Alessandro Antonucci, a senior researcher in probabilistic graphical models and machine learning

Alessandro Facchini, a convenience* logician

Lilith Mattei, research assistant, wannabe PhD student

*Concept and formulation by Yoichi Hirai
WHAT ARE CSDD? FIRST SOME ZOOLOGY
WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets (Pearl, 1984)
WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets (Pearl, 1984) \textit{Imprecise version?} Credal nets (Cozman, 2000)
WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets (Pearl, 1984)

Tractable “deep” model?

Sum-product nets (Poon & Domingos, 2012)

Imprecise version?

Credal nets (Cozman, 2000)
WHAT ARE CSDD? FIRST SOME ZOOLOGY

- **Bayesian nets** (Pearl, 1984)
- **Credal nets** (Cozman, 2000)
- **Sum-product nets** (Poon & Domingos, 2012)
- **Credal SPNs** (Mauá et al., 2017)

Imprecise version?
WHAT ARE CSDD? FIRST SOME ZOOLOGY

- Bayesian nets (Pearl, 1984)
- Credal nets (Cozman, 2000)
- Sum-product nets (Poon & Domingos, 2012)
- Credal SPNs (Mauá et al., 2017)
- ...and logical constraints?
- Probabilistic Sentential Decision Diagrams, PSDDs (Kisa et al., 2014)

Imprecise version?
WHAT ARE CSDD? FIRST SOME ZOOLOGY

- Bayesian nets (Pearl, 1984)
- Credal nets (Cozman, 2000)
- Sum-product nets (Poon & Domingos, 2012)
- Credal SPNs (Mauá et al., 2017)
- Probabilistic Sentential Decision Diagrams, PSDDs (Kisa et al., 2014)

Tractable “deep” model?
...and logical constraints?
Imprecise version?
FRAMING THE PROBLEM

WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets (Pearl, 1984)

Tractable "deep" model?

Sum-product nets (Poon & Domingos, 2012)

...and logical constraints?

Probabilistic Sentential Decision Diagrams, PSDDs (Kisa et al., 2014)

Imprecise version?

Credal nets (Cozman, 2000)

Imprecise version?

Credal SPNs (Mauá et al., 2017)

Do nice properties of CSPNs adapt to CSDD?

Imprecise version?

CSDD (here)
WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets
(Pearl, 1984)

Imprecise version?

Credal nets
(Cozman, 2000)

Tractable “deep” model?

Sum-product nets
(Poon & Domingos,
2012)

...and logical constraints?

Probabilistic Sentential
Decision Diagrams,
PSDDs (Kisa et al., 2014)

Imprecise version?

Credal SPNs
(Mauá et al.,
2017)

Do nice properties of
CSPNs adapt to CSDD?
YES!

Imprecise version?

CSDD (here)
WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets (Pearl, 1984)

Credal nets (Cozman, 2000)

Tractable “deep” model?

Sum-product nets (Poon & Domingos, 2012)

...and logical constraints?

Imprecise version?

Imprecise version?

Imprecise version?

Imprecise version?

Do nice properties of CSPNs adapt to CSDD?

Yes!

Probabilistic Sentential Decision Diagrams, PSDDs (Kisa et al., 2014)

Fast marginal inference algorithm for general CSPNs

Fast conditional inference algorithm for singly connected CSPNs

Credal SPNs (Mauá et al., 2017)

CSDD (here)
FRAMING THE PROBLEM

WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets (Pearl, 1984)

Tractable "deep" model?

Sum-product nets (Poon & Domingos, 2012)

…and logical constraints?

Probabilistic Sentential Decision Diagrams, PSDDs (Kisa et al., 2014)

Imprecise version?

Credal nets (Cozman, 2000)

Imprecise version?

Imprecise version?

Credal SPNs (Mauá et al., 2017)

Yes!

CSDD (here)

message of this work: 
CSDD’s stand to PSDD’s as CSPN’s stand to SPN’s
FRAMING THE PROBLEM

WHAT ARE CSDD? FIRST SOME ZOOLOGY

Bayesian nets (Pearl, 1984)

Tractable “deep” model?

Sum-product nets (Poon & Domingos, 2012)

…and logical constraints?

Probabilistic Sentential Decision Diagrams, PSDDs (Kisa et al., 2014)

Imprecise version?

Credal nets (Cozman, 2000)

Imprecise version?

Credal SPNs (Mauá et al., 2017)

Imprecise version?

CSDDs (here)

Do nice properties of CSPNs adapt to CSDD?

YES!

message of this work:

CSDD’s stand to PSDD’s as CSPN’s stand to SPN’s

…but what are CSDDs?
WHAT ARE CSDD’S?

A FIRST GLIMPSE TO CSDD
WHAT ARE CSDD’S?

A FIRST GLIMPSE TO CSDD

CSDD = *Credal version of Probabilistic Sentential Decision Diagrams*
WHAT ARE CSDD’S?

A FIRST GLIMPSE TO CSDD

• CSDD = *Credal version of Probabilistic Sentential Decision Diagrams*

• so, what are PSDDs?
WHAT ARE CSDD’S?

A FIRST GLIMPSE TO CSDD

- CSDD = *Credal version of Probabilistic Sentential Decision Diagrams*

  - so, what are PSDDs?
  - actually, what are SDDs?
TOY EXAMPLE (FROM KISA ET AL. 2014)

100 STUDENTS ENROLLING IN 4 CLASSES: LOGIC (L), KNOWLEDGE REPRESENTATION (K), PROBABILITY (P), AI (A)

<table>
<thead>
<tr>
<th>L</th>
<th>K</th>
<th>P</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- 16 joint states
- Three logical constraints

$ (P \lor L), \ (A \rightarrow P), \ (K \rightarrow A \lor L) $
TOY EXAMPLE (FROM KISA ET AL. 2014)

100 STUDENTS ENROLLING IN 4 CLASSES: LOGIC (L), KNOWLEDGE REPRESENTATION (K), PROBABILITY (P), AI (A)

<table>
<thead>
<tr>
<th>L</th>
<th>K</th>
<th>P</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- 16 joint states
- Three logical constraints

\[ \phi := (P \lor L) \land (A \rightarrow P) \land (K \rightarrow A \lor L) \]
100 STUDENTS ENROLLING IN 4 CLASSES: LOGIC (L), KNOWLEDGE REPRESENTATION (K), PROBABILITY (P), AI (A)

<table>
<thead>
<tr>
<th>L</th>
<th>K</th>
<th>P</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- 16 joint states
- Three logical constraints
  \[ \phi := (P \lor L) \land (A \rightarrow P) \land (K \rightarrow A \lor L) \]
- 7 states not satisfying the logical constraints (hence never observed)
TOY EXAMPLE (FROM KISA ET AL. 2014)

100 STUDENTS ENROLLING IN 4 CLASSES: LOGIC (L), KNOWLEDGE REPRESENTATION (K), PROBABILITY (P), AI (A)

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>K</th>
<th>P</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- 16 joint states
- Three logical constraints
  \[ \phi := (P \lor L) \land (A \rightarrow P) \land (K \rightarrow A \lor L) \]
- 7 states not satisfying the logical constraints (hence never observed)
- 1 state logically possible but never observed
MODELING CONSTRAINTS WITH CIRCUITS: SDD’S (DARWICHE 2011)

- A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit

$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit.

$$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$$
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit.

- take a subset of the variables, form a partition of the tautology, e.g.,

$$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$$

$$\phi = (P \lor L) \land (P \lor \neg A) \land (A \lor L \lor \neg K)$$
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit.

- take a subset of the variables, form a partition of the tautology, e.g.,
  \[ T = (\neg L \land K) \lor L \lor (\neg L \land \neg K) \]

What does $\phi$ becomes when $L = \bot$, $K = \top$?

$\phi = (P \lor L) \land (P \lor \neg A) \land (A \lor L \lor \neg K)$
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit.

Take a subset of the variables, form a partition of the tautology, e.g.,

$$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$$

\[
\phi = (P \lor L) \land (P \lor \neg A) \land (A \lor L \lor \neg K)
\]
A Sentential Decision Diagram representing $\phi$ is a "deterministic" logic circuit.

- take a subset of the variables, form a partition of the tautology, e.g.,

$$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$$

What does $\phi$ becomes when $L = \top$?

What does $\phi$ becomes when $L = \bot, K = \bot$?

$$\phi = (P \lor L) \land (P \lor \neg A) \land (A \lor L \lor \neg K)$$
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit.

Take a subset of the variables, form a partition of the tautology, e.g.,

$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$

$$\phi = (P \lor L) \land (P \lor \neg A) \land (A \lor L \lor \neg K)$$
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit.

Take a subset of the variables, form a partition of the tautology, e.g.,

$$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$$

$$\neg L \land K \land (P \land A) \lor L \land (P \lor \neg A) \lor (\neg L \land \neg K) \land P = \phi$$
A Sentential Decision Diagram representing $\phi$ is a “deterministic” logic circuit.

Take a subset of the variables, form a partition of the tautology, e.g.,

$$T = (\neg L \land K) \lor L \lor (\neg L \land \neg K)$$

$$\neg L \land K \land (P \land A) \lor L \land (P \lor \neg A) \lor (\neg L \land \neg K) \land P = \phi$$

Proceed recursively...
MODELING CONSTRAINTS WITH CIRCUITS: SDD’S (DARWICHE 2011)

\[(\neg L \land K \lor L \land \bot) \land (P \land A \lor \neg P \land \bot) \lor (L \land T \lor \neg L \land \bot) \land (\neg P \land \neg A \lor P \land T) \lor (\neg L \land \neg K \lor L \land \bot) \land (P \land T \lor \neg P \land \bot) = \phi\]
A Probabilistic Sentential Decision Diagrams (PSDDs) for $\phi$ is a parametrized SDD:
A Probabilistic Sentential Decision Diagrams (PSDDs) for $\phi$ is a parametrized SDD:

Parameters learned from data
MODELING DATA + CONSTRAINTS WITH CIRCUITS: PSDD’S (KISA, 2014)

- A Probabilistic Sentential Decision Diagrams (PSDDs) for. $\phi$ is a parametrized SDD:

  - Parameters learned from data

  - Inducing a joint probability $\mathbb{P}(A, L, P, K)$
CONSTRAINTS FIRST, DATA AFTER: PSDD

MODELING DATA + CONSTRAINTS WITH CIRCUITS: PSDD’S (KISA, 2014)

- A Probabilistic Sentential Decision Diagrams (PSDDs) for. $\phi$ is a parametrized SDD:

- Parameters learned from data

- Inducing a joint probability $\mathbb{P}(A, L, P, K)$
**MODELING DATA + CONSTRAINTS WITH CIRCUITS: PSDD’S (KISA, 2014)**

- **A Probabilistic Sentential Decision Diagrams (PSDDs)** for $\phi$ is a parametrized SDD:

- Parameters learned from data

- Inducing a joint probability $\mathbb{P}(A, L, P, K)$
A Probabilistic Sentential Decision Diagrams (PSDDs) for $\phi$ is a parametrized SDD:

- Parameters learned from data
- Inducing a joint probability $\mathbb{P}(A, L, P, K)$
- Context-specific independences $\mathbb{P}$ derived from the structure
A Probabilistic Sentential Decision Diagrams (PSDDs) for. \( \phi \) is a parametrized SDD:

- Parameters learned from data
- Inducing a joint probability \( \mathbb{P}(A, L, P, K) \)
- context-specific independences wrt \( \mathbb{P} \) derived from the structure
- Logically impossible events have zero probability: \( \mathbb{P}(x) > 0 \iff x \models \phi \)
DEFINING CSDD’S

CREDAL VERSION OF PSDD’S:
DEFINING CSDD’S

CREDAL VERSION OF PSDD’S: REPLACE PMF’S WITH CS’S
DEFINING CSDD’S

CREDAL VERSION OF PSDD’S: REPLACE PMF’S WITH CS’S

- Credal Sentential Decision Diagrams (CSDDs) for $\phi$

![Diagram showing CSDDs for $\phi$]
CREDAL VERSION OF PSDD’S: REPLACE PMF’S WITH CS’S

- Credal Sentential Decision Diagrams (CSDDs) for ϕ

- Syntax: CS attached to each decision node and to each terminal node T
DEFINING CSDD’S

CREDAL VERSION OF PSDD’S: REPLACE PMF’S WITH CS’S

- Credal Sentential Decision Diagrams (CSDDs) for $\phi$

- Syntax: CS attached to each decision node and to each terminal node $T$

- Semantics: collection of consistent PSDDs
DEFINING CSDD’S

CREDAL VERSION OF PSDD’S: REPLACE PMF’S WITH CS’S

- Credal Sentential Decision Diagrams (CSDDs) for $\phi$

- Syntax: CS attached to each decision node and to each terminal node $\top$

- Semantics: collection of consistent PSDDs

- PSDD induces joint $P$, CSDD induces joint CS ("Strong extension")
Marginal queries:

Given evidence $e$, calculate

$$
\mathbb{P}(e) = \min_{\mathbb{P}(X) \in \mathbb{K}(X)} \mathbb{P}(e)
$$
Marginal queries:

Given evidence $e$, calculate

$$\mathbb{P}(e) = \min_{\mathbb{P}(X) \in \mathbb{K}(X)} \mathbb{P}(e)$$

Conditional queries:

Given available evidence $e$ and queried variable, calculate

$$\mathbb{P}(x \mid e) = \min_{\mathbb{P}(X) \in \mathbb{K}(X)} \frac{\mathbb{P}(x, e)}{\mathbb{P}(e)}$$
TWO POLYTIME ALGORITHMS

Adaptation of CSPNs algorithms (Mauá et al.) to CSDDs:

**Marginal queries:**
- Bottom-up propagation of LP task’s results
- Coefficients of each LP task are computed in the lower level
- Feasible regions are the local CSs

**Conditional queries:**
- Decisional version of original task
- Bottom-up propagation of LP task’s results
- Coefficients of each LP task are computed in the lower level, depending on evidence
- Feasible regions are the local CSs
Adaptation of CSPNs algorithms (Mauá et al.) to CSDDs:

- **Marginal queries:**
  - Bottom-up propagation of LP task’s results
  - Coefficients of each LP task are computed in the lower level
  - Feasible regions are the local CSs

- **Conditional queries:**
  - Decisional version of original task
  - Bottom-up propagation of LP task’s results
  - Coefficients of each LP task are computed in the lower level, depending on evidence
  - Feasible regions are the local CSs

Needs singly connected topology
CONCLUSIONS AND FUTURE WORK

- CSDDs as a new tool for sensitivity analysis in PSDD
- Robust marginalisation and conditioning (for singly connected circuits) with poly complexity
- Application to “credal” ML with structured spaces
- Complexity and approximations results for multiply connected CSDDs
- Hybrid (structured/unstructured) models
- Structural learning (trade-off small SDD / likelihood / independences)
- CNs vs. CSDDs ?
CONCLUSIONS AND FUTURE WORK

- CSDDs as a new tool for sensitivity analysis in PSDD
- Robust marginalisation and conditioning (for singly connected circuits) with poly complexity
- Application to “credal” ML with structured spaces
- **Complexity and approximations results for multiply connected CSDDs**
- Hybrid (structured/unstructured) models
- Structural learning (trade-off small SDD / likelihood / independences)
- **CNs vs. CSDDs ?**
Credal Sentential Decision Diagrams (CSDDs)
Alessandro Antonucci, Alessandro Facchini, Lilith Mattei
(alessandro, alessandro.facchini, lilith)@idsia.ch

FROM SDDs TO CSDDs (THROUGH PSDDs)

- Logical skeleton? \( \phi \) as a circuit alternating OR AND gates
- This is a sentential decision diagram, (SDD, Choi & Darwiche, 2013)
- Probabilistic model? Probability mass functions annotating the OR gates of the SDD (PSDDs)
- PSDD is a joint probability mass function consistent with the constraints
  \[ P_L(K,P,A) \mid P(L,K,p,x) = 0 \text{ if } (L,k,p,x) \not\models \phi \]
- CSDD: Credal version of PSDD: credal sets instead of mass functions
- Credal sets on OR gates and terminal nodes \( \top \)
- Semantics: all PSDD with parameters consistent with the local credal sets
- Strong extension \( R(L,K,P,A) \) as the joint credal set of all the joint mass functions induced by the consistent PSDDs
- CSDD Inference? Lower/upper bounds sort the strong extension
- Bayes theorem: for each \( L \) \( P_L(x) > 0 \) if \( \not\models \phi \) and \( P_L(x) = 0 \) if \( \models \phi \)
- Learning CSDD Parameters are conditional probabilities, \( \not\models \phi \)
- Inverse Decision Model to learn local (conditional) credal sets
- Data scarcity issue on the leaves justifies imprecise approach!

REFERENCES
- Christian Poirier, Louis Gailly. Bayesian nets as classical (precise) probabilistic graphical models (BNs)
- With imprecise probabilities? Credal networks (CNs, Cozman, 2000)
- With deep structure (and tractable inference)? Sum-product networks (SPN, Poon & Domingos, 2011)
- With deep structure and imprecise probabilities? Credal sum-product networks (CSNP, Mauá et al., 2017)
- With deep structure and embedding logical constraints? Probabilistic sentential decision diagrams (PSDD, Kisa et al., 2014)
- Deep structure, imprecise probabilities and logical constraints?
- Credal sentential decision diagrams (CSDD, this paper)

FROM SDDs TO CSDDs (THROUGH PSDDs)

- Logical skeleton? \( \phi \) as a circuit alternating OR AND gates
- This is a sentential decision diagram, (SDD, Choi & Darwiche, 2013)
- Probabilistic model? Probability mass functions annotating the OR gates of the SDD (PSDDs)
- PSDD is a joint probability mass function consistent with the constraints
  \[ P_L(K,P,A) \mid P(L,K,p,x) = 0 \text{ if } (L,k,p,x) \not\models \phi \]
- CSDD: Credal version of PSDD: credal sets instead of mass functions
- Credal sets on OR gates and terminal nodes \( \top \)
- Semantics: all PSDD with parameters consistent with the local credal sets
- Strong extension \( R(L,K,P,A) \) as the joint credal set of all the joint mass functions induced by the consistent PSDDs
- CSDD Inference? Lower/upper bounds sort the strong extension
- Bayes theorem: for each \( L \) \( P_L(x) > 0 \) if \( \not\models \phi \) and \( P_L(x) = 0 \) if \( \models \phi \)
- Learning CSDD Parameters are conditional probabilities, \( \not\models \phi \)
- Inverse Decision Model to learn local (conditional) credal sets
- Data scarcity issue on the leaves justifies imprecise approach!

CONCLUSIONS & OUTLOOKS

- CSDDs as a new tool for sensitivity analysis in PSDD
- Fast robust marginalisation and conditioning (but conditioning works for singly connected circuits only)
- Complexity results and approximated algorithm are needed
- CNs vs. CSDDs? Credal classification with CSDDs?

CONCLUSIONS & OUTLOOKS

- CSDD as a tool for sensitivity analysis in PSDD
- Fast robust marginalisation and conditioning (but conditioning works for singly connected circuits only)
- Complexity results and approximated algorithm are needed
- CNs vs. CSDDs? Credal classification with CSDDs?

REFERENCES
- Christian Poirier, Louis Gailly. Bayesian nets as classical (precise) probabilistic graphical models (BNs)
- With imprecise probabilities? Credal networks (CNs, Cozman, 2000)
- With deep structure (and tractable inference)? Sum-product networks (SPN, Poon & Domingos, 2011)
- With deep structure and imprecise probabilities? Credal sum-product networks (CSNP, Mauá et al., 2017)
- With deep structure and embedding logical constraints? Probabilistic sentential decision diagrams (PSDD, Kisa et al., 2014)
- Deep structure, imprecise probabilities and logical constraints?
- Credal sentential decision diagrams (CSDD, this paper)

FROM SDDs TO CSDDs (THROUGH PSDDs)

- Logical skeleton? \( \phi \) as a circuit alternating OR AND gates
- This is a sentential decision diagram, (SDD, Choi & Darwiche, 2013)
- Probabilistic model? Probability mass functions annotating the OR gates of the SDD (PSDDs)
- PSDD is a joint probability mass function consistent with the constraints
  \[ P_L(K,P,A) \mid P(L,K,p,x) = 0 \text{ if } (L,k,p,x) \not\models \phi \]
- CSDD: Credal version of PSDD: credal sets instead of mass functions
- Credal sets on OR gates and terminal nodes \( \top \)
- Semantics: all PSDD with parameters consistent with the local credal sets
- Strong extension \( R(L,K,P,A) \) as the joint credal set of all the joint mass functions induced by the consistent PSDDs
- CSDD Inference? Lower/upper bounds sort the strong extension
- Bayes theorem: for each \( L \) \( P_L(x) > 0 \) if \( \not\models \phi \) and \( P_L(x) = 0 \) if \( \models \phi \)
- Learning CSDD Parameters are conditional probabilities, \( \not\models \phi \)
- Inverse Decision Model to learn local (conditional) credal sets
- Data scarcity issue on the leaves justifies imprecise approach!

CONCLUSIONS & OUTLOOKS

- CSDDs as a new tool for sensitivity analysis in PSDD
- Fast robust marginalisation and conditioning (but conditioning works for singly connected circuits only)
- Complexity results and approximated algorithm are needed
- CNs vs. CSDDs? Credal classification with CSDDs?

REFERENCES
- Christian Poirier, Louis Gailly. Bayesian nets as classical (precise) probabilistic graphical models (BNs)
- With imprecise probabilities? Credal networks (CNs, Cozman, 2000)
- With deep structure (and tractable inference)? Sum-product networks (SPN, Poon & Domingos, 2011)
- With deep structure and imprecise probabilities? Credal sum-product networks (CSNP, Mauá et al., 2017)
- With deep structure and embedding logical constraints? Probabilistic sentential decision diagrams (PSDD, Kisa et al., 2014)
- Deep structure, imprecise probabilities and logical constraints?
- Credal sentential decision diagrams (CSDD, this paper)