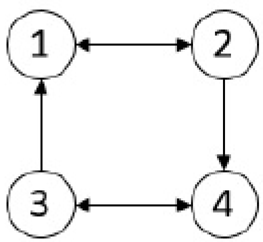
A Comparison of SAT Encodings for Acyclicity of Directed Graphs

Neng-Fa Zhou (CUNY & Relational AI)
Ruiwei Wang (NUS)
Roland H.C. Yap (NUS)

Synthesizing Acyclic Directed Graphs

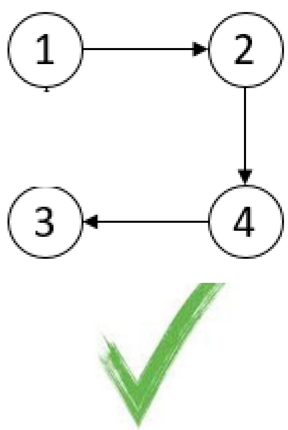
- Given a base directed graph, find a subgraph that is acyclic and satisfies some side constraints.
  - Utility networks (water, electricity, gas, ...)
  - Planning problems
  - Markov networks
  - Neural networks
  - Bayesian networks

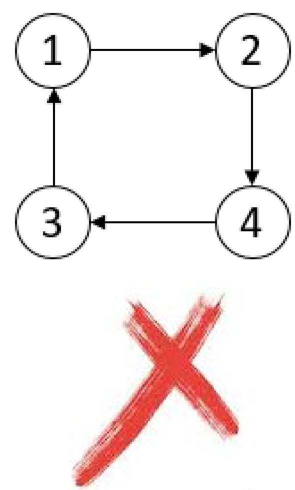
The acyclic_d(V, E) Constraint



acyclic_d(V, E)
V = [v1, v2, v3, v4]
E = [(v1,v2), (v2,v1), (v2, v4),
(v3, v1), (v3,v4), (v4,v3)]

b(v) = 1 iff v is in the subgraph
b((vi, vj)) = 1 iff (vi, vj) is in the subgraph





SAT Encodings of the acyclic_d(V, E) Constraint

- Transitive closure encoding (TCE)
  - A graph is acyclic iff its transitive closure is acyclic.
  - $O(n^3)$ encoding size
- Vertex-elimination encoding (VEE) [RankoohR22]

  - If G' is cyclic, then G is cyclic
  - Much more compact than TCE for sparse graphs
  - Asymptotically the same as TCE

SAT Encodings of the acyclic_d(V, E) Constraint (Cont.)

- Leaf-elimination encoding (LEE)
  - A graph is acyclic if the graph can be reduced to empty after leaves are repeatedly eliminated.

$$G_0 = G \longrightarrow G_1 \quad \cdots \quad \longrightarrow G_t$$

  - Use a time variable for each vertex
  - The same as *tree-reduction encoding* [GebserJR14]
- LEE-u (use unary encoding for the time variable)
- LEE-b (use binary encoding for the time variable)
  - Compact
  - Weak propagation
  - The same as *binary labeling* [JanotaGM17]

SAT Encodings of the acyclic_d(V, E) Constraint (Cont.)

- Hybrid encoding (HYB)
  - Start with VEE
  - Switch to LEE-b when the graph is dense
  - Enjoys VEE's strong propagation and LEE-b's conciseness

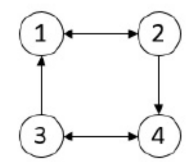
Vertex-Elimination Encoding (VEE)

- Vertex elimination operation

$$G_0 = ([v_1, v_2, \ldots, v_n], E_0)$$

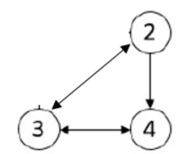
$$\vdots$$

$$G_{i-1} = ([v_i, \ldots, v_n], E_{i-1})$$



$$\downarrow$$

$$G_i = ([v_{i+1}, \ldots, v_n], E_i)$$



- Encoding
  - $b((u, v_i)) \wedge b((v_i, w)) \rightarrow b((u, w))$
  - $\neg b((u, v_i)) \vee \neg b((v_i, u))$
- Encoding size: $O(n^2)$, $O(n \times d^2)$

Leaf-Elimination Encoding Using Unary Encoding for Time Variables (LEE-u)

- Variables
  - $A_{v,t}$, $v \in \{v_1, v_2, \ldots, v_n\}$, $t \in 0..m$
  - $A_{v,t} = 1$ iff vertex v has been eliminated by time t
- Constraints

$$\text{For } v \in V\text{: } \sum_{u \in nbs^+(v)} b((v,u)) = 0 \leftrightarrow A_{v,0} \quad (1)$$

$$\text{For } v \in V,\ t \in 1..m\text{: } A_{v,t-1} \rightarrow A_{v,t} \quad (2)$$

$$\text{For } v \in V,\ t \in 1..m,\ u \in nbs^+(v)\text{: } b((v,u)) \wedge \neg A_{u,t-1} \rightarrow \neg A_{v,t} \quad (3)$$

$$\text{For } v \in V\text{: } A_{v,m} \quad (4)$$

- Encoding size: O(n × m), O(n × m × d)

Leaf-Elimination Encoding Using Binary Encoding for Time Variables (LEE-b)

- Variables
  - $T_v' \in 0..m$
- Constraints

For $v \in V$: $\sum_{u \in nbs^+(v)} b((v,u)) = 0 \leftrightarrow T_v = 0$ (5)

For $v \in V$, $u \in nbs^+(v)$: $b((v,u)) \rightarrow T_v > T_u$ (6)

- Encoding size: $O(n \times \log_2 m)$, $O(n \times \log_2 m \times d)$

Hybrid Encoding (HYB)

- Start with VEE
- Switch to LEE-b when the graph is dense
  - we show the correctness of HYB

$$|E_0 \bigcup E_1 \cdots \bigcup E_i| \geq \min(2.3\times|E_0|,\ 30\times|V_0|)$$

Experimental Results

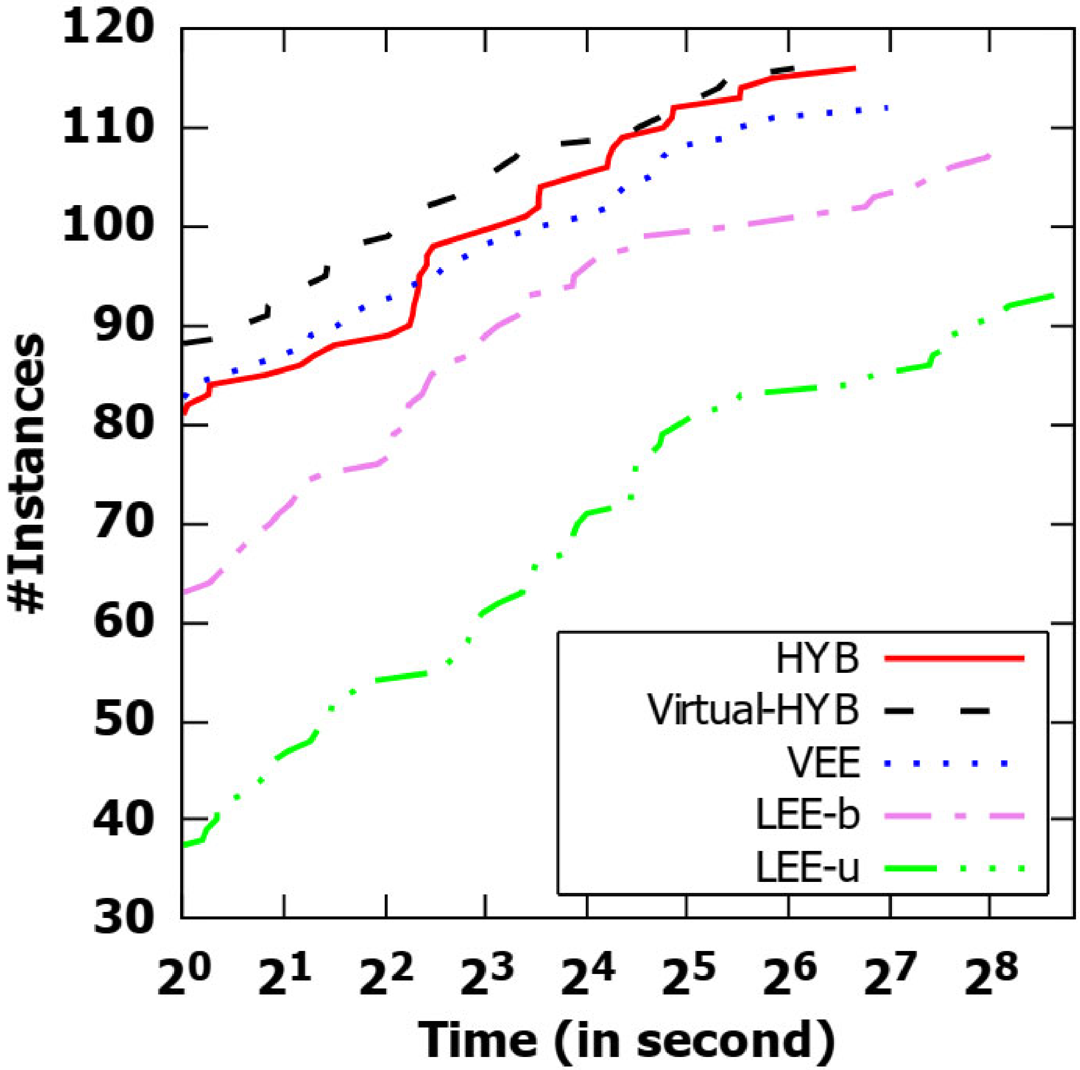
- Implementation
  - Picat (picat-lang.org)
  - Kissat
- Benchmarks
  - GraphSAT benchmarks (n from 15 to 10002)
- Time limit: 10m per instance
- Platform
  - Ubuntu with a 3.20GHz and 16G RAM
  - Intel i7-8700 machine.

Experimental Results (con.t)

#Insts	Benchmark	LEE-u	LEE-b	VEE	HYB	Virtual-HYB
26	COMB	28.03	42.54	**0.13**	**0.13**	**0.13**
31	EMPTYCORNER	160.89	1.99	3.84	1.99	**1.51**
36	EMPTYMIDDLE	67.01	6.93	0.92	2.02	**0.87**
9	ESCAPE	379.45	22.74	285.91	23.64	**15.38**
14	ROOMCHAIN	343.35	342.99	15.38	15.34	**11.94**
	TOTAL	16350.64	6423.84	2944.2	565.29	**387.09**
116	**AVE**	140.95	55.38	25.38	4.87	**3.34**
	#SOLVED	94	108	112	**116**	**116**

Table 1 Summary of results

Experimental Results (con.t)



(a) Runtime distribution

Conclusion

- Contributions
  - A comparison of SAT encodings for acyclicity, including VEE, LEE-u, and LEE-b
  - Finding of a hybrid encoding, which combines VEE and LEE-b
- Future work
  - Heuristics for switching from VEE to LEE-b
  - Explore hybridization for encoding other constraints