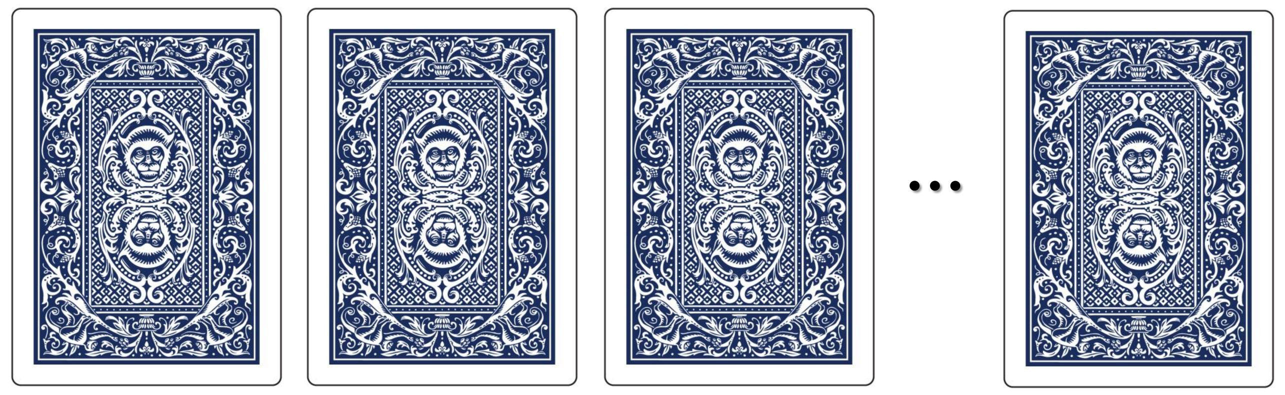
New Liftable Classes for First-Order Probabilistic Inference

Seyed Mehran Kazemi, Angelika Kimmig, Guy Van den Broeck & David Poole

UBC UCLA KU LEUVEN

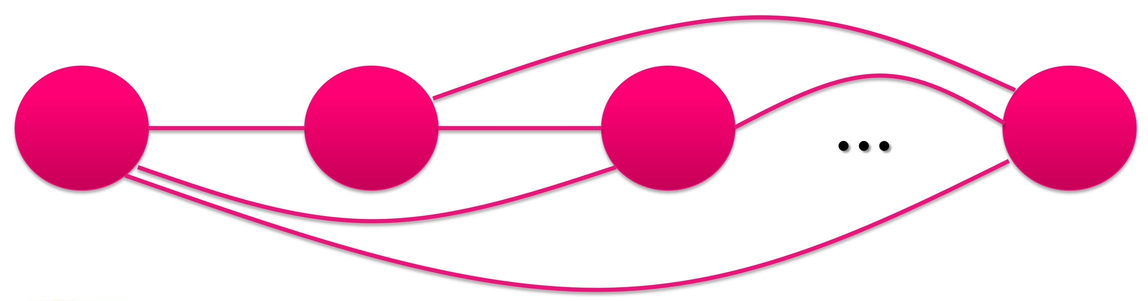
Motivation



Simple Probability Queries:

- P(First Card is a ♥) = ?
- P(First Card is the Q of ♥) = ?
- P(Last Card is a ♦ | First Card is a ♥) = ?
- …

Automating the query answering using, e.g., a Markov network



- 52 variables (nodes), each taking 52 values
- All variables connected to each other (no independence)
- Classical reasoning algorithms require a table with 52^{52} rows (tree-width = 52)

Why is answering such simple queries so difficult for classical reasoning algorithms?

P(First Card is the Q of ♥) = P(Second Card is the Q of ♥) = …= P(Last Card is the Q of ♥)

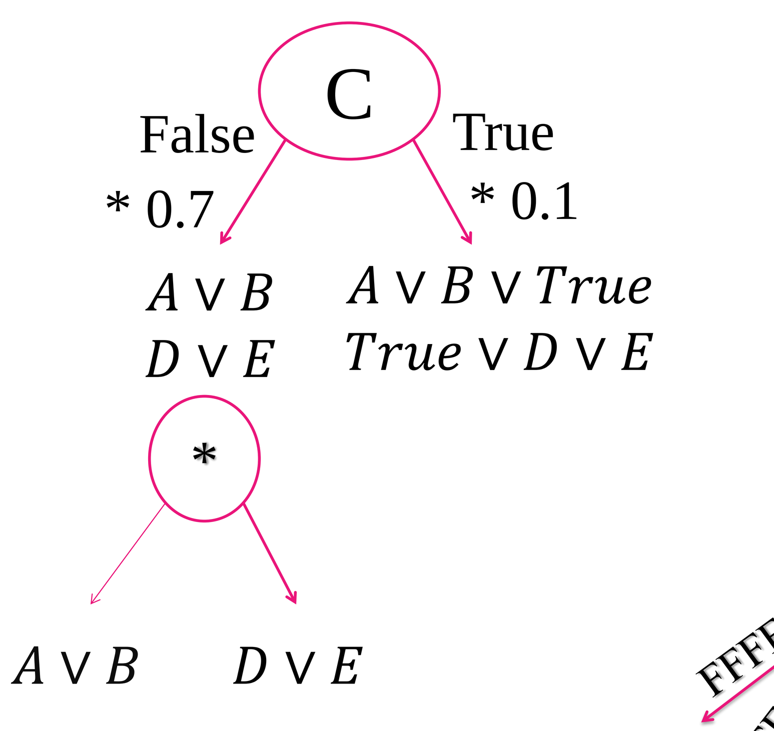
- The cards are exchangeable: swapping two cards does not change the probability distribution
- Classical reasoning algorithms do not exploit exchangeability
- Lifted (first-order) inference algorithms aim at speeding up inference by exploiting exchangeability

Weighted Model Counting (WMC)

Here are a set $\boldsymbol{R}$ of rules for efficient WMC:

$A \vee B \vee C$
$C \vee D \vee E$

w(A=True) = 0.2
w(A=False) = 0.6
w(B=True) = 1.1
w(B=False) = 1
w(C=True) = 0.1
w(C=False) = 0.7
w(D=True) = 2.2
w(D=False) = 0.5
w(E=True) = 1.5
w(E=False) = 2



Cache the results

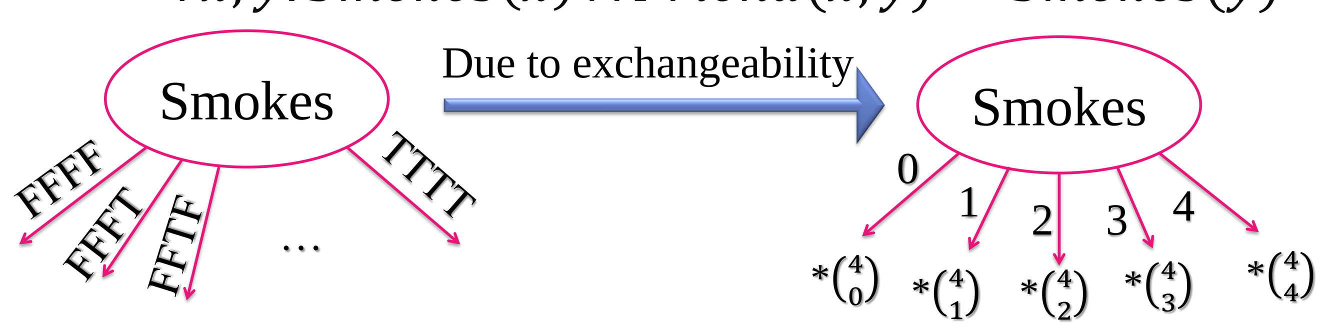
$\forall x: Smokes(x) \Rightarrow Cancer(x) \quad x \in \{A,B,C,D\}$

- $\boxed{Smokes(A) \Rightarrow Cancer(A)}$
- …
- $\boxed{Smokes(D) \Rightarrow Cancer(D)}$

*

Due to exchangeability: $\left[\boxed{Smokes(A) \Rightarrow Cancer(A)}\right]^4$

$\forall x,y: Smokes(x) \wedge Friend(x,y) \Rightarrow Smokes(y)$



Deck of Cards Example Revisited

- Instead of a Markov net, formulate the problem as follows:
  - $\forall pos, \exists card: In(card, pos)$
  - $\forall card, \exists pos: In(card, pos)$
  - $\forall pos, card1, card2 \neq card1: \neg In(card1, pos) \vee \neg In(card2, pos)$
- This formulation reveals the exchangeabilities/symmetries of the problem

Using the rules in $\boldsymbol{R}$ (and some preprocessing):

$$WMC = \sum_{k=0}^{n} \binom{n}{k} \sum_{l}^{n} \binom{n}{l} (l+1)^k (-1)^{2n-k-l} \in O(n^2)$$

Note: Any query can be converted into calculating two WFOMCs

Domain Recursion Rule

Consider the theory: $\forall x, y \neq x: Fr(x,y) \Rightarrow Fr(y,x) \qquad$ x, y ∈ {A,B,…,Z}

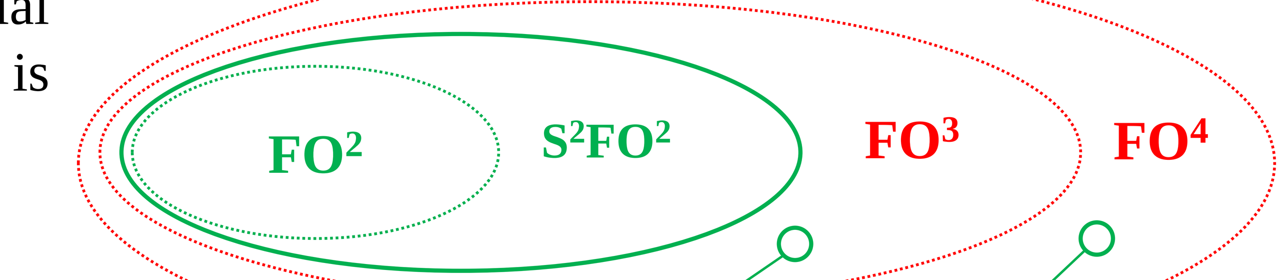
Reveal/Separate one person (e.g., A) from the population:
$\forall y': Fr(A,y') \Rightarrow Fr(y',A) \qquad \forall x': Fr(x',A) \Rightarrow Fr(A,x') \qquad \forall x', y' \neq x': Fr(x',y') \Rightarrow Fr(y',x') \qquad x', y' \in \{B,\ldots,Z\}$

Apply the rules in $\boldsymbol{R}$ on $Fr(A,y')$ and $Fr(x',A)$: $\forall x', y' \neq x': Fr(x',y') \Rightarrow Fr(y',x') \quad x', y' \in \{B,\ldots,Z\}$

This theory is equivalent to the initial theory, with the population size reduced by one.
Using a cache, the WFOMC of the above theory can be computed in polynomial time using dynamic programming.

Theoretical Results

- **Definitions:** A theory is **liftable** if calculating its WMC is polynomial in sizes of the populations. A class C is **liftable** if every $T \in C$ is liftable. $\boldsymbol{FO^i}$: class of theories with up to i variables per sentence.
- **Previously proved (without domain recursion)**
  - $\boldsymbol{FO^2}$ is liftable.
  - Not every $T \in \boldsymbol{FO^3}$ is liftable.
- **Our results (using domain recursion)**
  - $\boldsymbol{S^2FO^2}$ is liftable and subsumes $\boldsymbol{FO^2}$.
  - Symmetric transitivity and S4 are liftable.



Symmetric Transitivity
$\forall x,y,z: F(x,y) \wedge F(y,z) \Rightarrow F(x,z)$
$\forall x,y: F(x,y) \Rightarrow F(y,x)$

S4 Clause
$\forall x_1,x_2,y_1,y_2: S(x_1,y_1) \vee \neg S(x_1,y_2) \vee S(x_2,y_2) \vee \neg S(x_2,y_1)$

A New Liftable Class: S^2FO^2

- Let $T \in FO^2$, $S(x,m) \in T$, and for any sentence $c \in T$, if $S(x,m) \in c$, all other atoms in c have at most one variable.
- Add any sentence $\alpha(S)$ to T having exactly 2 S atoms, e.g.:
  - $\forall x, m_1, m_2: S(x,m_1) \vee S(x,m_2)$
  - $\forall x, m_1, m_2: \neg S(x,m_1) \vee S(x,m_2)$
  - $\forall x_1, x_2, m: S(x_1,m) \vee S(x_2,m)$
  - $\forall x_1, x_2, m_1, m_2: S(x_1,m_1) \vee S(x_2,m_2)$
  - …
- T is in S^2FO^2

Example: Volunteers (v) & Jobs (j)

$$\left.\begin{array}{r} \forall j, v: InvolvesGas(j) \wedge Smokes(v) \Rightarrow \neg Assigned(j,v) \\ \forall v1, v2: AUX(v1,v2) \Leftrightarrow Smokes(v1) \wedge Friends(v1,v2) \Rightarrow Smokes(v2) \end{array}\right\} \in FO^2$$

$$\left.\begin{array}{r} \forall v1, v2 \neq v1, j: \neg Assigned(j,v1) \vee \neg Assigned(j,v2) \\ \forall v, j1, j2 \neq j1: \neg Assgined(j1,v) \vee \neg Assgined(j2,v) \end{array}\right\} \in \alpha(Assgined)$$

Clause1: Jobs involving gas are not assigned to smokers
Clause2: Smokers are mostly friends with each other
Clause3: Each volunteer is assigned to at most one job
Clause4: At most one volunteer is assigned to any job

Empirical Evaluation

Theories from left to right:

- Volunteers & jobs
- Symmetric transitivity
- S4 clause