{13}
0	0	50
0	1	30

f_{23}

x_2	x_3	U_{23}
0	0	40
0	1	50

$D_1 = D_2 = \{0\}$
$D_3 = \{0, 1\}$

Worker x_1 owns variable x_1
Worker x_2 owns variable x_2
Assistant robot x_3 owns variable x_3

MOTIVATION

- In real-world applications, the utilities are stochastic.

f_{23}

x_2	x_3		U_{23}
0	0	(Fail)	0
		(Success)	40
0	1	(Fail)	0
		(Success)	50

UR-DCOP

- In real-world applications, the utilities are stochastic.

f_{23}

x_2	x_3		U_{23}	Good	Bad
0	0	(Fail)	0	50%	90%
		(Success)	40	50%	10%
0	1	(Fail)	0	50%	90%
		(Success)	50	50%	10%

- Stochastic utilities can be sampled from a known probability distribution space.

MOTIVATION

- In real-world applications, the utilities are stochastic.

f_{23}

x_2	x_3		U_{23}	Good	Bad
0	0	(Fail)	0	50%	90%
		(Success)	40	50%	10%
0	1	(Fail)	0	20%	50%
		(Success)	50	80%	50%

- Stochastic utilities can be sampled from a known probability distribution space.

- Expected-regret

MOTIVATION

- In real-world applications, the utilities are stochastic.

f_{23}

x_2	x_3		U_{23}	Good	Bad
0	0	(Fail)	0	50%	90%
		(Success)	40	50%	10%
0	1	(Fail)	0	20%	50%
		(Success)	50	80%	50%

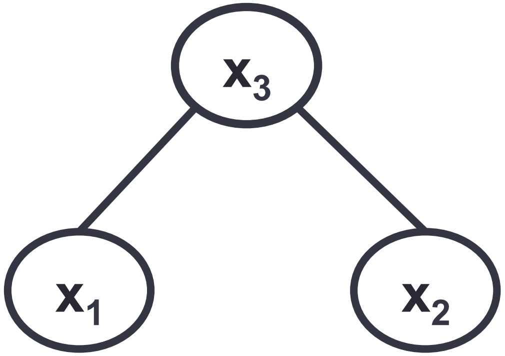
- Stochastic utilities can be sampled from a known probability distribution space.

ER-DCOP framework!

OUTLINE

- BACKGROUND
- **ER-DCOP**
- ER-DPOP ALGORITHM
- EXPERIMENTAL RESULTS
- CONCLUSION

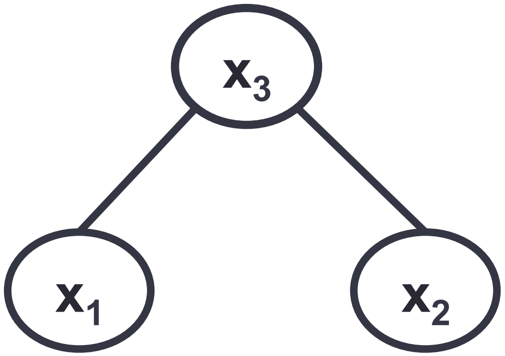
EXPECTED REGRET-DCOP (ER-DCOP)



x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

EXPECTED REGRET-DCOP (ER-DCOP)

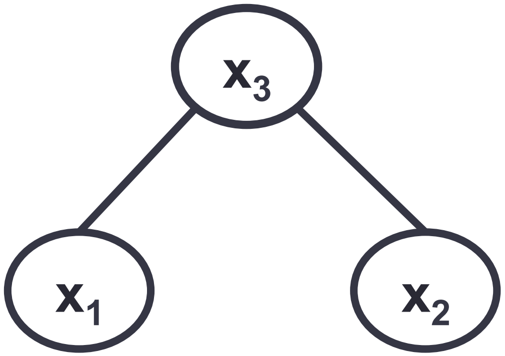


x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_2	x_3		U_{23}
0	0	(Fail)	0
		(Success)	40
0	1	(Fail)	0
		(Success)	50

EXPECTED REGRET-DCOP (ER-DCOP)

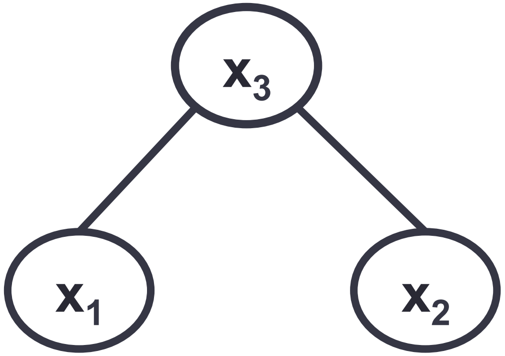


x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_2	x_3	r_2	U_{23}
0	0	0 (Fail)	0
		1 (Success)	40
0	1	0 (Fail)	0
		1 (Success)	50

EXPECTED REGRET-DCOP (ER-DCOP)



x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

Good: 12%

Bad: 88%

x_2	x_3	r_2	U_{23}	Good	Bad
0	0	0 (Fail)	0	50%	90%
		1 (Success)	40	50%	10%
0	**1**	**0 (Fail)**	**0**	**20%**	**50%**
		1 (Success)	**50**	**80%**	**50%**

EXPECTED REGRET-DCOP (ER-DCOP)

- ER-DCOP P = <X, D, A, α, R, S, F>

x_1	x_3	r_1	U_{13}	Good	Bad
0	0	0 (Fail)	0	10%	30%
		1 (Success)	50	90%	70%
0	1	0 (Fail)	0	30%	50%
		1 (Success)	30	70%	50%

x_2	x_3	r_2	U_{23}	Good	Bad
0	0	0 (Fail)	0	50%	90%
		1 (Success)	40	50%	10%
0	1	0 (Fail)	0	20%	50%
		1 (Success)	50	80%	50%

EXPECTED REGRET-DCOP (ER-DCOP)

- ER-DCOP P = <X, D, A, α, R, S, F>
- belief of r_1, belief of r_2
- x : joint belief for all random variables

x_1	x_3	r_1	U_{13}	Good
0	0	0 (Fail)	0	10%
		1 (Success)	50	90%
0	1	0 (Fail)	0	30%
		1 (Success)	30	70%

x_2	x_3	r_2	U_{23}	Good
0	0	0 (Fail)	0	50%
		1 (Success)	40	50%
0	1	0 (Fail)	0	20%
		1 (Success)	50	80%

EXPECTED REGRET-DCOP (ER-DCOP)

- ER-DCOP P = <X, D, A, α, R, S, F>
- Using Expected Utility (EU)

consider only 1 joint belief of good weather

x_1	x_3	r_1	U_{13}	Good	EU
0	0	0 (Fail)	0	10%	45
		1 (Success)	50	90%	
0	1	0 (Fail)	0	30%	21
		1 (Success)	30	70%	

x_2	x_3	r_2	U_{23}	Good	EU
0	0	0 (Fail)	0	50%	20
		1 (Success)	40	50%	
0	1	0 (Fail)	0	20%	40
		1 (Success)	50	80%	

EXPECTED REGRET-DCOP (ER-DCOP)

bad weather

x_1	x_2	x_3	EU
0	0	0	39
0	0	1	40

← Optimal assignment if bad weather (EU = 40)

good weather

x_1	x_2	x_3	EU
0	0	0	65
0	0	1	41

← Optimal assignment if good weather (EU = 65)

Belief Space
Good: 12%   Bad: 88%

EXPECTED REGRET-DCOP (ER-DCOP)

bad weather

x_1	x_2	x_3	EU	**Regret**
0	0	0	39	40-39=1
0	0	1	40	40-40=0

good weather

x_1	x_2	x_3	EU	**Regret**
0	0	0	65	65-65=0
0	0	1	41	65-61=4

Assignment $x_1 = x_2 = x_3 = 0$ has
regret of 1 if bad weather
regret of 0 if good weather

Belief Space
Good: 12% Bad: 88%

EXPECTED REGRET-DCOP (ER-DCOP)

bad weather

x_1	x_2	x_3	EU	Regret
0	0	0	39	40-39=1
0	0	1	40	40-40=0

88%

good weather

x_1	x_2	x_3	EU	Regret
0	0	0	65	65-65=0
0	0	1	41	65-61=4

12%

Expected-Regret (ER)

x_1	x_2	x_3	ER
0	0	0	12%*0 + 88%*1 = 0.88
0	0	1	12%*4 + 88%*0 = 0.48

Belief Space
Good: 12%   Bad: 88%

EXPECTED REGRET-DCOP (ER-DCOP)

bad weather

x_1	x_3	EU	x_2	x_3	EU	Regret
0	0	35	0	0	4	40-39=1
0	1	15	0	1	25	40-40=0

88%

good weather

x_1	x_3	EU	x_2	x_3	EU	Regret
0	0	45	0	0	20	65-65=0
0	1	21	0	1	40	65-61=4

12%

Expected-Regret (ER)

x_1	x_2	x_3	ER
0	0	0	12%*0 + 88%*1 = 0.88
0	**0**	**1**	**12%*4 + 88%*0 = 0.48**

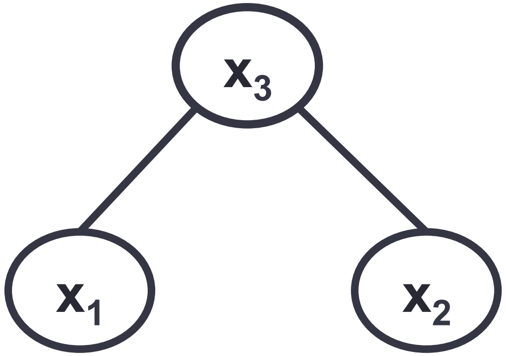
The solution minimizes the expected-regret

OUTLINE

- BACKGROUND
- ER-DCOP
- **ER-DPOP ALGORITHM**
- EXPERIMENTAL RESULTS
- CONCLUSION

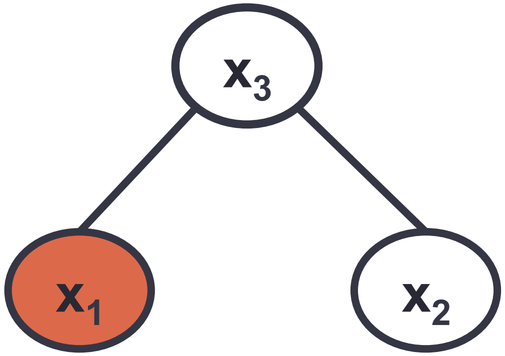
ER-DPOP

- Phase 1: Generation of the pseudo-tree



ER-DPOP

- Phase 2: Resolution of subproblems



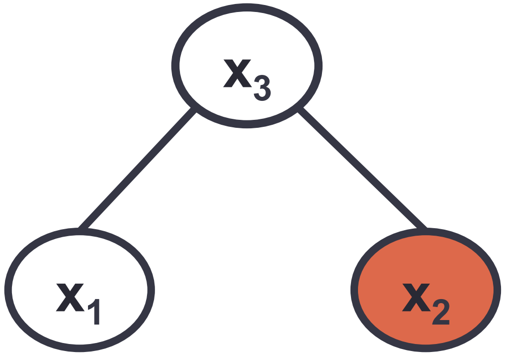
x_1	x_3	r_1	U_{13}	Good	Bad
0	0	0 (Fail)	0	10%	30%
		1 (Success)	50	90%	70%
0	1	0 (Fail)	0	30%	50%
		1 (Success)	30	70%	50%

x_3	EU(Good)	EU(Bad)
0	45	35
1	21	15

EU = Expected Utility

ER-DPOP

- Phase 2: Resolution of subproblems



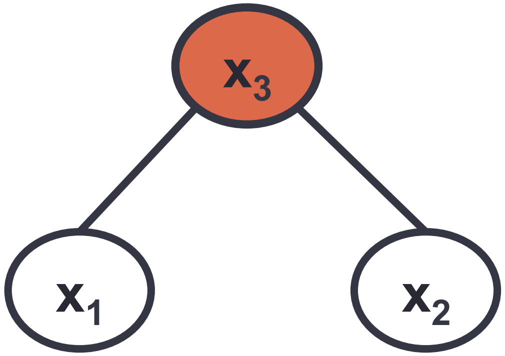
x_2	x_3	r_2	U_{23}	Good	Bad
0	0	0 (Fail)	0	50%	90%
		1 (Success)	40	50%	10%
0	1	0 (Fail)	0	20%	50%
		1 (Success)	50	80%	50%

x_3	EU(Good)	EU(Bad)
0	20	4
1	40	25

EU = Expected Utility

ER-DPOP

- Phase 2: Resolution of subproblems



x_3	EU(Good)	EU(Bad)
0	45+20=65	35+4=39
1	21+40=61	15+25=40

EU = Expected Utility

ER-DPOP

- Phase 3: Resolution of the main problems
  - Generate DCOP with expected-regret as utilities
  - Use DPOP [Petcu et al. AAAI2007] to solve that DCOP

x_1	x_3	EU(Good)	EU(Bad)
0	0	45	35
0	1	21	15

x_2	x_3	EU(Good)	EU(Bad)
0	0	20	4
0	1	40	25

x_1	x_3	Expected-Regret
0	0	12%*(45-45) + 88%*(15-35) = -17.6
0	1	12%*(45-21) + 88%*(15-15) = 2.88

x_2	x_3	Expected-Regret
0	0	12%*(20-20) + 88%*(25-4) = 18.48
0	1	12%*(20-40) + 88%*(25-25) = 2.4

ER-DPOP IMPLEMENTATIONS

- GPU-ER-DPOP (GPU-based ER-DPOP)
  - Utilizes the parallelism offered by Graphical Processing Unit (GPU) to speed up computations in ER-DPOP
- ASP-ER-DPOP (ASP-based ER-DPOP)
  - Prunes the search space offered by logic-programming based inference rules in Answer Set Programming (ASP)

RELATED WORK

- UR-DCOP (F. Wu et al. AAAI 2014)
  - Beliefs of random variables are independent with values of decision variables;
  - Belief space does not exhibit probabilistic model;
  - Minimizing the worst-case loss (regret) over belief space.

OUTLINE

- BACKGROUND
- ER-DCOP
- ER-DPOP ALGORITHM
- **EXPERIMENTAL RESULTS**
- CONCLUSION

EXPERIMENTAL RESULTS

- Algorithms:
  - GPU-ER-DPOP
  - ASP-ER-DPOP
  - FRODO-ER (solve subproblems in Phase 2 sequentially)

- Domains:
  - Random Graph (varying |X|, |D|, constraint density p_1, constraint tightness p_2, or belief space's size)
  - Power Network Problems (varying Topology or |D|)

EXPERIMENTAL RESULTS

|X|	ASP-ER-DPOP	GPU-ER-DPOP	FRODO-ER
8	3.1	0.1	0.3
13	9.4	0.2	61.1
18	44.1	N/A	N/A
23	120.8	N/A	N/A

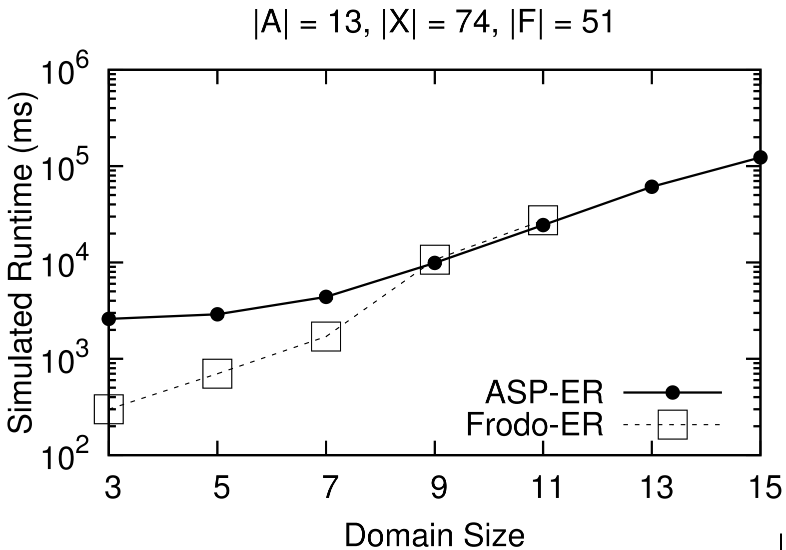
runtime in second

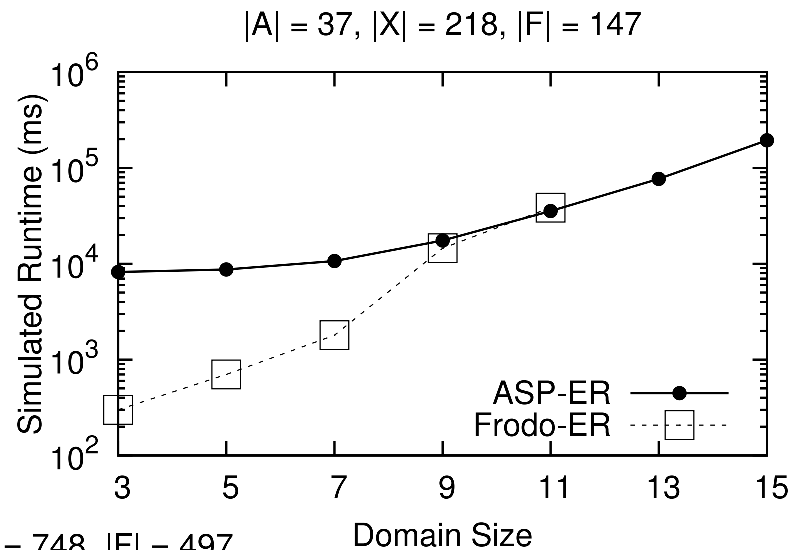
N/A: not available

|D|	ASP-ER-DPOP	GPU-ER-DPOP	FRODO-ER
4	4.5	0.1	1.8
6	8.9	0.1	33.6
8	22.2	1.2	143.2
10	80.4	4.8	N/A
12	121.2	15.4	N/A

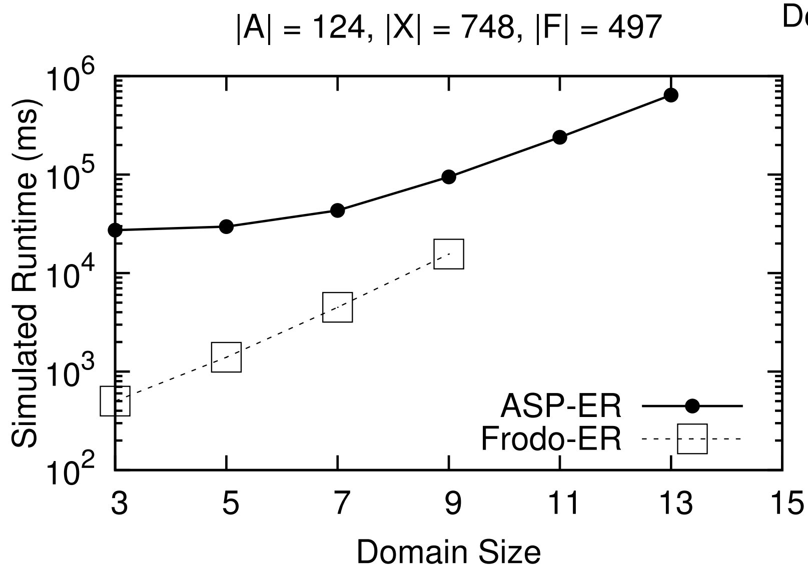
Random Graphs

EXPERIMENTAL RESULTS





Power Network Problems



EXPERIMENTAL RESULTS

- Compare the actual regret between
  - ER-DCOP
  - UR-DCOP (F. Wu et al. AAAI 2014)
    - Beliefs of random variables are independent with values of decision variables;
    - Belief space does not exhibit probabilistic model;
    - Minimizing the worst-case loss (regret) over belief space.
- Domain:
  - Random Graph
  - UR-DCOPs instances augmented a probability for each joint belief according to a normal distribution.

EXPERIMENTAL RESULTS

Belief Space's Size	Better	Worse	Equal
5	45%	20%	35%
10	36%	28%	36%
15	47%	20%	33%

Compare Actual Regret ER-DCOP solution vs UR-DCOP solution

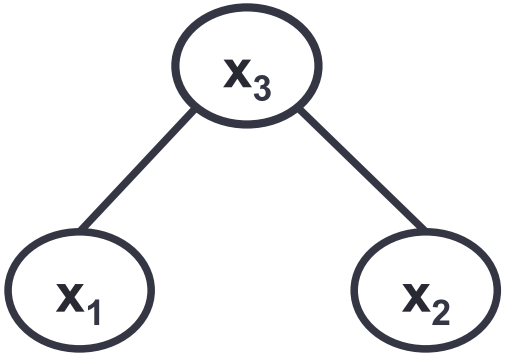
CONCLUSION

- ER-DCOPs to model DCOPs with uncertainty in constraint utilities.
- ER-DPOP, a distributed complete algorithm to solve ER-DCOPs.
- GPU-ER-DPOP harnesses the parallelism offered by GPU.
- ASP-ER-DPOP exploits logic programming-based inference rules to prune the search space.
- ER-DCOP solution outperforms UR-DCOP solution in terms of actual regret (belief space exhibits normal distribution).

THANK YOU FOR YOUR ATTENTION!

DPOP

- 3 phases: Pseudo-tree Generation, UTIL Propagation, and VALUE Propagation.

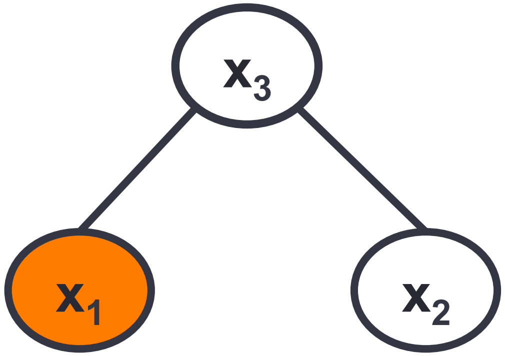


x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

DPOP

- 3 phases: Pseudo-tree Generation, UTIL Propagation, and VALUE Propagation.



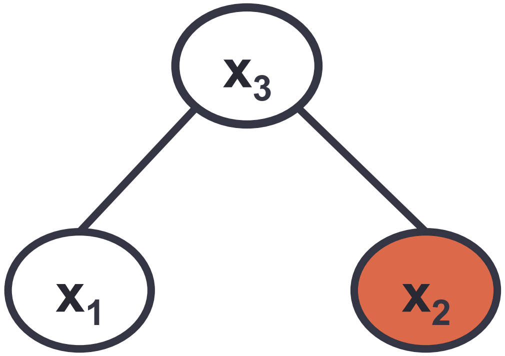
x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_3	U_{max}
0	50
1	30

DPOP

- 3 phases: Pseudo-tree Generation, UTIL Propagation, and VALUE Propagation.



x_1	x_3	U_{13}
0	0	50
0	1	30

x_3	U_{max}
0	50
1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_3	U_{max}
0	40
1	50

ANSWER SET PROGRAMMING (ASP)

- Π = { rule | rule's form : $c \leftarrow a_1, \ldots, a_m, \text{not } b_1, \ldots, \text{not } b_n$ }
- The answer sets of an ASP program which encodes a problem P represent solutions for P.

BENEFITS: RULE vs TABLE

$D_{x1} = D_{x2} = [0,1]$.
$U(X_1,X_2) = X_1 + X_2$

x_1	x_3	U_{13}
0	0	0
0	1	1
1	0	1
1	1	2

$domain\_x_1(0..1).$
$domain\_x_2(0..1).$
$utility_{1\_2}(U,X_1,X_2) \leftarrow domain\_x_1(X_1),$
$domain\_x_2(X_2),\ U = X_1 + X_{2.}$

Implicit Representation

BENEFITS: OPTIMIZED ASP SOLVER (GROUNDING)

$D_{x1} = D_{x2} = [0,1]$.
The message $U(X_1,X) = 0$ if $X_1 = X_2 = 0$; otherwise, $-\infty$

x_1	x_2	U_{12}
0	0	0
0	1	$-\infty$
1	0	$-\infty$
1	1	$-\infty$

domain_x$_1$(0..1).
domain_x$_2$(0..1).
utility$_{1_2}$(0,X_1,X_2) ← domain_x$_1$(X_1),
domain_x$_2$(X_2), $X_1 = 0$, $X_2 = 0$.

EXPECTED REGRET-DCOP (ER-DCOP)

- ER-DCOP P = <X, D, A, α, R, S, F>
- The conditional probability distribution of a random variable is a belief of the random variable.

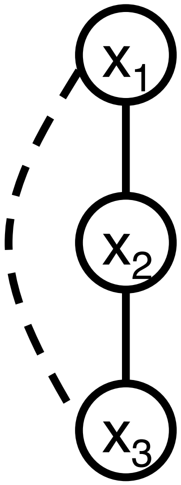
x_1	x_3	r_1	U_{13}	**Good**
0	0	0 (Fail)	0	10%
		1 (Success)	50	90%
0	1	0 (Fail)	0	30%
		1 (Success)	30	70%

x_2	x_3	r_2	U_{23}	**Good**
0	0	0 (Fail)	0	50%
		1 (Success)	40	50%
0	1	0 (Fail)	0	20%
		1 (Success)	50	80%

Good: 12% Bad: 88%

GPU-ER-DPOP

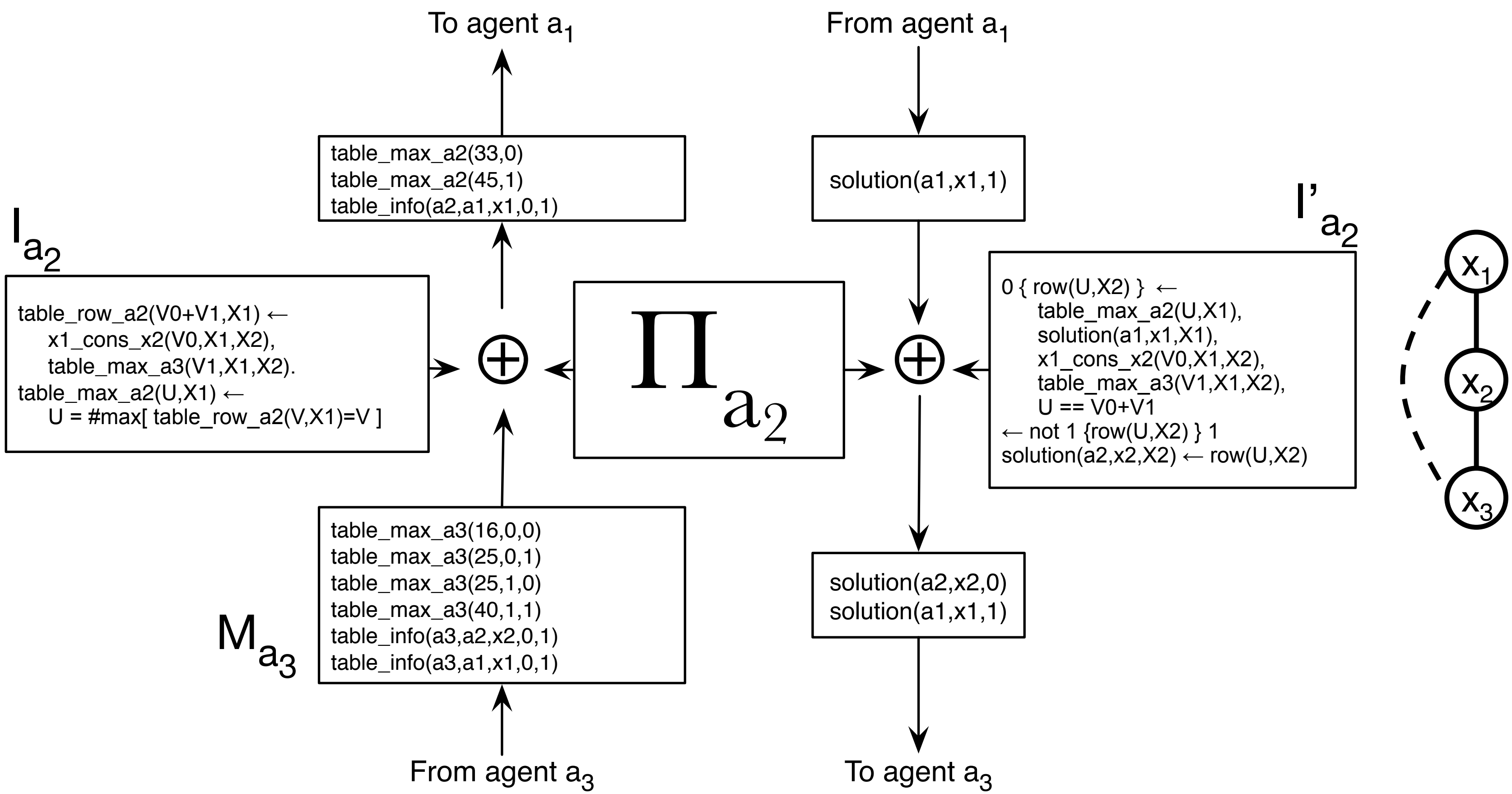
- Specifying a DCOP using ASP
  - var(x_3). dom(x_3, 0..1).
  - constraint(u_{1_3}). scope(u_{1_3},x_1,x_3). util$_{1_3}$(5,0,0). (facts or rules)
  - agent(a_3). owner(a_3,x_3).



- 3 phases as DPOP.

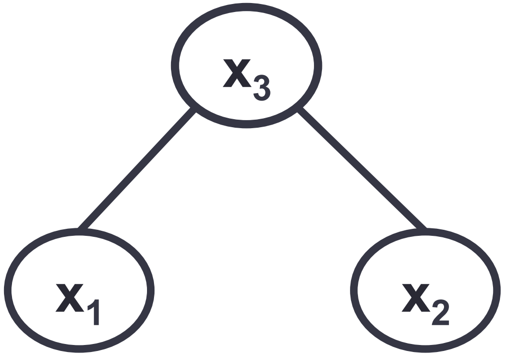
- Information about children, ancestor.
  - In x_2:
    - ancestor(x_1).
    - children(x_3).

ASP-DPOP



Agent Controller in Agent 2

EXPECTED REGRET-DCOP (ER-DCOP)

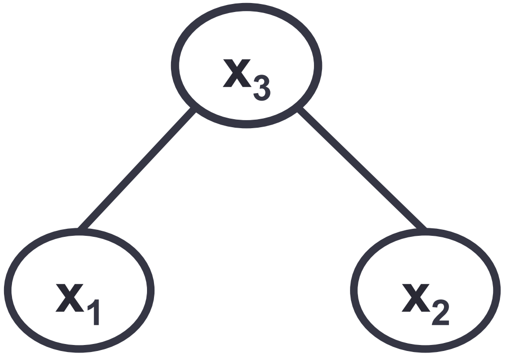


x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_2	x_3	r_2	U_{23}	Good
0	0	0 (Fail)	0	50%
		1 (Success)	40	50%
0	1	0 (Fail)	0	20%
		1 (Success)	50	80%

EXPECTED REGRET-DCOP (ER-DCOP)



x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

Good: 12%

Bad: 88%

x_2	x_3	r_2	U_{23}	Good	Bad
0	0	0 (Fail)	0	50%	90%
		1 (Success)	40	50%	10%
0	1	0 (Fail)	0	20%	50%
		1 (Success)	50	80%	50%

EXPECTED REGRET-DCOP (ER-DCOP)

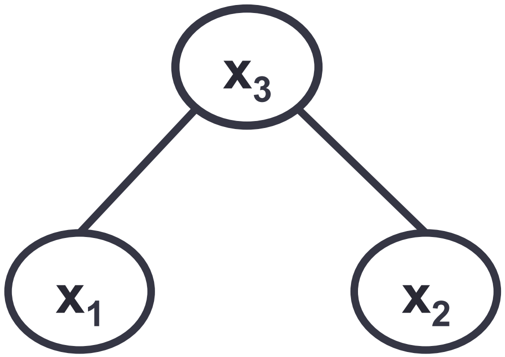
- ER-DCOP P = <X, D, A, α, R, S, F>
- EU = Expected Utility

x_1	x_3	r_1	U_{13}	Good	EU
0	0	0 (Fail)	0	10%	45
		1 (Success)	50	90%	
0	1	0 (Fail)	0	30%	21
		1 (Success)	30	70%	

x_2	x_3	r_2	U_{23}	Good	EU
0	0	0 (Fail)	0	50%	20
		1 (Success)	40	50%	
0	1	0 (Fail)	0	20%	40
		1 (Success)	50	80%	

DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

- DCOP P = <X, D, F, A, α>
- Goal: The assignment for all variables maximizes the aggregate utility.



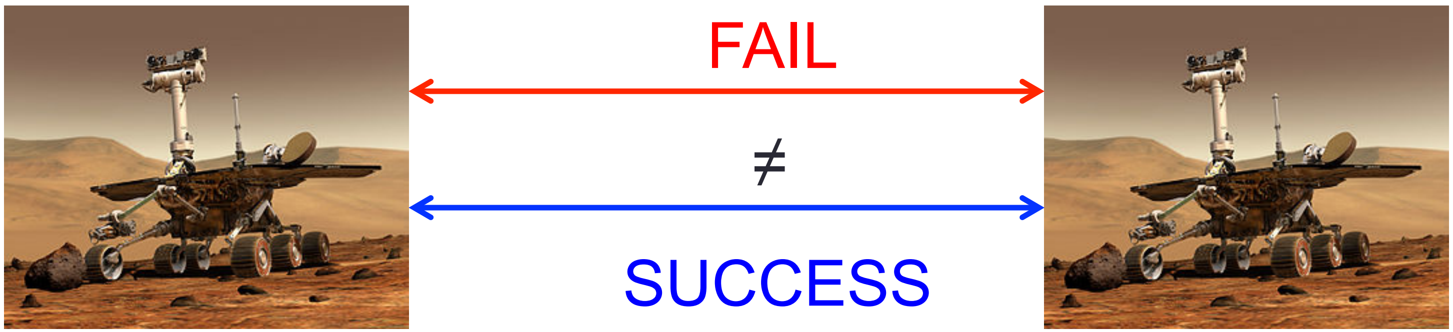
x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

$D_1 = D_2 = \{0\}$
$D_3 = \{0, 1\}$

MOTIVATION

- In real-world applications, the utilities are stochastic.

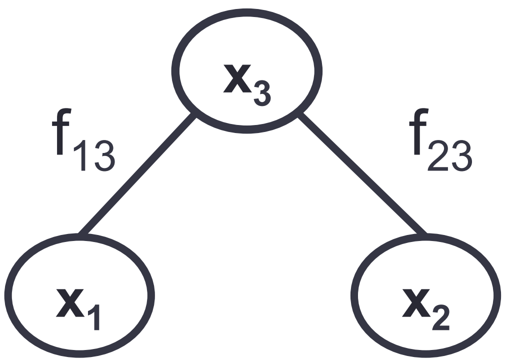


- Stochastic utilities can be sampled from a known probability distribution space.

ER-DCOP framework

DISTRIBUTED CONSTRAINT OPTIMZATION PROBLEMS

- DCOP P = <X, D, F, A, α>



f_{13}

x_1	x_3	U_{13}
0	0	50
0	1	30

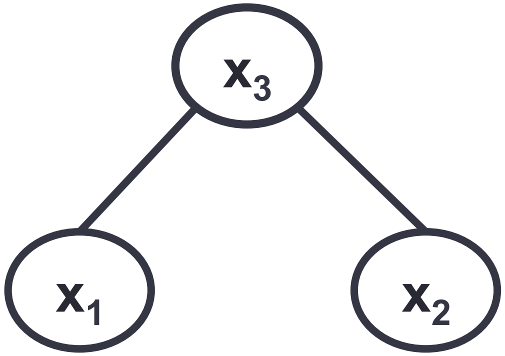
f_{23}

x_2	x_3	U_{23}
0	0	40
0	1	50

$D_1 = D_2 = \{0\}$
$D_3 = \{0, 1\}$

DPOP[1]

- 3 phases: **Pseudo-tree Generation**, UTIL Propagation, and VALUE Propagation.



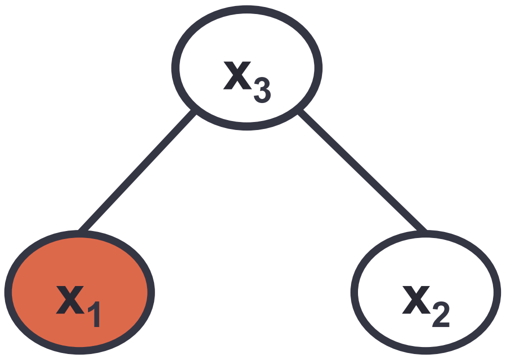
x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

1: (A. Petcu et al. IJCAI 2005)

DPOP

- 3 phases: Pseudo-tree Generation, **UTIL Propagation**, and VALUE Propagation.



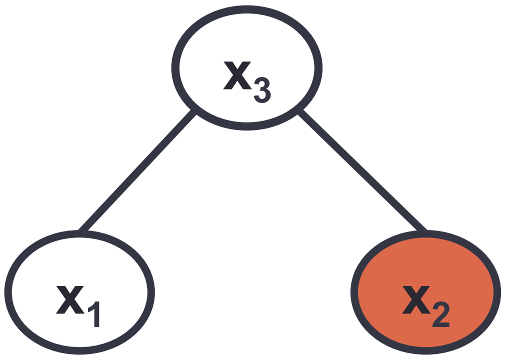
x_1	x_3	U_{13}
0	0	50
0	1	30

x_3	U_{max}
0	50
1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

DPOP

- 3 phases: Pseudo-tree Generation, **UTIL Propagation**, and VALUE Propagation.



x_1	x_3	U_{13}
0	0	50
0	1	30

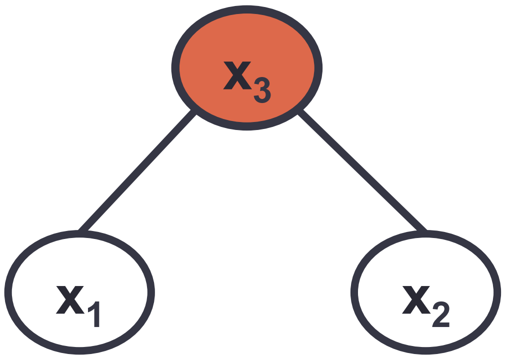
x_3	U_{max}
0	50
1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_3	U_{max}
0	40
1	50

DPOP

- 3 phases: Pseudo-tree Generation, **UTIL Propagation**, and VALUE Propagation.



x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_3	U_{max}
0	50
1	30

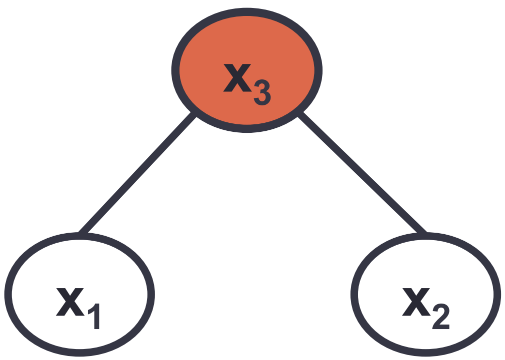
+

x_3	U_{max}
0	40
1	50

x_3	U_{max}
0	50+40=90
1	30+50=80

DPOP

- 3 phases: Pseudo-tree Generation, UTIL Propagation, and **VALUE Propagation**.



x_1	x_3	U_{13}
0	0	50
0	1	30

x_2	x_3	U_{23}
0	0	40
0	1	50

x_3	U_{max}
0	50
1	30

\+

x_3	U_{max}
0	40
1	50

x_3	U_{max}
0	90
1	80