Improving the Performance of Consistency Algorithms by Localizing and Bolstering Propagation in a Tree Decomposition

Shant Karakashian, Robert Woodward & Berthe Y. Choueiry

Constraint Systems Laboratory
University of Nebraska-Lincoln

Acknowledgments:
- Experiments conducted at UNL's Holland Computing Center
- NSF Award RI-111795

Outline

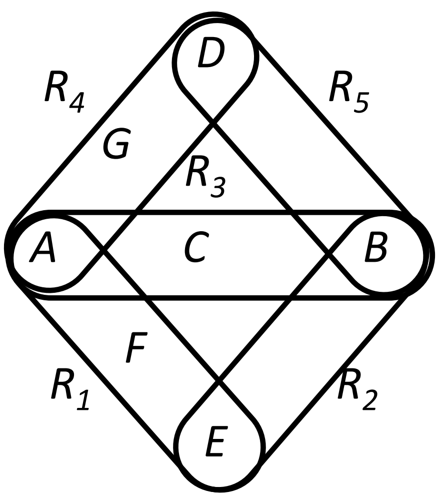
- Introduction
- Background
  - Tree decomposition
  - Relational consistency property R(∗,m)C
- Key ideas
  - Localize consistency to clusters of a tree decomposition
  - Bolstering propagation at separators
- Evaluation
  - Theoretical: Comparing resulting consistency properties
  - Empirical: Solving CSPs in a backtrack-free manner
- Conclusions & Future Work

Introduction

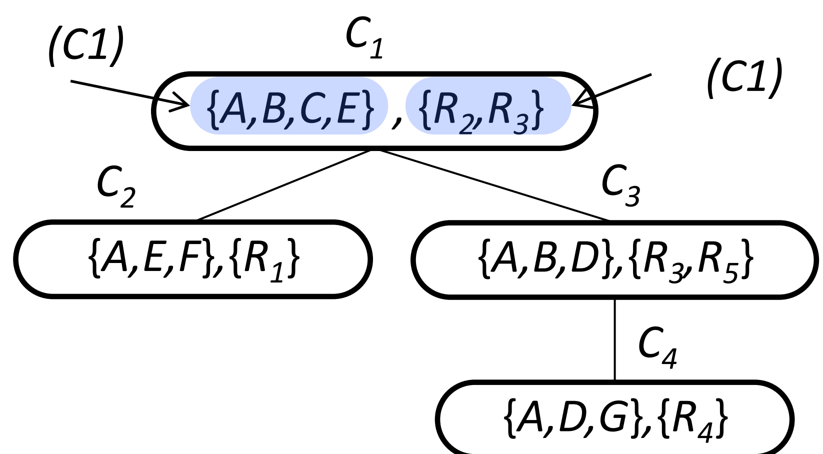
- Constraint Satisfaction Problems (CSPs)
  - NP-complete in general
  - Islands of tractability are classes of CSPs solvable in polynomial time
- One tractability condition links [Freuder 82]
  - Consistency level to
  - Width of the constraint network, a structural parameter
- Our approach: exploit a tree decomposition
  - Localize application of the consistency algorithm
  - Add redundant constraints at separators to enhance propagation
  - Practical tractability aims to solve CSP instances in a backtrack-free manner

Tree Decomposition

- A tree decomposition:$\langle \mathcal{T}, \quad , \quad \rangle$
  - $\mathcal{T}$: a tree of clusters
  - : maps variables to clusters
  - : maps constraints to clusters
- Conditions
  - Each constraint appears in at least one cluster with all the variables in its scope
  - For every variable, the clusters where the variable appears induce a connected subtree



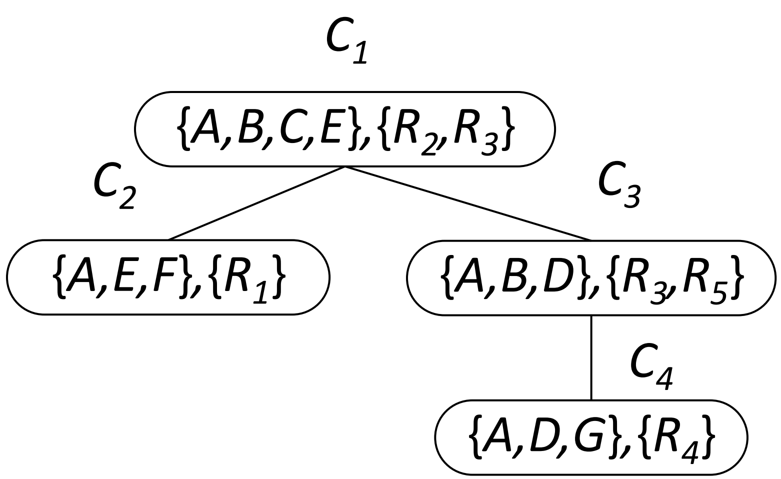
Hypergraph

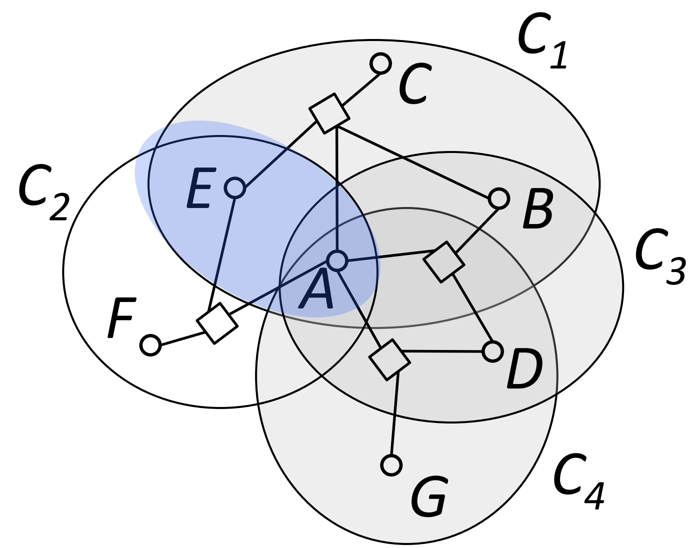


Tree decomposition

Tree Decomposition: Separators

- A separator of two adjacent clusters is the set of variables associated to both clusters



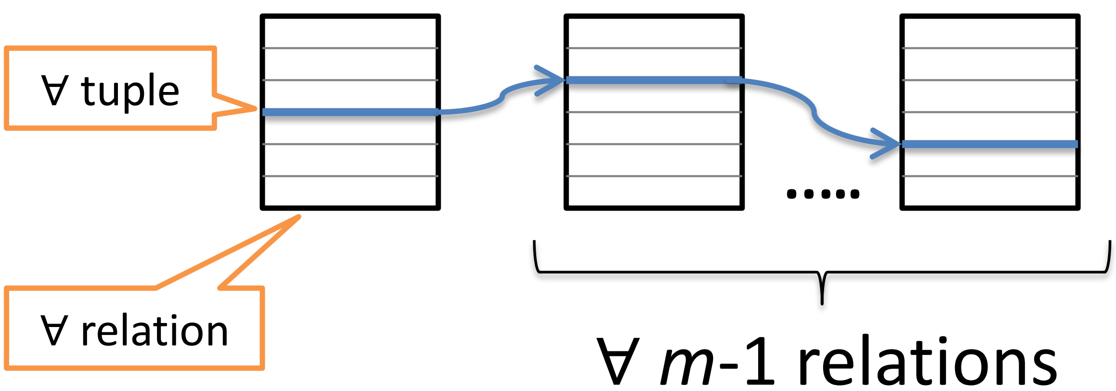


- Width of a decomposition/network
  - Treewidth = maximum number of variables in clusters - 1

Relational Consistency Property R(∗,m)C

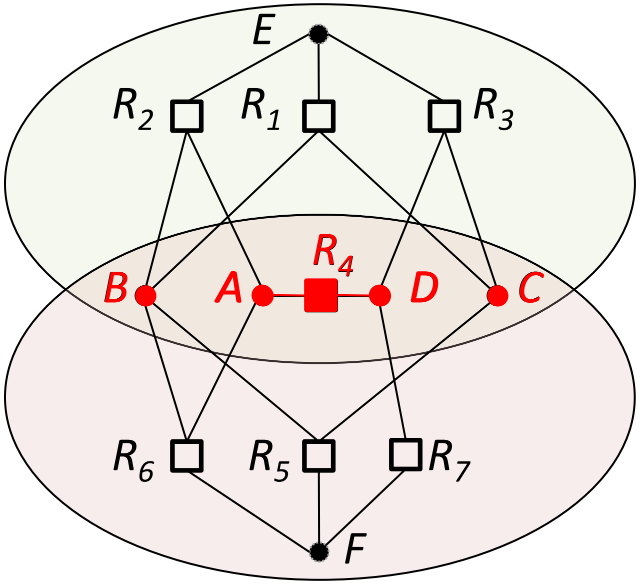
[Karakashian+ AAAI 10]

- A CSP is R(∗,m)C iff
  - Every tuple in a relation can be extended
  - to the variables in the scope of any (m-1) other relations
  - in an assignment satisfying all m relations simultaneously
- R(∗,m)C ≡ Every set of m relations is minimal



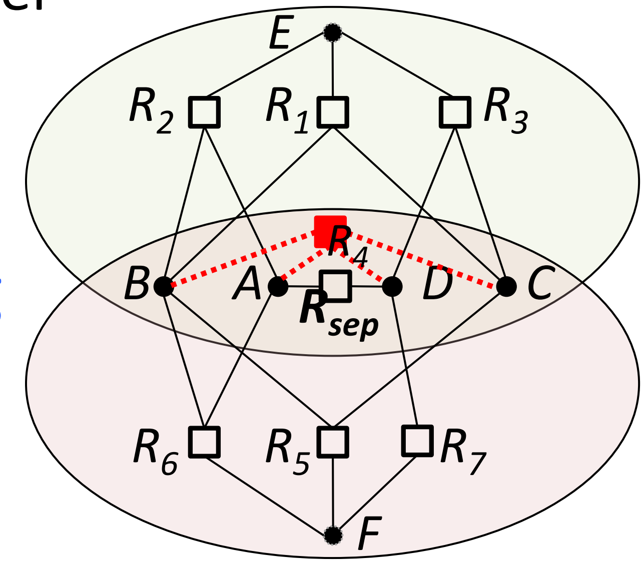
Localize Consistency

- Restricting R($*$,m)C to clusters: cl-R($*$,m)C
- Two clusters communicate via their separator
  - Constraints common to the two clusters
  - Domains of variables common to the two clusters

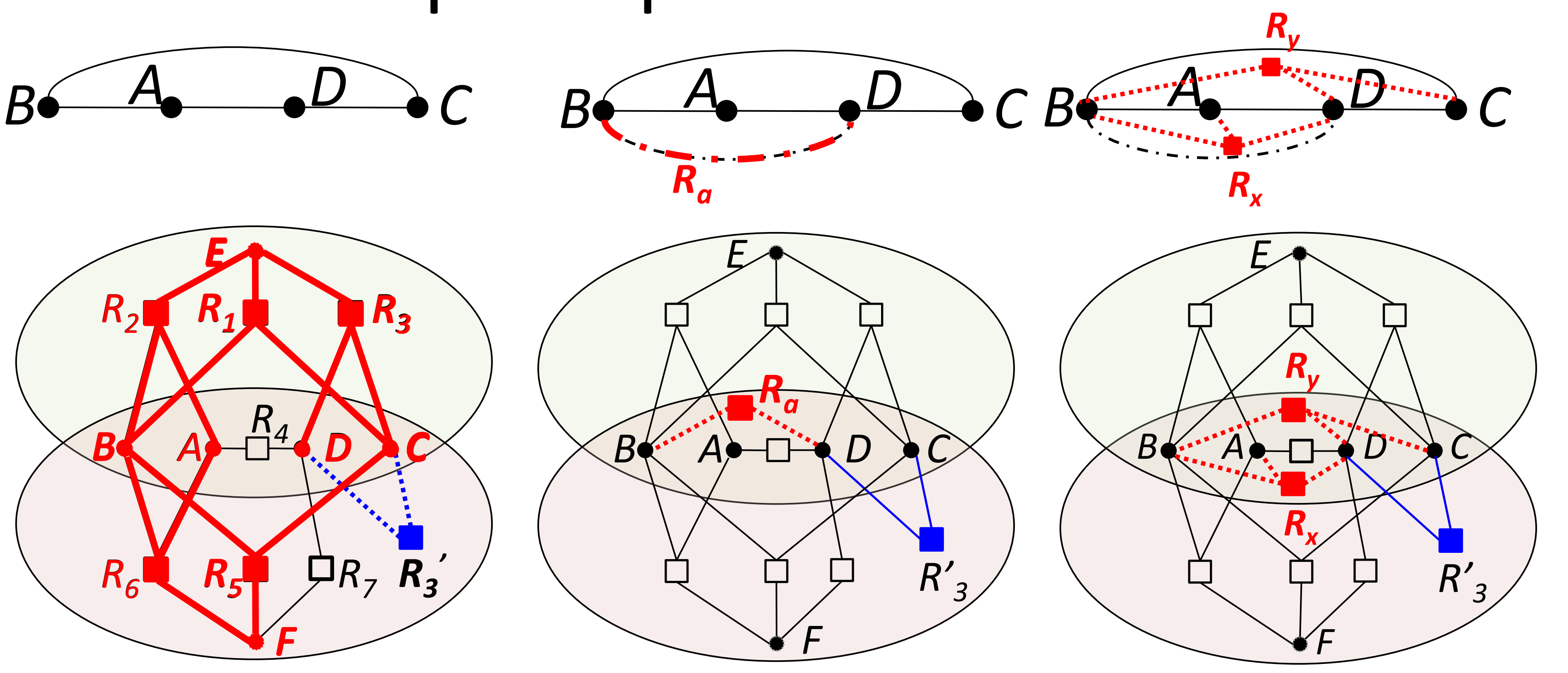


Bolstering Propagation at Separators

- Localization cl-R(*,*m*)C
  - Fewer combinations of *m* relations
  - Reduces the enforced consistency level
- Ideally: add unique constraint
  - Space overhead, major bottleneck
- Enhance propagation by bolstering
  - Projection of existing constraints
  - Adding binary constraints
  - Adding clique constraints



Bolstering Schemas: Approximate Unique Separator Constraint

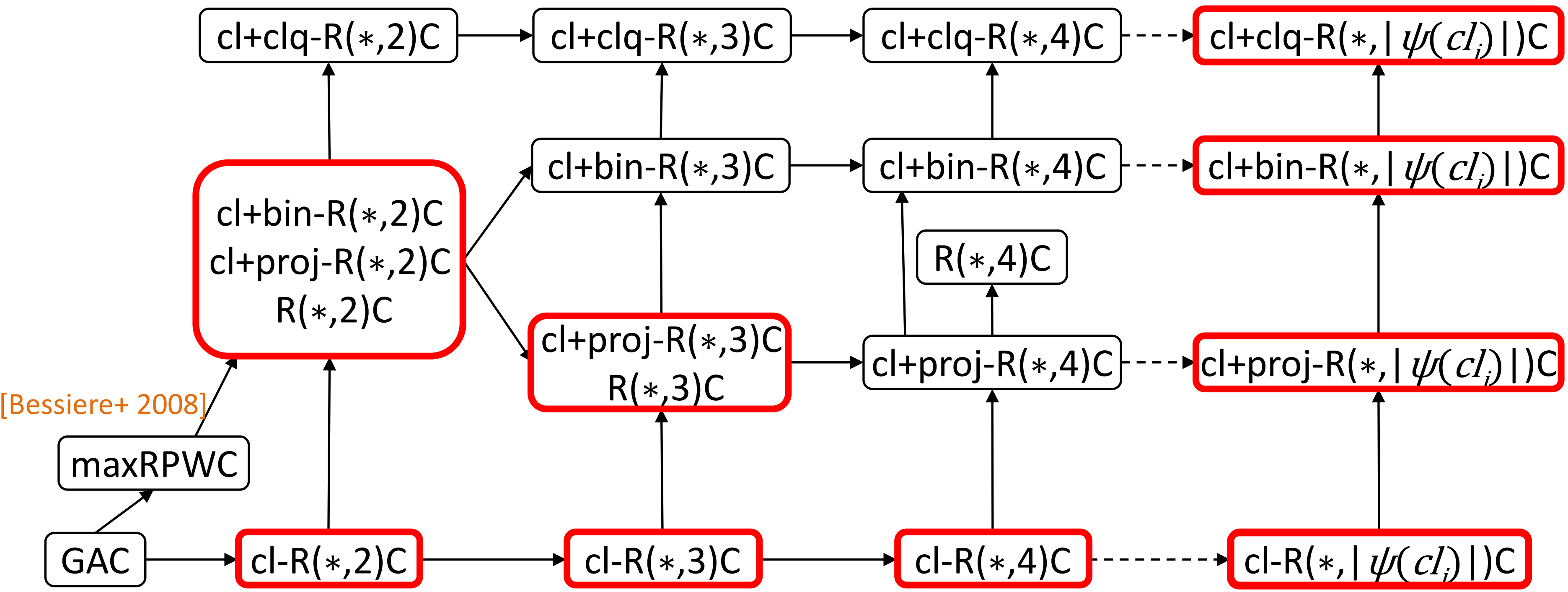


Projection
cl+proj-R(*,m)C

Binary constraints
cl+bin-R(*,m)C

Clique constraints
cl+clq-R(*,m)C

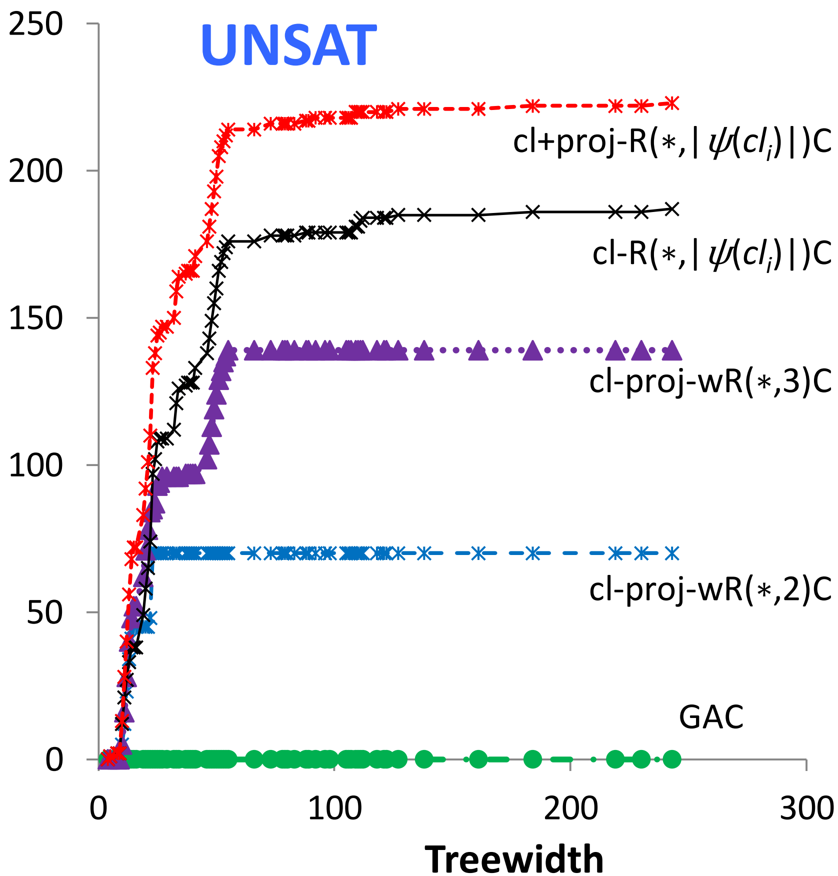
Resulting Consistency Properties

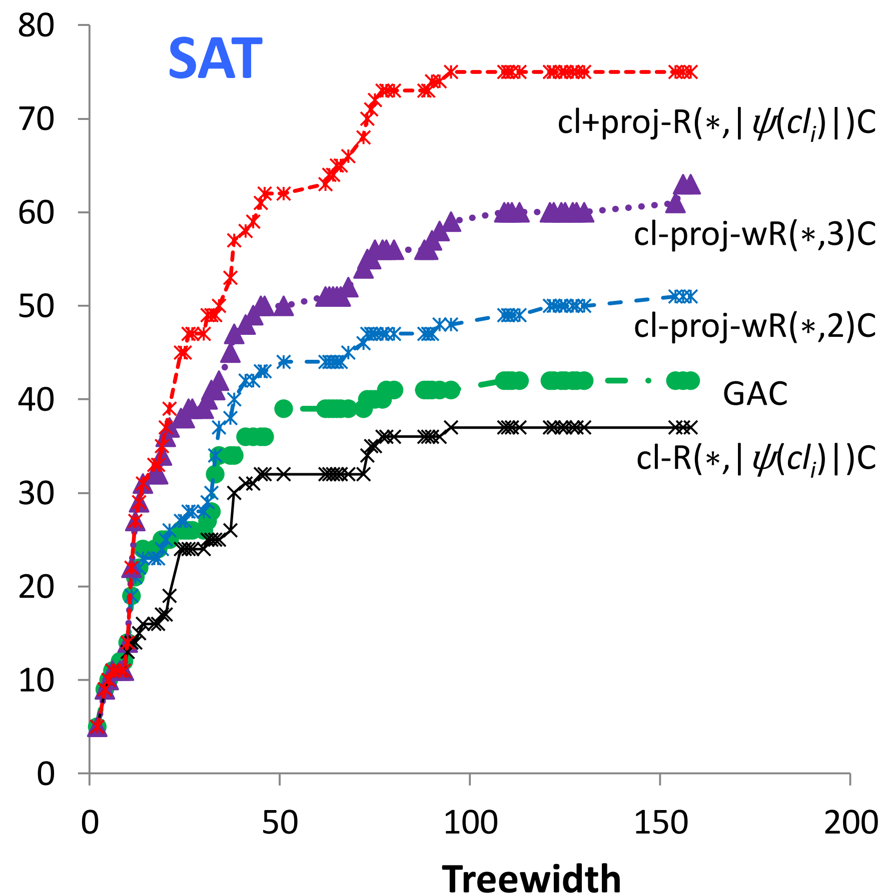


Empirical Evaluations

+ maxRPWC, *m*=3,4			wR(∗,2)C					R(∗,| (cl$_i$)|)C			
	#inst.	**GAC**	**global**	**local**	**Proj.**	**binary**	**clique**	**local**	**Proj.**	**binary**	**clique**
Completed	UNSAT	167	170	167	172	169	162	285	**286**	282	271
	479	34.9%	35.5%	34.9%	35.9%	35.3%	33.8%	59.5%	**59.7%**	58.9%	56.6%
	SAT	174	**179**	178	176	169	104	152	138	124	113
	200	87.0%	**89.5%**	89.0%	88.0%	84.5%	52.0%	76.0%	69.0%	62.0%	56.5%
BT-Free	UNSAT	0	70	39	70	70	74	187	**223**	223	213
	479	0.0%	14.6%	8.1%	14.6%	14.6%	15.4%	39.0%	**46.6%**	46.6%	44.5%
	SAT	44	55	37	53	52	38	39	**77**	71	58
	200	22.0%	27.5%	18.5%	26.5%	26.0%	19.0%	19.5%	**38.5%**	35.5%	29.0%
Min(#NV)	UNSAT	17	73	43	72	72	77	220	**249**	248	239
	479	3.5%	15.2%	9.0%	15.0%	15.0%	16.1%	45.9%	**52.0%**	51.8%	49.9%
	SAT	47	64	37	62	61	39	83	**111**	100	79
	200	23.5%	32.0%	18.5%	31.0%	30.5%	19.5%	41.5%	**55.5%**	50.0%	39.5%
Fastest	UNSAT	72	13	35	5	1	1	**176**	108	42	37
	479	15.0%	2.7%	7.3%	1.0%	0.2%	0.2%	**36.7%**	22.5%	8.8%	7.7%
	SAT	**121**	45	47	23	14	12	34	18	13	12
	200	**60.5%**	22.5%	23.5%	11.5%	7.0%	6.0%	17.0%	9.0%	6.5%	6.0%

Cumulative Count of Instances Solved w/o Backtracking





Acknowledgment: Charts suggested by Rina Dechter

Conclusions & Future Work

- Adapted R($*$,m)C to a tree decomposition of the CSP
  - Localizing R($*$,m)C to the clusters
  - Bolstering separators to strengthen the enforced consistency
- Directions for future work
  - R($*$,m)C on non-table constraints via domain filtering
  - Automating the selection of a consistency property
    - Inside clusters
    - During search
  - Modify the structure of a tree decomposition to improve performance (e.g., merging clusters [Fattah & Dechter 1996])

Thank You for Your Attention