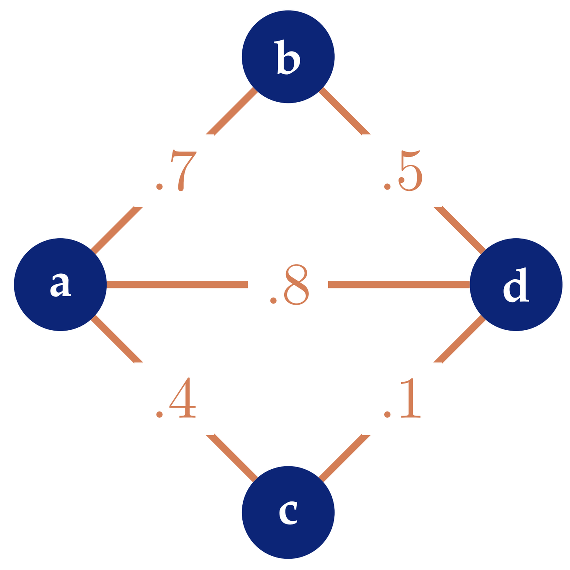
A Propagator for Ordered Binary Decision Diagrams

Universiteit Leiden
The Netherlands

Anna L.D. Latour, Behrouz Babaki, Siegfried Nijssen
a.l.d.latour@liacs.leidenuniv.nl

Example Problem



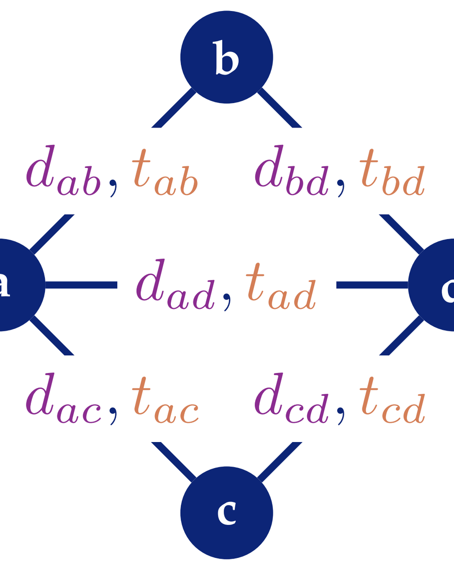
Vertices in probabilistic networks are related by **stochastic processes**. Information/interaction travels along undirected paths **probabilistically**.

Theory compression problem: given pairs of interacting nodes, find a subset of edges such that:

- the size of this subset is minimized;
- the expected number of interacting pairs in the subgraph is at least k.

Decide which edges to include in the subset/theory.

Logical Model



For edge (x, y), introduce two **variables**:

$d_{xy} \in \{0, 1\}$ (by **decision**),

$t_{xy} \in \{0, 1\}$ (by **chance**),

where x and y interact if $d_{xy} \wedge t_{xy} = \top$.

Stochastic Constraint

The **probability** that a and c interact must be **at least** θ:

$$P(\phi \mid \sigma) \geq \theta,$$

with σ a **strategy** (a **choice** of 0 or 1 for each d_{xy}), and

$$\phi = (d_{ac} \wedge t_{ac}) \vee (d_{ad} \wedge t_{ad} \wedge d_{cd} \wedge t_{cd}) \vee (d_{ab} \wedge t_{ab} \wedge d_{bd} \wedge t_{bd} \wedge d_{cd} \wedge t_{cd}).$$

Solving requires

- Weighted Model Counting;
- evaluating (all) **strategies**.

Constraint Programming: Search & Propagation

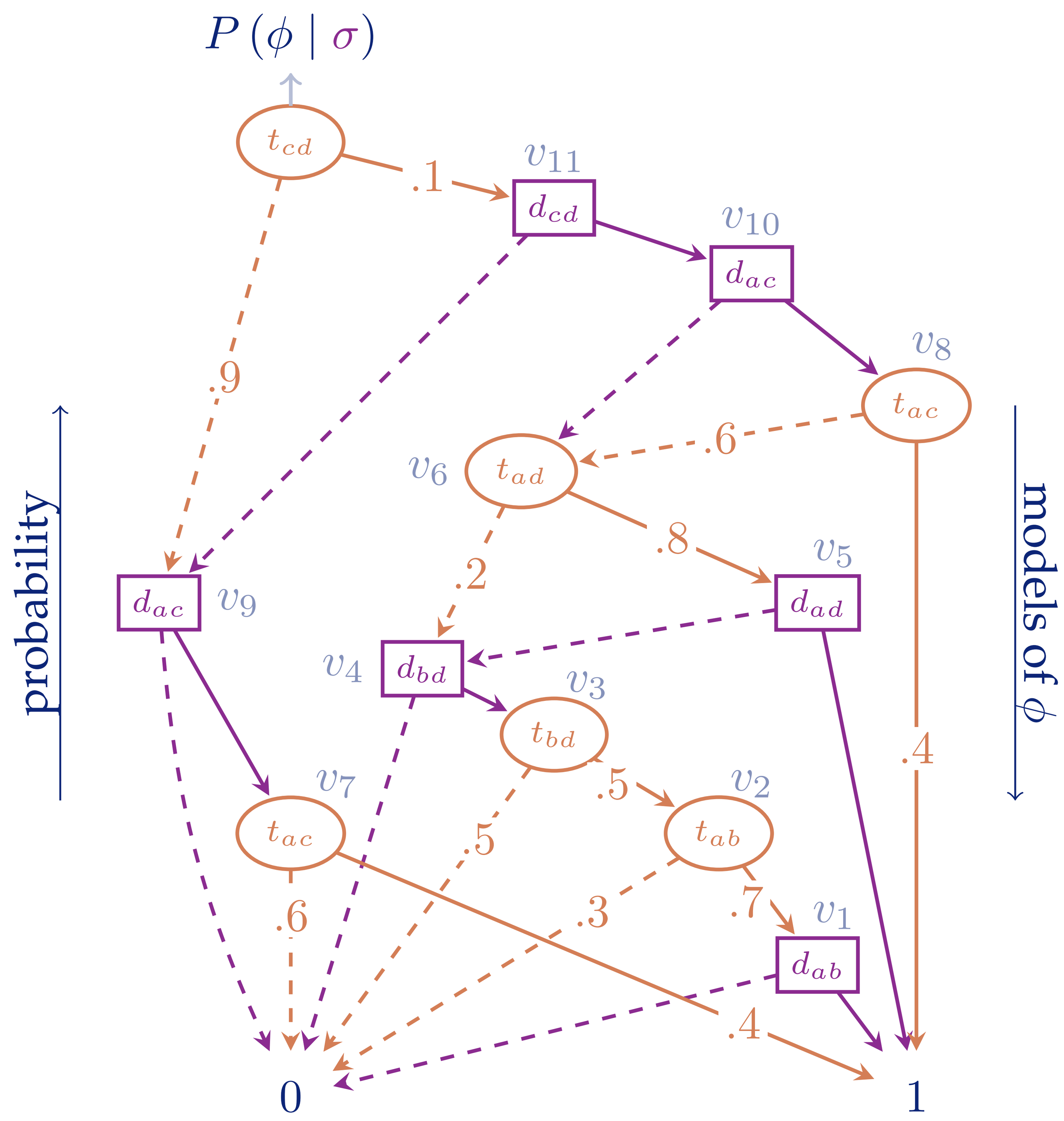
There are **two main** Constraint Programming (CP) techniques. We traverse a **search tree** by selecting a variable, assigning a value to it, evaluating the constraint under this **partial strategy**, backtracking when needed, until a solution is found. **Propagation** makes the domains of variables **domain consistent**, reducing size of the search space.

Ordered Binary Decision Diagram (OBDD)

- Compile ϕ into OBDD.
- Paths from root to 1 represent **models** of ϕ.
- Associate weight $w = p_{xy}$ with outgoing **hi** arc of nodes labeled with t_{xy}.
- Associate weight $(1 - w)$ with outgoing **lo** arc of nodes labeled with t_{xy}.
- Outgoing arcs of nodes labeled with d_{xy} get weights $w \in \{0, 1\}$ to reflect σ.
- Compute **value** v of each node as:

$$v = w * v_{hi} + (1 - w) * v_{lo}$$

- Value of root equals $P(\phi \mid \sigma)$.



Evaluate OBDD for each **strategy** σ, and **check constraint**. Improve efficiency by using **constraint solver** that uses **search** and **propagation** to **prune** part of the search space.

Partial Derivatives

Goal: identify all ds that need to be fixed to $\top$ for domain consistency.

Method: compute partial derivatives $\partial f / \partial d$, where f is a polynomial that expresses the value of an OBDD.

The following holds for each d:

$$\frac{\partial f\left(d, \sigma_\top \setminus d\right)}{\partial d} = f\left(\sigma_\top\right) - f\left(\sigma_\top \setminus d, d = \bot\right),$$

with $\sigma_\top$ a **strategy** where each $d = \top$. If

$$f\left(\sigma_\top\right) - f\left(\sigma_\top \setminus d, d = \bot\right) < \theta,$$

we fix d to $\top$. This guarantees domain consistency.

Derivatives allow **incremental** computation of $f = P(\phi \mid \sigma)$. This yields a propagation **complexity** that is **linear** rather than **polynomial**.

Decomposition

Method: decompose OBDD into small constraints, solve with CP solver.

$\theta \leq 0.1 * v_{11} + 0.9 * v_9$

$v_{11} = d_{cd} * v_{10} + (1 - d_{cd}) * v_9$ $\quad 0 \leq v_{11} \leq 0.93$

$v_{10} = d_{ac} * v_8 + (1 - d_{ac}) * v_6$ $\quad 0 \leq v_{10} \leq 0.93$

$\vdots$ $\quad \vdots$

$v_2 = 0.7 * v_1$ $\quad 0 \leq v_2 \leq 0.7$

$v_1 = d_{ab}$

$d_{ab}, d_{ac}, d_{ad}, d_{bd}, d_{cd} \in \{0, 1\}$

Problem: domain consistency not guaranteed for these constraints.

Contributions

- A general approach for solving problems modeled in ProbLog with stochastic constraints;
- A new solving method based on using derivatives in CP.

Weighted Model Counting (WMC)

Consider a **strategy** $\sigma_i = \{d_{ab}, d_{ad}, d_{bd}, d_{cd}\}$.

To compute $P(\phi \mid \sigma_i)$, we

- enumerate models of $\phi(\sigma_i)$;
- compute their weights;
- sum the weights.

model	weight
$\{t_{ad}, t_{cd}\}$	$.3 * .6 * .8 * .5 * .1 = .0072$
$\{t_{ac}, t_{ad}, t_{cd}\}$	$.3 * .4 * .8 * .5 * .1 = .0048$
$\{t_{ab}, t_{ad}, t_{bd}, t_{cd}\}$	$.7 * .4 * .8 * .5 * .1 = .0112$
$\vdots$	$\vdots$
	$.087$

References

A. Darwiche, *A Differential Approach to Inference in Bayesian Networks*, ACM, 2003

L. De Raedt, A. Kimmig, H. Toivonen, *ProbLog: A Probabilistic Prolog and Its Application in Link Discovery*, IJCAI, 2007

A.L.D. Latour, B. Babaki, A. Dries, A. Kimmig, G. Van den Broeck, S. Nijssen, *Combining Stochastic Constraint Optimization and Probabilistic Programming*, CP, 2017

Anna Latour
www.annalatour.nl
a.l.d.latour@liacs.leidenuniv.nl