Stochastic Constraint Propagation for Mining Probabilistic Networks

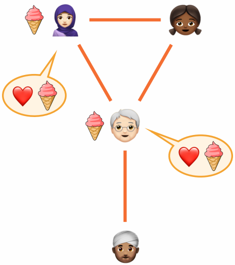
Anna Louise Latour, Behrouz Babaki, Siegfried Nijssen.

Universiteit Leiden The Netherlands

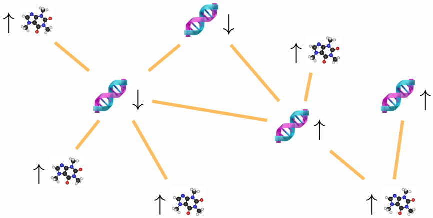
POLYTECHNIQUE MONTRÉAL

UCLouvain

Viral Marketing

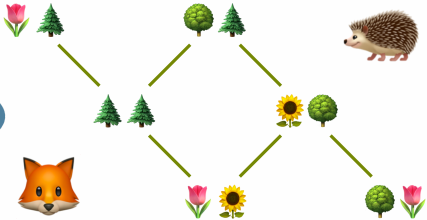


Signalling Regulatory Pathways

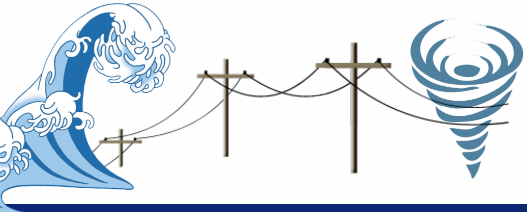


Stochastic **C**onstraint **O**ptimization **P**roblems

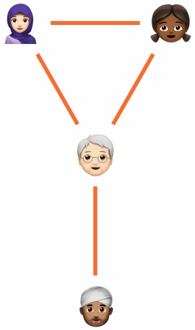
Landscape Connectivity



Powergrid Reliability

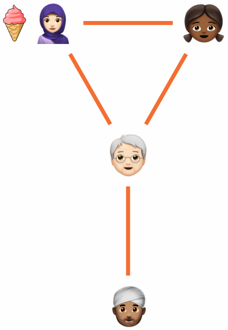


Example: Viral Marketing Problem



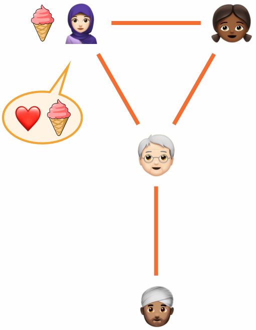
David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



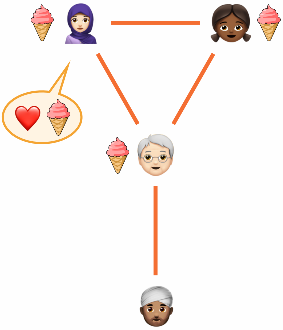
David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



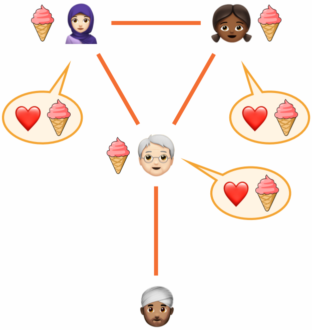
David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



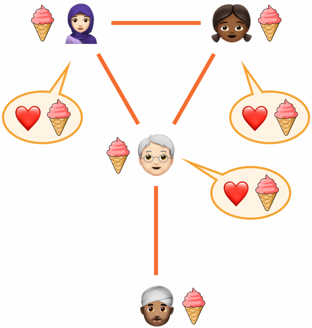
David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



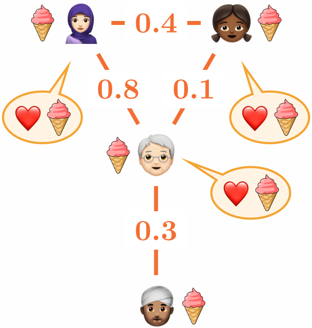
David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



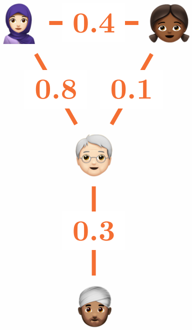
David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

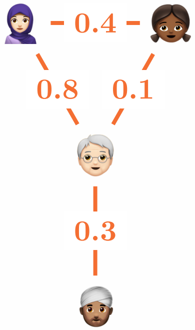
Example: Viral Marketing Problem



Properties

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem

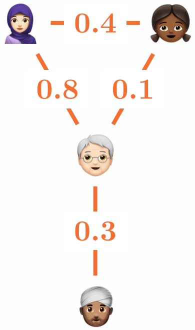


Properties

- **Probabilistic** influence;

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem

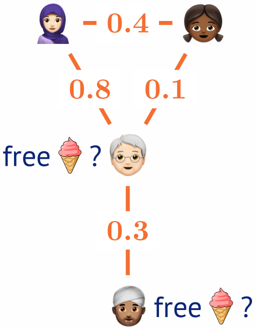


Properties

- **Probabilistic** influence;
- limited **budget** of free samples ;

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem

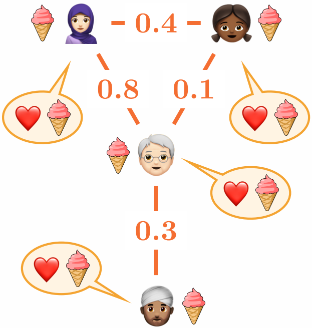


Properties

- **Probabilistic** influence;
- limited **budget** of free samples ;

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem

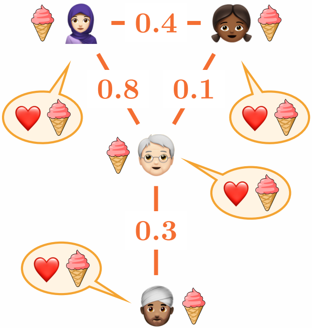


Properties

- **Probabilistic** influence;
- limited **budget** of free samples 🍦🍦;
- **maximize** expected # people buying your ice cream.

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



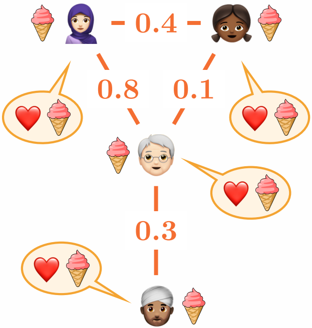
Properties

- **Probabilistic** influence;
- limited **budget** of free samples 🍦🍦;
- **maximize** expected # people buying your ice cream.

Exact solving is **NP-hard**

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



Properties

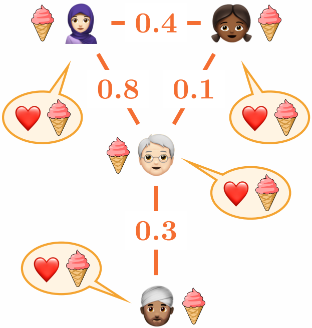
- **Probabilistic** influence;
- limited **budget** of free samples 🍦🍦;
- **maximize** expected # people buying your ice cream.

Exact solving is **NP-hard**

- Exponential # of **strategies**;

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Example: Viral Marketing Problem



Properties

- **Probabilistic** influence;
- limited **budget** of free samples 🍦🍦;
- **maximize** expected # people buying your ice cream.

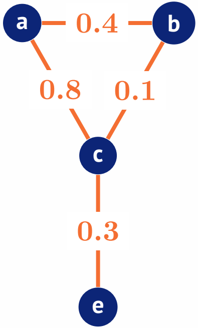
Exact solving is **NP-hard**

- Exponential # of **strategies**;
- **Probabilistic inference** is **#P-complete**.

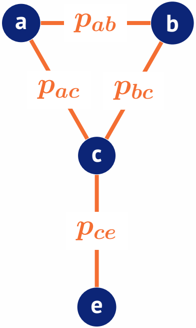
David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the spread of influence through a social network
KDD, 2003

Dan Roth
The hardness of approximate reasoning
Artif. Intell., 1996

Example: Viral Marketing Problem



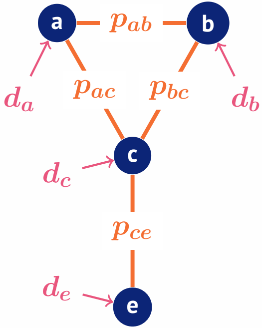
Example: Viral Marketing Problem



$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$

Boolean influence relationships are **independent**.

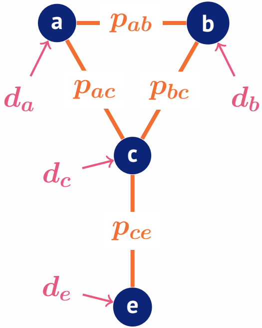
Example: Viral Marketing Problem



$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Boolean influence relationships are **independent**.

Example: Viral Marketing Problem

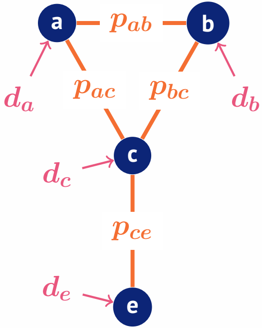


$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Boolean influence relationships are **independent**.

Simplifying assumptions

Example: Viral Marketing Problem



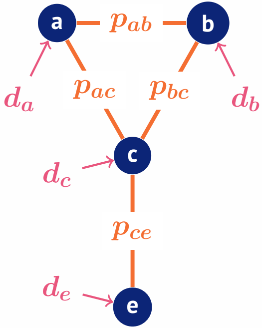
$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Boolean influence relationships are **independent**.

Simplifying assumptions

- influence relationships are symmetric;

Example: Viral Marketing Problem



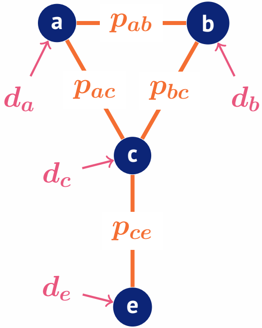
$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Boolean influence relationships are **independent**.

Simplifying assumptions

- influence relationships are symmetric;
- if person i gets a free sample ($d_i = 1$), they will buy it in the future;

Example: Viral Marketing Problem



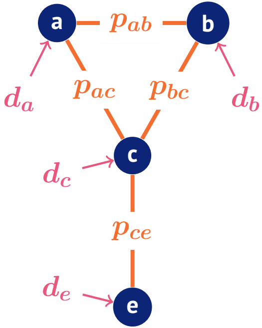
$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Boolean influence relationships are **independent**.

Simplifying assumptions

- influence relationships are symmetric;
- if person i gets a free sample ($d_i = 1$), they will buy it in the future;
- if person i buys ice cream and they have influence over j ($t_{ij} = 1$), then j will buy ice cream.

Example: Viral Marketing Problem

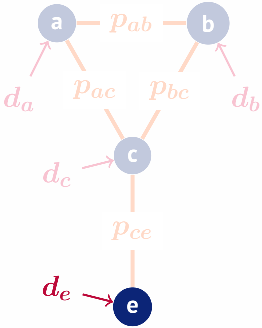


$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

$$\phi_e =$$

Example: Viral Marketing Problem

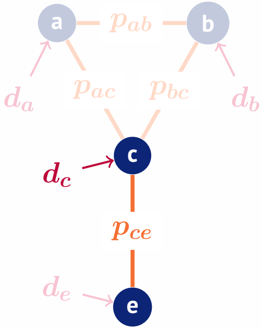


$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

$$\phi_e = d_e \vee$$

Example: Viral Marketing Problem

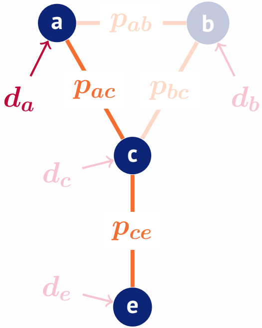


$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

$$\phi_e = d_e \vee (d_c \wedge t_{ce}) \vee$$

Example: Viral Marketing Problem

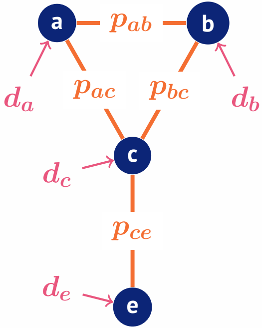


$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

$$\phi_e = d_e \lor (d_c \land t_{ce}) \lor$$
$$(d_a \land t_{ac} \land t_{ce}) \lor$$

Example: Viral Marketing Problem

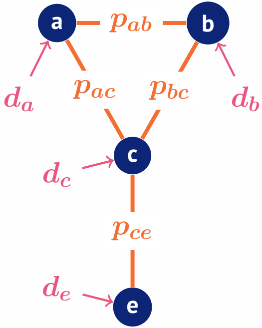


$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

$$\phi_e = d_e \vee (d_c \wedge t_{ce}) \vee (d_a \wedge t_{ac} \wedge t_{ce}) \vee (d_b \wedge t_{bc} \wedge t_{ce}) \vee (d_a \wedge t_{ab} \wedge t_{bc} \wedge t_{ce}) \vee (d_b \wedge t_{ab} \wedge t_{ac} \wedge t_{ce})$$

Example: Viral Marketing Problem



$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

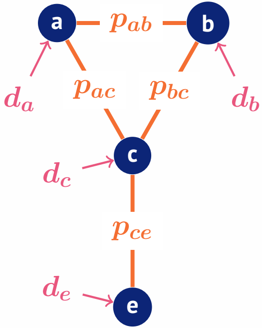
Person e buys ice cream:

$$\phi_e = d_e \vee (d_c \wedge t_{ce}) \vee (d_a \wedge t_{ac} \wedge t_{ce}) \vee (d_b \wedge t_{bc} \wedge t_{ce}) \vee (d_a \wedge t_{ab} \wedge t_{bc} \wedge t_{ce}) \vee (d_b \wedge t_{ab} \wedge t_{ac} \wedge t_{ce})$$

find **strategy σ**:

$$\arg\max_{\sigma} \sum_{i \in \{a,b,c,e\}} P(\phi_i \mid \sigma)$$

Example: Viral Marketing Problem



$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

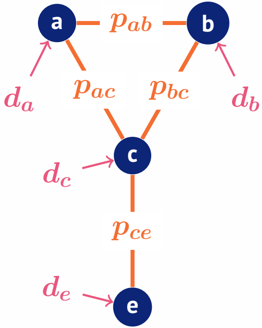
$$\begin{aligned}\phi_e = \; & d_e \vee (d_c \wedge t_{ce}) \vee \\ & (d_a \wedge t_{ac} \wedge t_{ce}) \vee \\ & (d_b \wedge t_{bc} \wedge t_{ce}) \vee \\ & (d_a \wedge t_{ab} \wedge t_{bc} \wedge t_{ce}) \vee \\ & (d_b \wedge t_{ab} \wedge t_{ac} \wedge t_{ce})\end{aligned}$$

find **strategy σ**:

$$\arg\max_{\sigma} \sum_{i \in \{a,b,c,e\}} P(\phi_i \mid \sigma)$$

subject to: $\sum_{i \in \{a,b,c,e\}} d_i \leq k$

Example: Viral Marketing Problem



$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

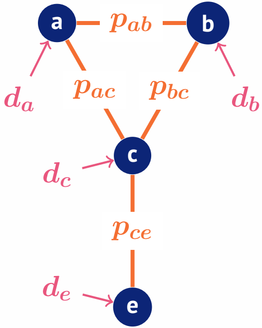
$$\begin{aligned}\phi_e = \; & d_e \vee (d_c \wedge t_{ce}) \vee \\ & (d_a \wedge t_{ac} \wedge t_{ce}) \vee \\ & (d_b \wedge t_{bc} \wedge t_{ce}) \vee \\ & (d_a \wedge t_{ab} \wedge t_{bc} \wedge t_{ce}) \vee \\ & (d_b \wedge t_{ab} \wedge t_{ac} \wedge t_{ce})\end{aligned}$$

repeatedly solve:

$$\sum_{i \in \{a,b,c,e\}} P(\phi_i \mid \sigma) > \theta$$

subject to: $\sum_{i \in \{a,b,c,e\}} d_i \leq k$

Example: Viral Marketing Problem



$$P(t_{xy} = 1) = p_{xy}$$
$$P(t_{xy} = 0) = (1 - p_{xy})$$
$$d_i \in \{0, 1\}$$

Person e buys ice cream:

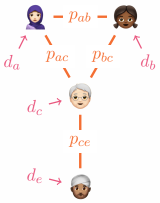
$$\begin{aligned}\phi_e = \; & d_e \vee (d_c \wedge t_{ce}) \vee \\ & (d_a \wedge t_{ac} \wedge t_{ce}) \vee \\ & (d_b \wedge t_{bc} \wedge t_{ce}) \vee \\ & (d_a \wedge t_{ab} \wedge t_{bc} \wedge t_{ce}) \vee \\ & (d_b \wedge t_{ab} \wedge t_{ac} \wedge t_{ce})\end{aligned}$$

repeatedly solve:

$$\sum_{i \in \{a,b,c,e\}} P(\phi_i \mid \sigma) > \theta$$

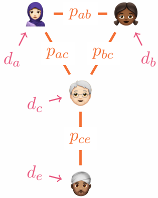
subject to: $\sum_{i \in \{a,b,c,e\}} d_i \leq k$

Existing (generic) method



A.L.D. Latour, B. Babaki, A. Dries, A. Kimmig, G. Van den Broeck, S. Nijssen.
Combining Stochastic Constraint Optimization and Probabilistic Programming: From Knowledge Compilation to Constraint Solving. CP, 2017

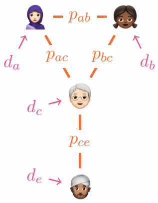
Existing (generic) method



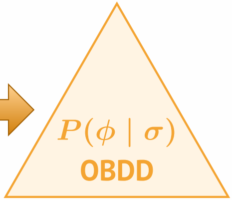
$$\begin{aligned}\phi_e = & d_e \lor (d_c \land t_{ce}) \lor \\ & (d_a \land t_{ac} \land t_{ce}) \lor \\ & (d_b \land t_{bc} \land t_{ce}) \lor \\ & (d_a \land t_{ab} \land t_{bc} \land t_{ce}) \lor \\ & (d_b \land t_{ab} \land t_{ac} \land t_{ce})\end{aligned}$$

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

Existing (generic) method

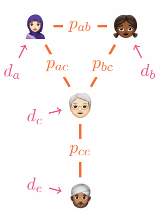


$$\begin{aligned}\phi_e = & d_e \lor (d_c \land t_{ce}) \lor \\ & (d_a \land t_{ac} \land t_{ce}) \lor \\ & (d_b \land t_{bc} \land t_{ce}) \lor \\ & (d_a \land t_{ab} \land t_{bc} \land t_{ce}) \lor \\ & (d_b \land t_{ab} \land t_{ac} \land t_{ce})\end{aligned}$$



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

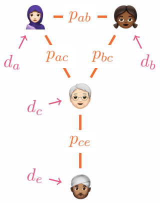
Existing (generic) method



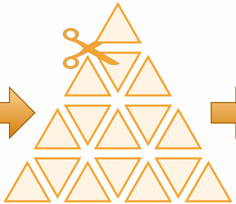
$$
\begin{aligned}
\phi_e = & d_e \lor (d_c \land t_{ce}) \lor \\
& (d_a \land t_{ac} \land t_{ce}) \lor \\
& (d_b \land t_{bc} \land t_{ce}) \lor \\
& (d_a \land t_{ab} \land t_{bc} \land t_{ce}) \lor \\
& (d_b \land t_{ab} \land t_{ac} \land t_{ce})
\end{aligned}
$$

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

Existing (generic) method



$$\phi_e = d_e \lor (d_c \land t_{ce}) \lor (d_a \land t_{ac} \land t_{ce}) \lor (d_b \land t_{bc} \land t_{ce}) \lor (d_a \land t_{ab} \land t_{bc} \land t_{ce}) \lor (d_b \land t_{ab} \land t_{ac} \land t_{ce})$$

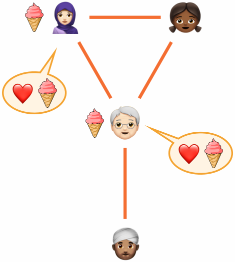


CP
or
MIP

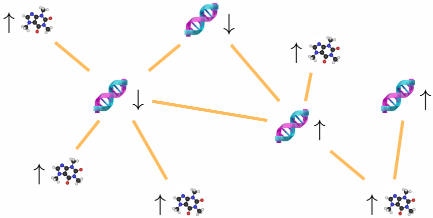
A.L.D. Latour, B. Babaki, A. Dries, A. Kimmig, G. Van den Broeck, S. Nijssen. *Combining Stochastic Constraint Optimization and Probabilistic Programming: From Knowledge Compilation to Constraint Solving.* CP, 2017

Observation 1: existing method does **not guarantee** Generalized Arc Consistency (**GAC**) $\longrightarrow$ **inefficient**;

Viral Marketing

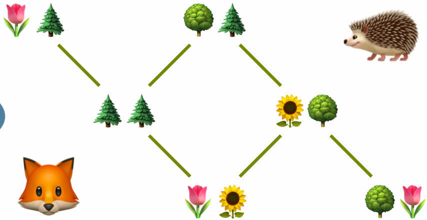


Signalling Regulatory Pathways

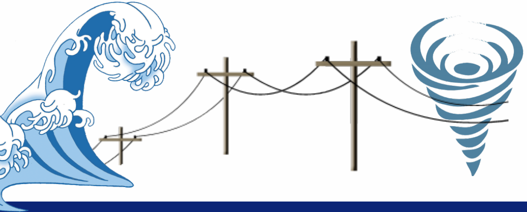


monotonic distributions

Landscape Connectivity



Powergrid Reliability



Observation 1: existing method does **not guarantee** Generalized Arc Consistency (**GAC**) $\longrightarrow$ **inefficient**;

Observation 2: probability distribution is **monotonic**;

Observation 1: existing method does **not guarantee** Generalized Arc Consistency (**GAC**) $\longrightarrow$ **inefficient**;

Observation 2: probability distribution is **monotonic**;

Recall: optimization is repeated **constraint solving**:

$$\text{solve } \sum_{\phi \in \mathbf{\Phi}} P(\phi \mid \sigma) > \theta \text{ for increasing } \theta;$$

Observation 1: existing method does **not guarantee** Generalized Arc Consistency (**GAC**) $\longrightarrow$ **inefficient**;

Observation 2: probability distribution is **monotonic**;

Recall: optimization is repeated **constraint solving**:

$$\text{solve } \sum_{\phi \in \boldsymbol{\Phi}} P(\phi \mid \sigma) > \boldsymbol{\theta} \text{ for increasing } \boldsymbol{\theta};$$

GOAL: create constraint propagation algorithm for Stochastic Constraints on Monotonic Distributions (SCMDs), which guarantees GAC.

Recall:

- Stochastic Constraint Optimization Problems (SCOPs) are **NP-hard**

Recall:

- Stochastic Constraint Optimization Problems (SCOPs) are **NP-hard**
  - **exponential** search space;

Recall:

- Stochastic Constraint Optimization Problems (SCOPs) are **NP-hard**
  - **exponential** search space;

leverage CP technology (propagation)

OscaR: Scala in OR, 2012
`oscarlib.org`

Recall:

- Stochastic Constraint Optimization Problems (SCOPs) are **NP-hard**
  - **exponential** search space;
  - probabilistic inference **#P-complete**;

leverage CP technology (propagation)

OscaR: Scala in OR, 2012
`oscarlib.org`

Recall:

- Stochastic Constraint Optimization Problems (SCOPs) are **NP-hard**
  - **exponential** search space;
  - probabilistic inference **#P-complete**;

leverage CP technology (propagation)

leverage PP technology (knowledge compilation)

OscaR: Scala in OR, 2012
`oscarlib.org`

Recall:

- Stochastic Constraint Optimization Problems (SCOPs) are **NP-hard**
  - **exponential** search space;
  - probabilistic inference **#P-complete**;
- Existing method **not GAC**.

leverage CP technology (propagation)

leverage PP technology (knowledge compilation)

OscaR: Scala in OR, 2012
`oscarlib.org`

Recall:

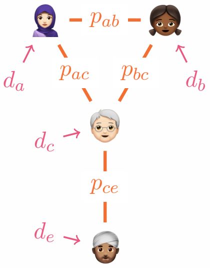
- Stochastic Constraint Optimization Problems (SCOPs) are **NP-hard**
  - **exponential** search space;
  - probabilistic inference **#P-complete**;
- Existing method **not GAC**.

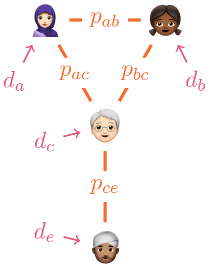
leverage CP technology (propagation)

leverage PP technology (knowledge compilation)

leverage structure (global propagator)

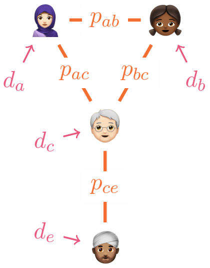
OscaR: Scala in OR, 2012
`oscarlib.org`


p_{ab}
d_a
p_{ac}
p_{bc}
d_b
d_c
p_{ce}
d_e

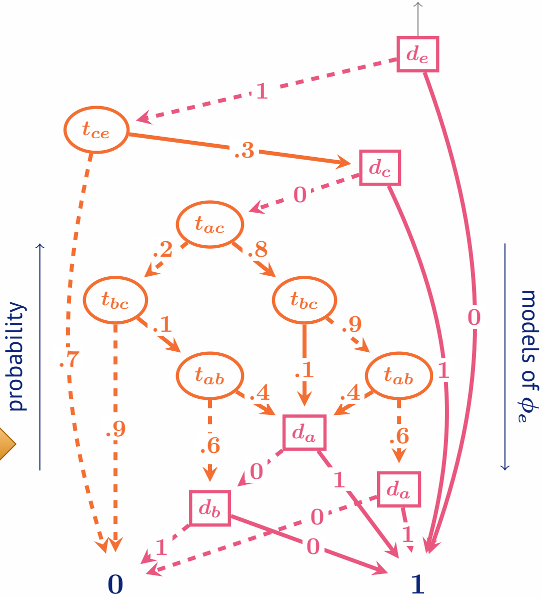


$$
\begin{aligned}
\phi_e = d_e \vee (d_c \wedge t_{ce}) \vee \\
(d_a \wedge t_{ac} \wedge t_{ce}) \vee \\
(d_b \wedge t_{bc} \wedge t_{ce}) \vee \\
(d_a \wedge t_{ab} \wedge t_{bc} \wedge t_{ce}) \vee \\
(d_b \wedge t_{ab} \wedge t_{ac} \wedge t_{ce})
\end{aligned}
$$

$$P\left(\phi_e \mid \sigma = \{d_a, d_c\}\right) \geq \theta$$

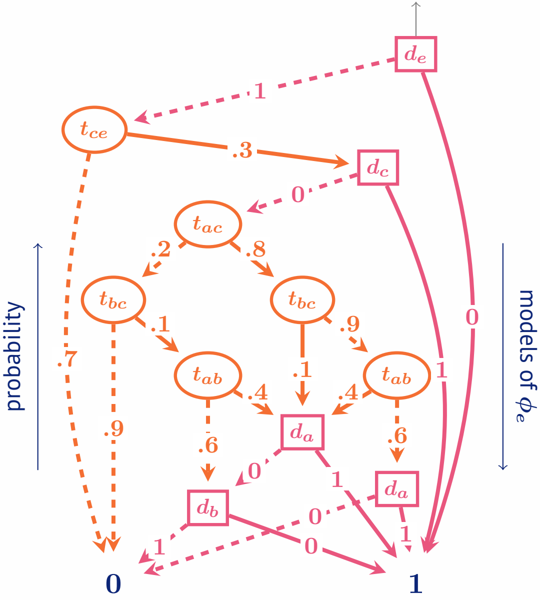


$$\phi_e = d_e \lor (d_c \land t_{ce}) \lor (d_a \land t_{ac} \land t_{ce}) \lor (d_b \land t_{bc} \land t_{ce}) \lor (d_a \land t_{ab} \land t_{bc} \land t_{ce}) \lor (d_b \land t_{ab} \land t_{ac} \land t_{ce})$$



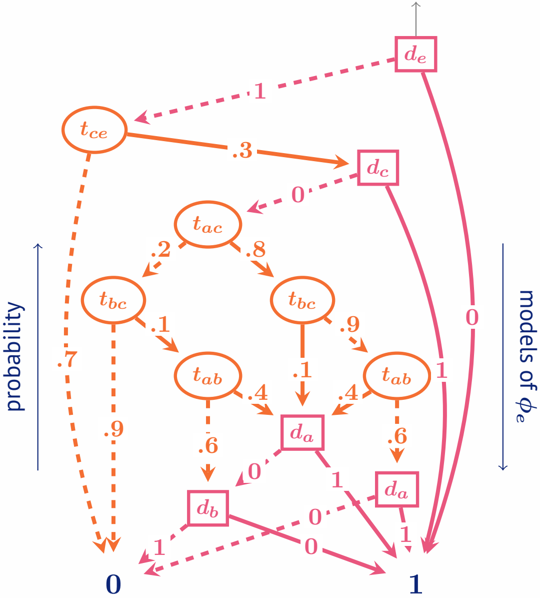
Ordered Binary Decision Diagram (**OBDD**) encodes **probability distribution**

$$P\left(\phi_e \mid \sigma = \{d_a, d_c\}\right) \geq \theta$$



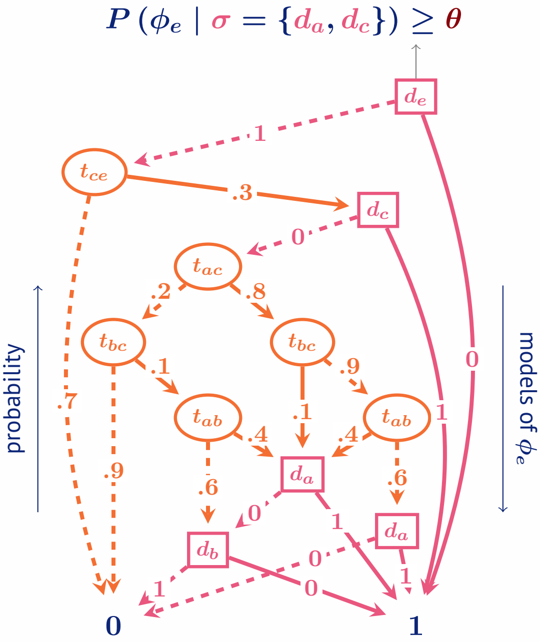
Ordered Binary Decision Diagram (**OBDD**) encodes **probability distribution** (*not* the solutions to the constraint).

$$P(\phi_e \mid \sigma = \{d_a, d_c\}) \geq \theta$$



Ordered Binary Decision Diagram (**OBDD**) encodes **probability distribution** (*not* the solutions to the constraint).

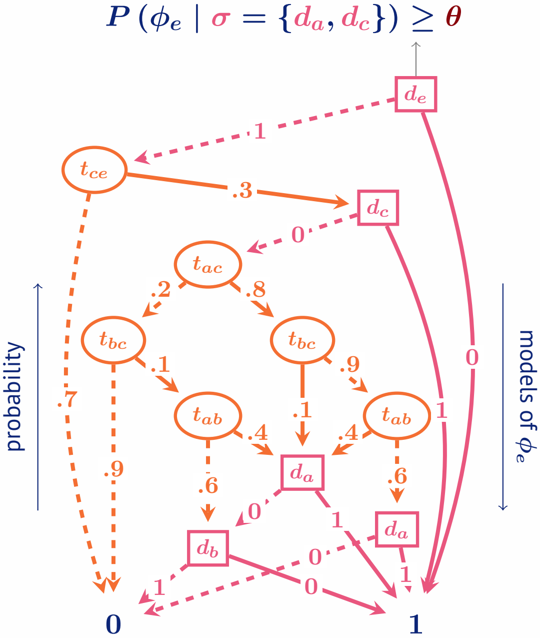
solve $\sum_{\phi \in \Phi} P(\phi \mid \sigma) \geq \theta$



Ordered Binary Decision Diagram (**OBDD**) encodes **probability distribution** (*not* the solutions to the constraint).

solve $\sum_{\phi \in \Phi} P(\phi \mid \sigma) \geq \theta$

Use OBDD to evaluate **strategy σ**. **Complexity** of one sweep: $O(m)$, with $m = |\text{OBDD}|$.

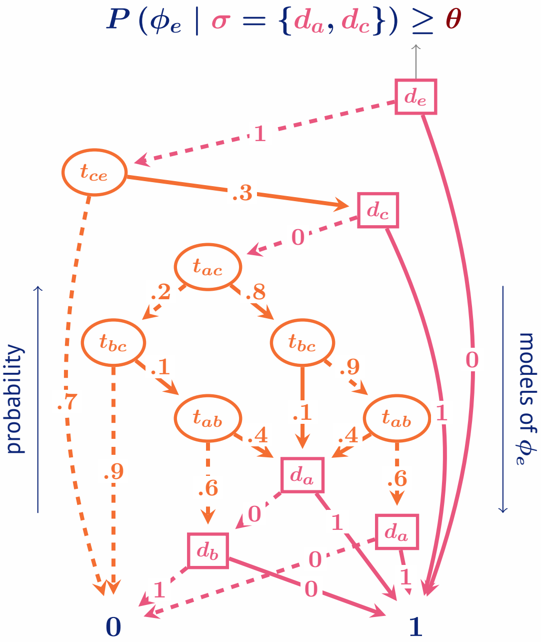


Ordered Binary Decision Diagram (**OBDD**) encodes **probability distribution** (*not* the solutions to the constraint).

solve $\sum_{\phi \in \Phi} P(\phi \mid \sigma) \geq \theta$

Use OBDD to evaluate **strategy σ**. **Complexity** of one sweep: $O(m)$, with $m = |\text{OBDD}|$.

Naïve method has complexity $O(m \cdot n)$, where n is the number of unbound variables

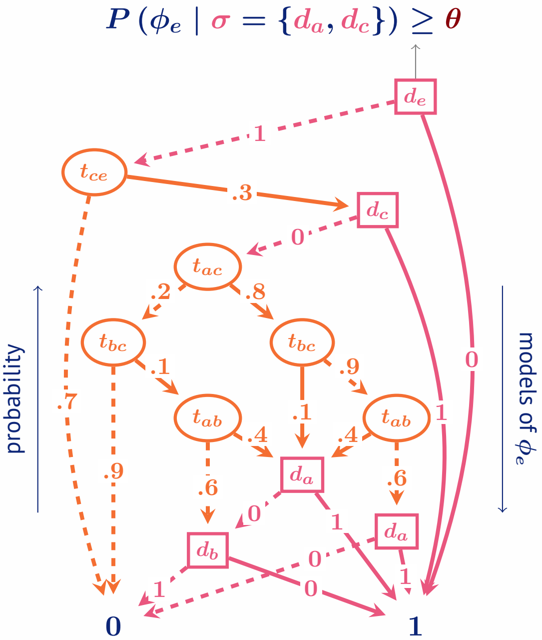


Ordered Binary Decision Diagram (**OBDD**) encodes **probability distribution** (*not* the solutions to the constraint).

solve $\sum_{\phi \in \Phi} P(\phi \mid \sigma) \geq \theta$

Use OBDD to evaluate **strategy σ**. **Complexity** of one sweep: $O(m)$, with $m = |\text{OBDD}|$.

Smart, incremental method has complexity $O(m + n)$, using **derivatives**.



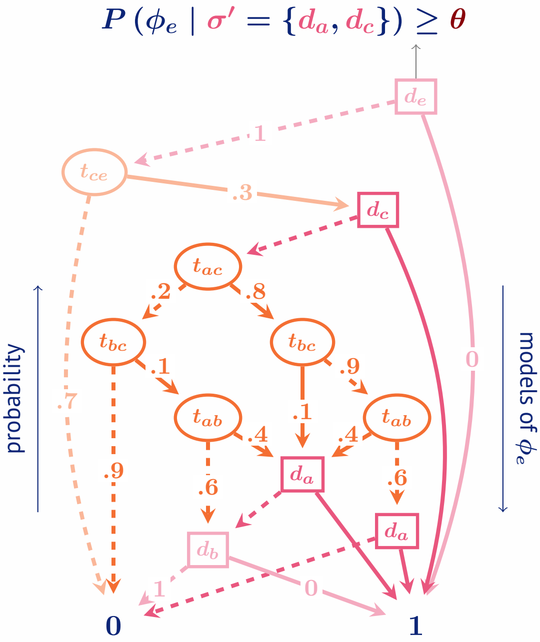
Adnan Darwiche.
On the tractable counting of theory models and its application to belief revision and truth maintenance. JANCL, 2001

Ordered Binary Decision Diagram (**OBDD**) encodes **probability distribution** (*not* the solutions to the constraint).

$$\text{solve} \sum_{\phi \in \Phi} P(\phi \mid \sigma) \geq \theta$$

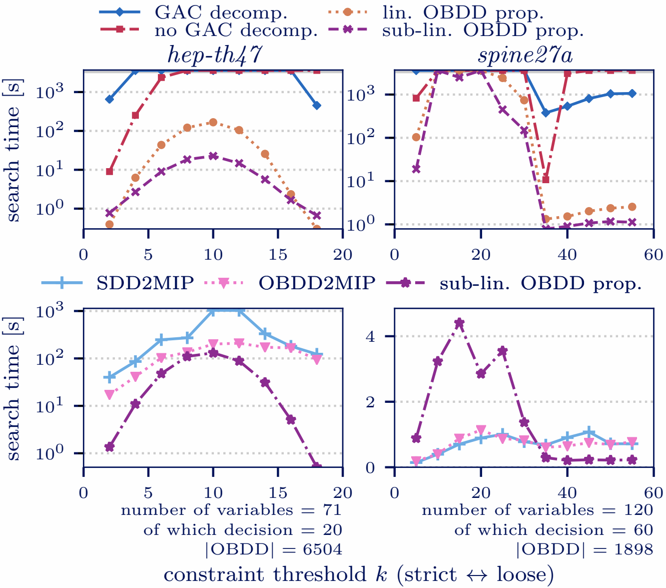
Use OBDD to evaluate **strategy σ**. **Complexity** of one sweep: $O(m)$, with $m = |\text{OBDD}|$.

Smart, incremental method has complexity $O(m+n)$, using **derivatives**.

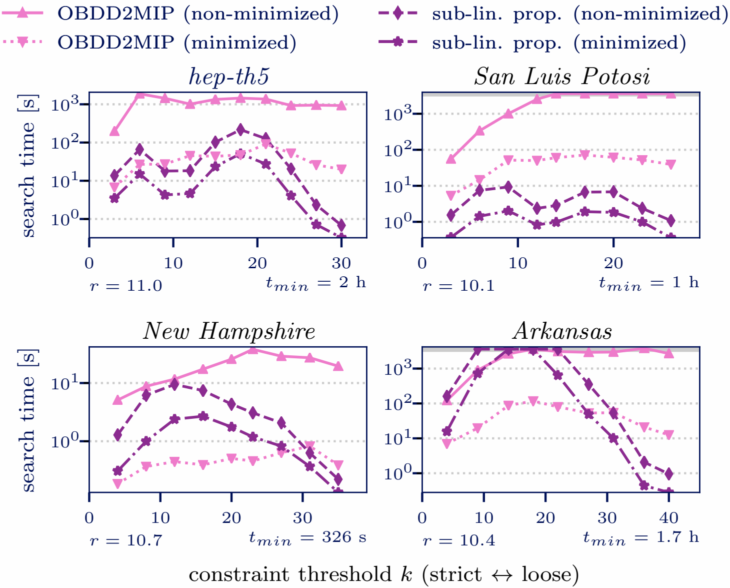


Adnan Darwiche.
On the tractable counting of theory models and its application to belief revision and truth maintenance. JANCL, 2001

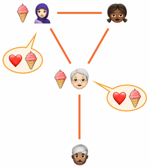
SCMD propagator vs existing methods

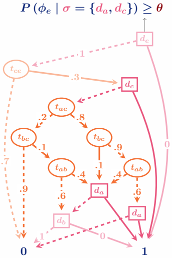


Scalability of SCMD propagator vs MIP



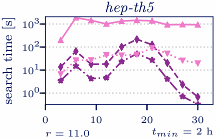
Main contribution





A new global constraint propagator for Stochastic Constraints on Monotonic Distributions (SCMDs) which:

- guarantees GAC;
- has linear complexity;
- outperforms existing CP-based methods and complements MIP-based methods;
- scales better with OBDD size than existing MIP-based methods.



contact: `a.l.d.latour@liacs.leidenuniv.nl`

code & more **results**: `github.com/latower/SCMD`

new work: D. Fokkinga, A.L.D. Latour, M. Anastacio, S. Nijssen, H. Hoos. *Programming a Stochastic Constraint Optimisation Algorithm, by Optimisation.* IJCAI Data Science meets Optimization workshop, 2019.

References I

Randal E. Bryant.
Graph-based algorithms for Boolean function manipulation.
IEEE Trans. Computers, 1986

Adnan Darwiche.
On the tractable counting of theory models and its application to belief revision and truth maintenance.
JANCL, 2001

Adnan Darwiche.
A differential approach to inference in Bayesian Networks.
ACM 2003

Luc De Raedt, Angelika Kimmig, and Hannu Toivonen
A Probabilistic Prolog and its Application in Link Discovery.
IJCAI 2007

David Kempe, Jon Kleinberg, and Éva Tardos
Maximizing the Spread of Influence Through a Social Network.
KDD 2003

References II

Anna L.D. Latour, Behrouz Babaki, Anton Dries, Angelika Kimmig, Guy Van den Broeck, and Siegfried Nijssen.
Combining Stochastic Constraint Optimization and Probabilistic Programming: From Knowledge Compilation to Constraint Solving.
CP 2017

Anna L.D. Latour, Behrouz Babaki, and Siegfried Nijssen.
Stochastic Constraint Optimization using Propagation on Ordered Binary Decision Diagrams.
StarAI 2018

OscaR Team.
OscaR: Scala in OR.
2012

Francesca Rossi, Peter van Beek, and Toby Walsh, editors
Handbook of Constraint Programming
Elsevier, 2006

References III

Dan Roth
On the Hardness of Approximate Reasoning
AI 1996

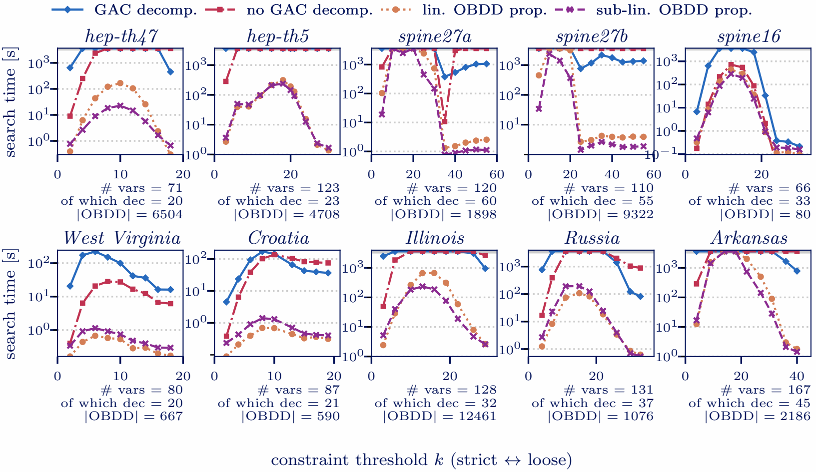
Toby Walsh
Stochastic Constraint Programming
ECAI 2002

Theme by Joost Schalken. Updated by Pepijn van Heiningen & Anna Latour.

Acknowledgements

We thank Hélène Verhaeghe for her input and suggestions. This work was supported by the Netherlands Organisation for Scientific Research (NWO). Behrouz Babaki is supported by a postdoctoral scholarship from IVADO through the Canada First Research Excellence Fund (CFREF) grant.

SCMD propagator vs existing CP methods



SCMD propagator vs existing MIP methods